Lecture Notes in Control and Information Sciences 236

Editor: M. Thoma

Anibal T. de Almeida and Oussama Khatib (Eds)

Autonomous Robotic Systems

Springer

Editors

Professor Anibal T. de Almeida
Instituto de Sistemas e Robótica
Departamento de Engenharia Electrotécnica, Universidade de Coimbra,
Polo II, 3030 Coimbra, Portugal

Professor Oussama Khatib
Department of Computer Science, University of Stanford, Palo Alto, CA 94305,
USA

ISBN 1-85233-036-8 Springer-Verlag Berlin Heidelberg New York

British Library Cataloguing in Publication Data
Autonomous robotic systems. - (Lecture notes in control and
information sciences ; 236)
1.Robotics 2.Automation
I.Almeida, Anibal T. de II.Khatib, O. (Oussama)
629.8'92
ISBN 1852330368

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the Library of Congress

Printed in Great Britain

The use of registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant laws and regulations and therefore free for general use.

The publisher makes no representation, express or implied, with regard to the accuracy of the information contained in this book and cannot accept any legal responsibility or liability for any errors or omissions that may be made.

Typesetting: Camera ready by editors
Printed and bound at the Athenæum Press Ltd., Gateshead, Tyne & Wear
69/3830-543210 Printed on acid-free paper

Preface

The Advanced Research Workshop on "*Autonomous Robotic Systems*" was held in the University of Coimbra, in Coimbra, Portugal, from June 19 to 21, 1997. The aim of this meeting was to bring together leading researchers from around the world to present and discuss the recent developments in the area of autonomous systems for mobility and manipulation. The presentations at the workshop were made by researchers from Europe, Asia, and North America, and the meeting was attended by 80 participants from 15 countries.

Autonomous robotic systems have been the focus of much attention in recent years and significant progress has been made in this growing area. These efforts have resulted in a host of successful applications. However, there is a vast potential for new applications, which require further research and technological advances.

This volume includes the key contributions presented at the workshop. These contributions represent a wide coverage of the state-of-the-art and the emerging research directions in autonomous robotic systems. The material was developed in an advanced tutorial style making its contents more accessible to interested readers. These contributions are organised in four parts: Sensors and Navigation, Cooperation and Telerobotics, Applications, and Legged and Climbing Robots.

The first part concerns sensors and navigation in mobile robotics. An effective navigation system developed for natural unstructured environments, as well as its implementation results on a cross-country rover, are presented. Various active vision systems, with potential application to surveillance tasks, are described, and the integration of active vision in mobile platforms is analysed. A survey of sensors for mobile robot navigation is presented. The synergy of combining inertial sensors with absolute sensors seems to overcome the limitations of either type of systems when used alone. The emerging area of odour sensors in mobile robotics, based on biological systems, is also analysed.

The second part focuses on cooperation and telerobotics. Different approaches for the generation of smooth robot motion and desired forces in a natural way, are outlined. Issues of position versus velocity control are discussed and alternatives to force-reflection and pure force feed-forward are described. Cooperation is central to distributed autonomous robot systems. The development of cooperative behaviours is discussed from a local and global coordination point of view and new cooperation methodologies are proposed. Mobile manipulation capabilities are key to many new applications of robotics. The inertial properties of holonomic mobile manipulation systems are discussed, and the basic strategies developed for their dynamic coordination and control are presented.

The third part is devoted to applications. Existing and emerging new applications of autonomous systems are discussed. These applications include operations in the forestry sector, floor cleaning in buildings, mining industry, hospitals and tertiary buildings, assistance to the elderly and handicapped, and surgery.

The fourth part is concerned with legged and climbing robots. These machines are becoming increasingly important for dealing with highly irregular environments and steep surfaces. A survey of walking and climbing machines, as well as the characterisation of machines with different configurations, are presented.

On behalf of the Organising Committee, we would like to express our appreciation and thanks to the European Commission, Junta Nacional de Investigacao Cientifica e Tecnologica, FLAD, and the University of Coimbra, for the financial support they extended to this workshop. Also we would like to thank the University of Coimbra and the Department of Electrical Engineering for hosting the workshop.

Our special thanks go to the researchers, staff, and students of the Institute of Systems and Robotics, who generously gave of their time to help in the organisation of this meeting.

The Editors

Anibal T. de Almeida
Oussama Khatib

February, 1998

Contents

Part IV - Legged and Climbing Robots

Part One

Sensors and Navigation

Autonomous Outdoor Mobile Robot Navigation: The EDEN Project
Raja Chatila, Simon Lacroix, Michel Devy, Thierry Siméon

Active Vision for Autonomous Systems
Helder J. Araújo, J. Dias, J. Batista, P. Peixoto

Sensors for Mobile Robot
Jorge Lobo, Lino Marques, J. Dias, U. Nunes, A.T. de Almeida

Application of Odour Sensors in Mobile Robotics
Lino Marques, A.T. de Almeida

Autonomous Outdoor Mobile Robot Navigation The EDEN Project

Raja Chatila   Simon Lacroix   Michel Devy
Thierry Siméon
LAAS-CNRS
7, Ave. du Colonel Roche
31077 Toulouse Cedex 4, France
E-mail: {raja,simon,michel,nic}@laas.fr

Abstract

A cross-country rover cannot rely in general on permanent and immediate communications with a control station. This precludes direct teleoparation of its motions. It has therefore to be endowed with a large autonomy in achieving its navigation. We have designed and experimented with the mobile robot ADAM a complete system for autonomous navigation in a natural unstructured environment. We describe this work in this paper. The approach is primarly based on the adaptation of the perception and motion actions to the environment and to the status of the robot. The navigation task involves several levels of reasoning, several environment representations, and several action modes. The robot is able to select sub-goals, navigation modes, complex trajectories and perception actions according to the situation.

1 Introduction

Navigation is the basic task that has to be solved by a cross-country rover. Effectiveness in achieving the task is essential given the constraints of energy. Navigation is in general an incremental process that can be summarized in four main steps:

- Environment perception and modelling: any motion requires a representation of the local environment at least, and often a more global knowledge. The navigation process has to decide where, when and what to perceive.

- Localization: the robot needs to know where it is with respect to its environment and goal.

- Motion decision and planning: the robot has to decide where or which way to go, locally or at the longer term, and if possible compute a trajectory for avoiding obstacles and terrain difficulties;

- Motion execution: the commands corresponding to the motion decisions are executed by control processes - possibly sensor-based and using environment features.

The complexity of the navigation processes depends on the general context in which this task is to be executed (nature of the environment, efficiency constraints,...) and should be adapted to it.

Navigation in outdoors environments was addressed either for specific tasks, e.g., road following [8], or motion in rather limited environment conditions [9, 10]. Ambler [4] is a legged machine and the main problem was computing footholes. The UGV, as well as Ratler [4] achieve autonomous runs avoiding obstacles, but not coping to our knowledge with irregular terrain.

In a natural outdoor environment, the robot has to cope with different kinds of terrain: flat with scattered rocks or irregular/uneven in which its motion control system should take into account its stability. Limitations of computing capacities (processing power and memory) and of power consumption on the one hand, and the objective of achieving an efficient behaviour on the other hand, have lead us to consider an *adaptive* approach to mobile robot navigation in natural environments.

The objective of the EDEN project described here is to achieve a canonical navigation task, i.e., the task: `''Go To [goal]''`, where the argument `goal` is a distant target to reach autonomously. Any more complex robotic mission (exploration, sample collection...) will include one or more instances of this task. Given the variety of terrains the robot will have to traverse, this task involves in our approach several levels of reasoning, several environment representations, and various motion modes. It raises a need for a specific decisional level (the *navigation* level), that is in charge of deciding which environment representation to update, which sub-goal to reach, and which motion mode to apply. This level, which is a key component of the system, controls the perception and motion activities of the robot for this task.

The paper is organised as follows: the next section presents the adaptive approach to autonomous navigation in unknown outdoor environments. Section 3 then mentions the various terrain models required during navigation, and presents how the terrain representations required by the navigation decisional level and for the purpose of localization are incrementally built. The algorithms that produce motion plans, both at the navigation and trajectory level, are described in section 4. Finally, we present experimental results and conclude the paper.

2 A General Strategy for Navigation in Outdoor Unknown Environments

2.1 An adaptive approach

Using its own sensors, effectors, memory and computing power efficiently is certainly a feature that we would like to implement in a robot. This becomes even more a necessity for a rover (such as a planetary rover for instance) which has important limitations on its processing capacities, memory and energy. To achieve an efficient behavior, the robot must adapt the manner in which it executes the navigation task to the nature of the terrain and the quality of its knowledge on it [2, 5]. Hence, three motion modes are considered:

- A **reflex** mode: on large flat and lightly cluttered zones, it is sufficient to determine robot locomotion commands on the basis of a goal (heading or position) and informations provided by "obstacle detector" sensors. The terrain representation required by this mode is just the description of the borders of the region within which it can be applied;
- A **2D planned** mode: when the terrain is mainly flat, but cluttered with obstacles, it becomes necessary for efficiency reasons to plan a trajectory. The trajectory planner reasons on a binary description of the environment, which is described in terms of empty/obstacle areas.
- A **3D planned** mode: when the terrain is highly constrained (uneven), collision and stability constraints have to be checked to determine the robot locomotion commands. This is done thanks to a 3D trajectory planner 4.2, that reasons on a fine 3D description of the terrain (an elevation map or numerical terrain model [6]);

The existence of different motion modes enables more adpated and efficient behavior, at the price of complicating the system since it must be able to deal with several different terrain representations and motion planning processes. It must especially have the ability to determine which motion mode to apply: this is performed thanks to a specific planning level, the *navigation planner* which requires its own representations.

2.2 The Navigation Planner

We assume the terrain on which the robot must navigate is initially unknown, or mapped with a very low resolution. In this last case, it is possible for a user to specify a graph of possible *routes*, *i.e.* corridors that avoid large difficult areas, and within which the robot has to move autonomously. In this context, the navigation task ` ` `Go To` ' ' is achieved at three layers of planning (figure 1):

- *route planning* which selects long-term paths to the goal on the basis of the initial informations when available (the route map that may cover

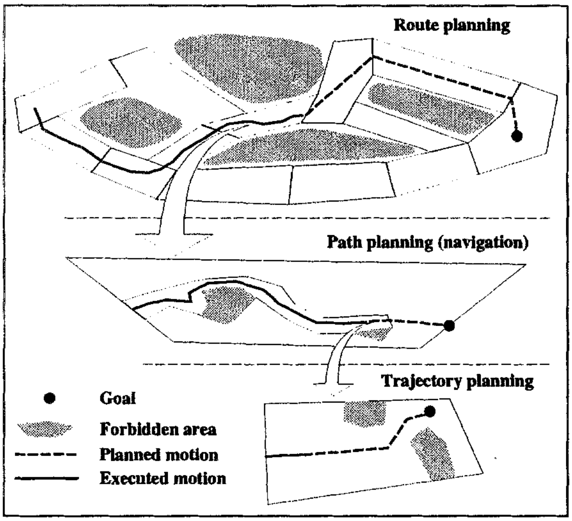


Figure 1: *Three levels of planning*

several kilometers). The route planner selects a sub-goal for the navigation planning level;

- *navigation planning* which reasons on a global qualitative representation of the terrain, built from the data acquired by the robot's sensors. The navigation planner selects the next perception task to perform, the sub-goal to reach and the motion mode to apply: it selects and controls the trajectory planner;

- *trajectory planning* which determines the trajectory to execute (in one of the above-mentioned three motion modes) to reach the goal defined by the navigation planning level.

Splitting the decisional process into three layers of planning has the advantage of structuring the problem: each planning layer controls the one that is directly below by specifying its goal and its working domain. It has also the great advantage of helping to analyze failures: when a planner fails to reach its goal, it means that the environment representation of the immediatly higher layer is erroneous and therefore that it has to be revised.

The navigation planner (layer 2 - section 4.1) is *systematically* activated at each step of the incremental execution of the task: each time 3D data are acquired, they are analyzed to provide a description of the perceived zone in terms of navigation classes. This description is fused with the model acquired so far to maintain a *global qualitative representation* of the environment, (the *region map* - section 3.1), on which the navigation planner reasons to select a sub-goal, a motion mode, and the next perception task to execute.

3 Several terrain representations

Each of the three different motion modes requires a particular terrain representation. The navigation planner also requires a specific terrain representation, and during navigation, an exteroceptive localisation process has to be activated frequently, which requires an other terrain representation. Aiming at building a "universal" terrain model that contains all the necessary informations for these various processes is extremely difficult, inefficient, and moreover not really useful. It is more direct and easier to build different representations adapted to their use: the environment model is then *multi-layered and heterogeneous*. Several perception processes coexist in the system, each dedicated to the extraction of specific representations: perception is *multi-purpose*.

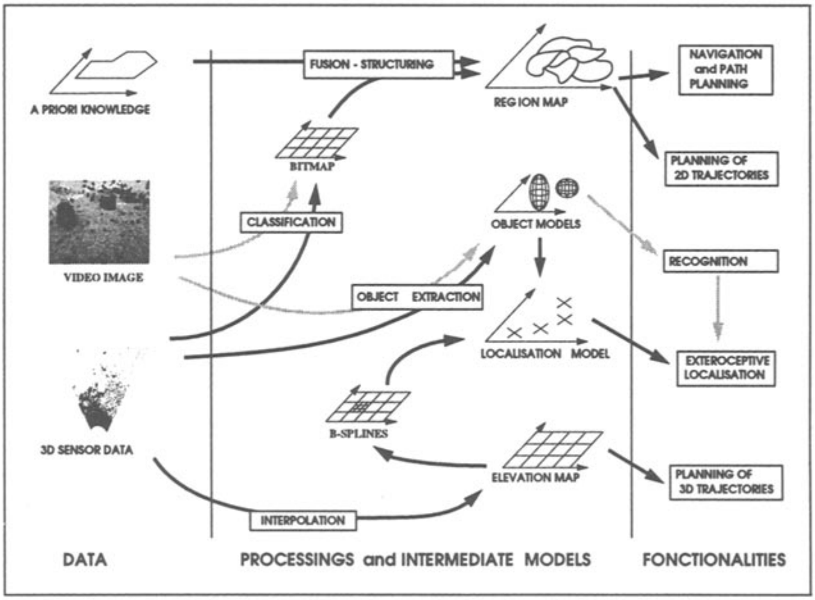


Figure 2: *The various representations used in the system. Arrows represent the constructive dependencies between them*

Figure 2 presents the various terrain representations required during navigation: one can distinguish the numerical terrain model [6] necessary to the 3D trajectory planner, the region map dedicated to the navigation planner, and three different ways to build a localisation model: *(i)* by modelling objects (rocks) with ellipsoids or superquadrics [1], *(ii)* by detecting interesting zones in the elevation map represented by a B-spline based model [3], or *(iii)* by detecting poles in the 3D raw data. Coherence relationships between these various representations are to be maintained when necessary.

3.1 Building the region map

For the purpose of navigation planning, a global representation that describes the terrain in terms of navigation classes (flat, uneven, obstacle, unknown) is required. This representation enables to select the adequate motion mode. We focus in this section on the algorithms developed to build such a model from 3D data (produced either by a laser range finder or a correlation stereo-vision algorithm).

3.1.1 3D data classification

Applied each time 3D data are acquired, the classification process produces a description of the perceived areas in term in *terrain classes*. It relies on a specific discretization of the perceived area respecting the sensor resolution (figure 3), that defines "cells" on which different characteristics (attributes) are determined: density (number of 3D points in a cell compared with a nominal density defined by the discretization rates), mean altitude, variance on the altitude, mean normal vector and corresponding variances...

A non-parametric Bayesian classification procedure is used to label each cell: a learning phase, based on prototypes classified by a human, leads to the determination of probability density functions, and the classical Bayesian approach is applied, which provides an estimate of the probability for each possible label. A decision function that prefers false alarms (*i.e.* labelling a flat area as obstacle or uneven) instead of the non-detections (*i.e.* the opposite: labelling an obstacle as a flat area) is used (figure 4). A simpler but faster technique based on thresholds on the cell attributes has also been implemented.

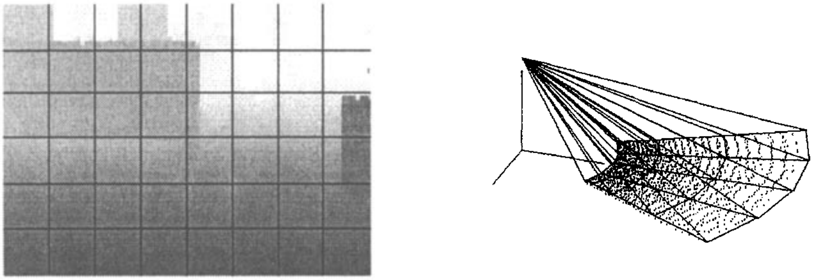


Figure 3: *Discretization used for the classification procedure: a regular discretisation in the sensor frame (left: a 3D image is represented as a video image, where the gray levels corresponds to point depth) defines a discretization of the perceived zone that respects the sensor resolution (right)*

This technique proved its efficiency and robustness on several hundreds of 3D images. Its main interest is that it provides an estimate of the confidence of its results: this information is given by the *entropy* of a cell. Moreover, a statistical analysis of the cell labelling confidence as a function of its label and distance to the sensor (directly related to the measure uncertainty) defines a predictive model of the classification process.

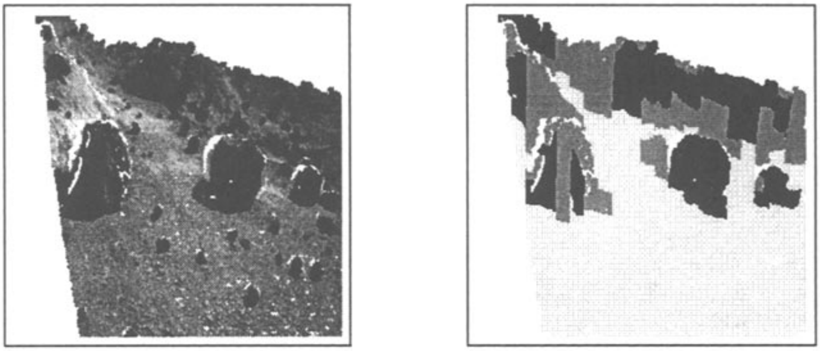


Figure 4: *Classification of a correlated stereo image: correlated pixels (top) and reprojection of the result in the camera frame (bottom - from clear to dark: unknown, flat, uneven and obstacle)*

3.1.2 Incremental fusion

The partial probabilities of a cell to belong to a terrain class and the variance on its elevation allow to perform a fusion procedure of several classified images. The fusion procedure is performed on a bitmap, in the pixels of which are encoded cell attributes determined by the classification procedure (label, label confidence, elevation and variance on the elevation).

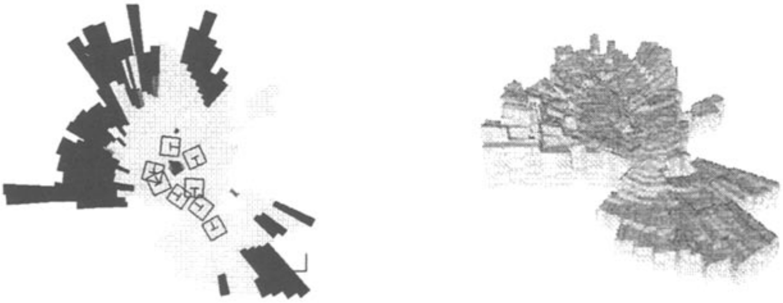


Figure 5: *Fusion of 8 different classified laser images: terrain classes (top) and elevation (bottom)*

3.1.3 Model structure and management

For the purpose of navigation planning, the global bitmap model is structured into a region map, that defines a connection graph. Planning a *path* (as opposed to planning a *trajectory*) does not require a precise evaluation of the static and kinematic constraints on the robot: we simply consider a robot point model, and therefore perform an obstacle growing in the bitmap before segmenting it into regions. The regions define a connection graph, whose nodes are on their borders, and whose arcs correspond to a region crossing (figure 6).

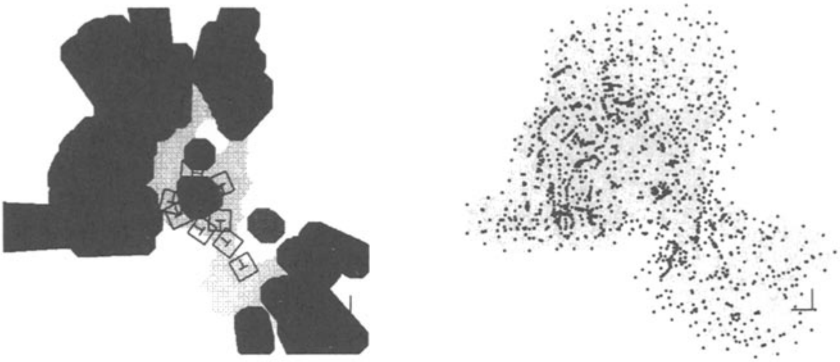


Figure 6: *The model of figure 5 after obstacle growing (top) and the nodes defined by the region segmentation (bottom)*

In order to satisfy memory constraints, the global model is explicited as a bitmap only in the surroundings of the robot's current position, and the region model (much more compact) is kept in memory during the whole mission (figure 7).

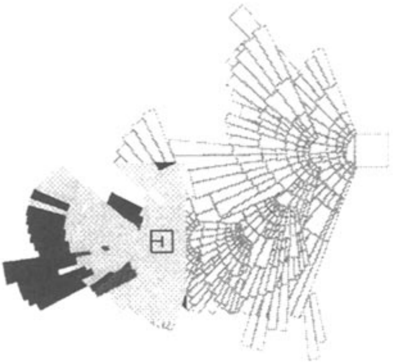


Figure 7: *Robot surroundings explicited as a bitmap*

3.2 Object-based representations

In order to be able to localize itself with respect to environment features, it is necessary for the robot to build representations of objects that it could recognize and use as landmarks.

When the terrain has been classified as flat but scattered with obstacles, we extract by a segmentation process the objects that are lying on the ground. Some of these objects that are salient and easily recognisable (e.g., peak-like) can be considered as landmarks for localisation.

To each object corresponds a referential - within which its surface is to be represented - this referential being itself located in a global coordinate frame in which it is represented by an uncertain state vector and a variance-covariance matrix. During navigation, the robot selects some objects possessing remarkable properties (visibility, selectivity, accuracy) that make them easily recognisable, and uses them as landmarks for anchoring the environment model and to locate itself. Landmarks will actually be specific features on such objects. As the robot moves the updating of the model is based on an extended Kalman filter. To recognize objects and landmarks, we use the two following complementary criteria:

- Comparison of the *global shape and features* of the objects (i.e., their intrinsic parameters).
- Comparison of the *relative position* of the objects themselves in order to obtain a *consistent* set of matchings. This is more interesting for landmarks because we could extract *precise* features on them.

A general approach will consist in exploiting the two criteria, but in our case, objects (mainly rocks) have similar shapes, and the position criterion will be proeminent.

The feature that defines the landmark is the object top, if it is salient enough. Landmarks here are local and should be *lower* that the robot sensor so that we can guarantee that the topmost point is perceived without ambiguity.

In a segmented 3D image, an object is selected as candidate landmark if:

1. It is not occluded by another object or by the image contour. If this object is occluded, it will be both difficult to recognize and to have a good estimate of its top, which may be not in the visible part.
2. Its topmost point is accurate. This is function of sensor noise, resolution and object top shape.
3. It must be in "ground contact". This is not an essential condition, but it reduces the recognition ambiguity.

In the model, an object is a landmark if it has been classified as landmark in at least one of its perceptions.

3.2.1 An Example of Landmark Selection

The top part of Figure 8 represents ten objects segmented in a 3D image. Six objects are selected as landmarks by the robot (objects number 1 to 6). Object 7 has not been selected because it is occluded by the image contour. This is also the case for object 9 (the hill) which top is not enough precisely defined. Object 8 is not occluded but it is not precise. Finally, object 10, which is on the hill, has not been selected because it is not in contact with the ground.

The bottom part of Figure 8 shows the environment map with the six selected landmarks and their corresponding uncertainties (ellipsoid projections corresponding to Gaussian distributions drawn at 99%). We can easily notice that precision decreases when the distance increases.

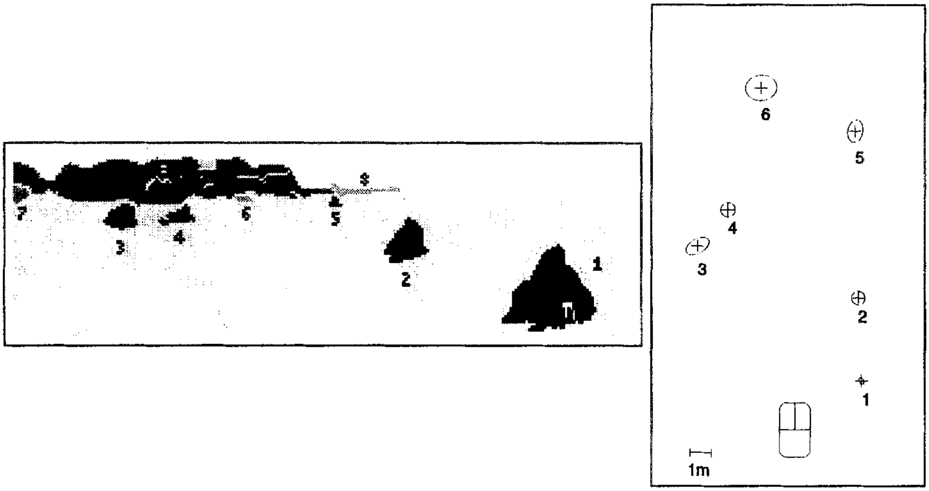


Figure 8: *Segmented objects (left) and selected landmarks with their uncertainty according to sensor noise and resolution and object shape (right)*

4 Planning actions

4.1 Navigation planning

Each time 3D data are acquired, classified and fused in the global model, the robot has to answer the following questions:

- Where to go? (sub-goal selection)
- How to go there? (motion mode selection)
- Where to perceive? (data acquisition control)
- What to do with the acquired data? (perception task selection)

For that purpose, the navigation planner reasons on the robot capabilities (action models for perception and motion tasks) and the global terrain representation.

4.1.1 Planning motion versus planning perception

A straightforward fact is that motion and perception tasks are strongly interdependent: planning and executing a motion requires to have formerly modelled the environment, and to acquire some specific data, a motion is often necessary to reach the adequate observation position.

Planning motion tasks in an environment modelled as a connection graph is quite straightforward: it consists in finding paths in the graph that minimise some criteria (time and energy, that are respectively related to the terrain classes and elevation variations). This is easily solved by classical graph search techniques, using cost functions that express these criteria.

Planning perception tasks is a much more difficult issue: one must be able to predict the results of such tasks (which requires a model of the perception processes), and the *utility* of these results to the mission:

- Localization processes can be modelled by a simple function that expresses the gain on the robot position accuracy, as a function of the number and distances of perceivable landmarks - assuming all landmarks are intrinsically equally informative;

- With the confidence model of the 3D data classification process, one can predict the amount of information a classification task can provide. But it is much more difficult to express the utility of a classification task to reach the goal: the model of the classification task cannot predict *what* will be effectively perceived. It is then difficult to estimate the interest of these tasks (figure 9).

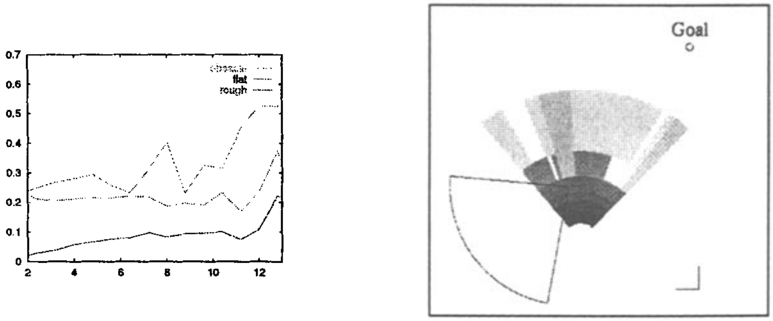


Figure 9: *The confidence model of the classification procedure (left) and the labelling confidence of the terrain model (represented as grey levels in the bottom image) allow to determine the classification task that maximises the gain in confidence from a given view point. But the result of this task is of a poor interest to reach the goal*

4.1.2 Approach

A direct and brute force approach to answer the former questions would be to perform a search in the connection graph, in which *all* the possible perception tasks would be predicted and evaluated at *each* node encountered during the search. Besides its drastic algorithmic complexity, this approach appeared unrealistic because of the difficulty to express the utility of a predicted classification task to reach the goal.

We therefore choose a different approach to tackle the problem: the perception task selection is *subordinated* to the motion task. A search algorithm provides an *optimal* path, that is analyzed afterwards to deduce the perceptions tasks to perform along it. The "optimality" criterion takes here a crucial importance: it is a linear combination of time and energy consumed, weighted by the terrain class to cross and the confidence of the terrain labelling (figure 10).

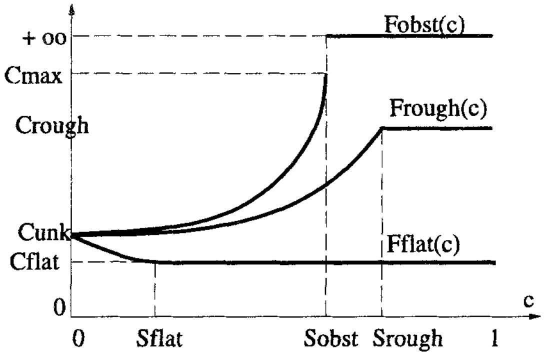


Figure 10: *Weighting functions of an arc cost, as a function of the arc label and confidence*

Introducing the labelling confidence in the crossing cost of an arc comes to consider *implicitly* the modelling capabilities of the robot: tolerating to cross obstacle areas labelled with a low confidence means that the robot is able to acquire easily informations on this area. Off course, the returned path is not executed directly, it is analysed according the following procedure:

1. The sub-goal to reach is the last node of the path that lies in a crossable area;

2. The labels of the regions crossed to reach this sub-goal determine the motion modes to apply;

3. And finally the rest of the path that reaches the global goal determines the aiming angle of the sensor.

Controlling localization: the introduction of the robot position uncertainty in the cost function allows to plan localization tasks along the path. The cost to minimise is the integral of the robot position accuracy as a function of the cost expressed in terms of time and energy (figure 11)

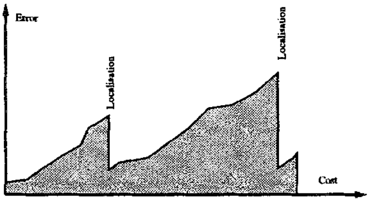


Figure 11: *Surface to minimise to control localisation tasks*

4.2 Trajectory planning

Depending on the label of the regions produced by the navigation planner, the adequate trajectory planner (2D or 3D) is selected to compute the actual trajectory within these regions.

4.2.1 Flat Terrain

The trajectory is searched with a simplified and fast method, based on bitmap and potential fields techniques. In a natural environment, and given the uncertainties of motion, perception and modelling, we consider it sufficient to approximate the robot by a circle and its configuration space is hence two dimensional, corresponding to the robot's position in the horizontal plane. Path planning is done according the following procedure :

- a binary bitmap *free/obstacle* is first extracted from the global bitmap model over the region to be crossed;
- a classical wavefront expansion algorithm then produces a distance map from which the skeleton of the free-space is computed (figure 12);
- the path reaching the sub-goal is obtained by propagating a potential through this skeleton. This path is finally transformed into a sequence of line segments and rotations (figure 12).

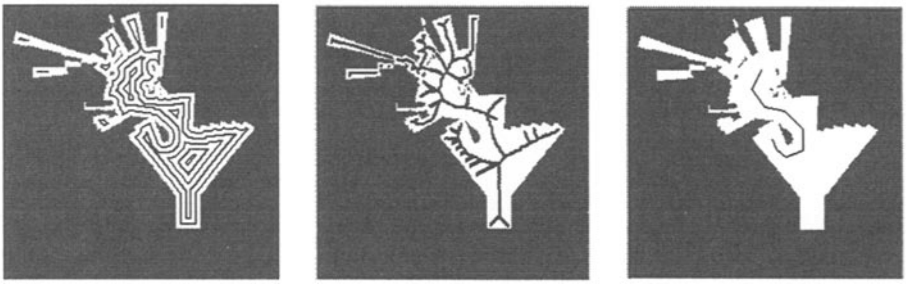


Figure 12: *The 2D planner: distance to the obstacles (left), skeleton of the free space (center), and a trajectory produced by the planner (right)*

Search time only depends on the bitmap discretization, and not on the complexity of the environment. The final trajectory is obtained within less than 2 seconds (on a Sparc 10) for a 256 × 256 bitmap.

4.2.2 Uneven Terrain

On uneven terrain, irregularities are important enough and the binary partition into *free/obstacle* areas is not anymore sufficient: the notion of obstacle clearly depends on the capacity of the locomotion system to overcome terrain irregularities and also on specific constraints acting on the placement of the robot over the terrain. The trajectory planner therefore requires a 3D description of the terrain, based on the elevation map, and a precise model of the robot geometry in order to produce collision-free trajectories that also guarantee vehicle stability and take into account its kinematic constraints.

This planner, described in [7], computes a motion verifying such constraints by exploring a three dimensional configuration space $CS = (x, y, \theta)$ (the x-y position of the robot frame and its heading θ). The obstacles are defined in CS as the set of configurations which do not verify some of the constraints imposed to the placement of the robot (figure 13). The ADAM robot is modelled by a rigid body and six wheels linked to the chassis by passive suspensions. For a given configuration, its placement results from the interaction between the wheels and the terrain, and from the balance of the suspensions. The remaining parameters of the placement vector (the z coordinate, the roll and pitch angles ϕ, ψ), are obtained by minimizing an energy function.

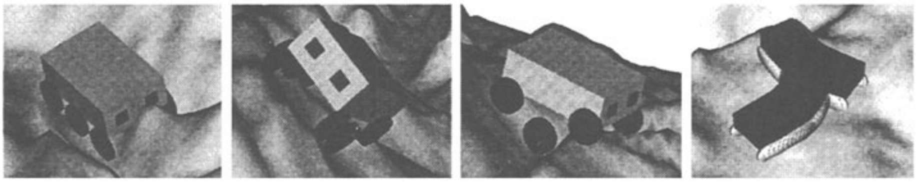


Figure 13: *The constraints considered by the 3D planner. From left to right : collision, stability, terrain irregularities and kinematic constraint*

The planner builds incrementally a graph of discrete configurations that can be reached from the initial position by applying sequences of discrete controls during a short time interval. Typical controls consist in driving forward or backwards with a null or a maximal angular velocity. Each arc of the graph corresponds to a trajectory portion computed for a given control. Only the arcs verifying the placement constraints mentionned above are considered during the search. In order to limit the size of the graph, the configuration space is initially decomposed into an array of small cuboid cells. This array is used during the search to keep track of small CS-regions which have already been crossed by some trajectory. The configurations generated into a visited cell are discarded and therefore, one node is at most generated in each cell.

In the case of incremental exploration of the environment, an additional constraint must be considered: the existence of unknown areas on the terrain elevation map. Indeed, any terrain irregularity may hide part of the ground. When it is possible (this caution constraint can be more or less relaxed), the path must avoid such unknown areas. If not, it must search the best way through unknown areas, and provide the best perception point of view on the way to the goal. The avoidance of such areas is obtained by an adapted weight of the arc cost and also by computing for the heuristic guidance of the search, a potential bitmap which includes the difficulty of the terrain and the proportion of unknown areas around the terrain patches [6].

The minimum-cost trajectory returned by the planner realizes a compromise

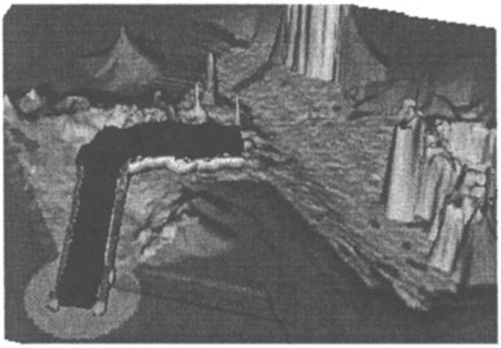


Figure 14: *A 3D trajectory planned on a real elevation map*

between the distance crossed by the vehicle, the security along the path and a small number of maneuvers. Search time strongly depends on the difficulty of the terrain. The whole procedure takes between 40 seconds to a few minutes, on an Indigo R4000 Silicon Graphics workstation. Figure 14 shows a trajectory computed on a real terrain, where darker areas correspond to interpolated unknown terrain.

5 Navigation Results

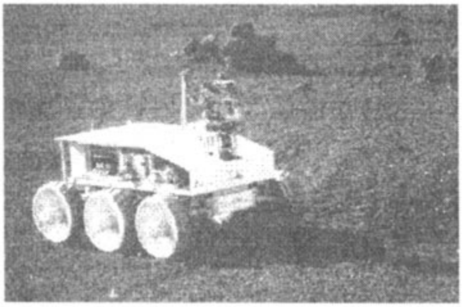


Figure 15: *ADAM in the Geroms test site*

The terrain modelling procedures and navigation planning algorithm have been intensively tested with the mobile robot Adam[1]. We performed experiments on the Geroms test site in the French space agency CNES, where Adam achieved several ``Go To [goal]'' missions, travelling over 80 meters, avoiding obstacles and getting out of dead-ends (for more details concerning Adam and the experimental setup, refer to [2]).

[1] ADAM is property of Framatome and Matra Marconi Space currently lent to LAAS

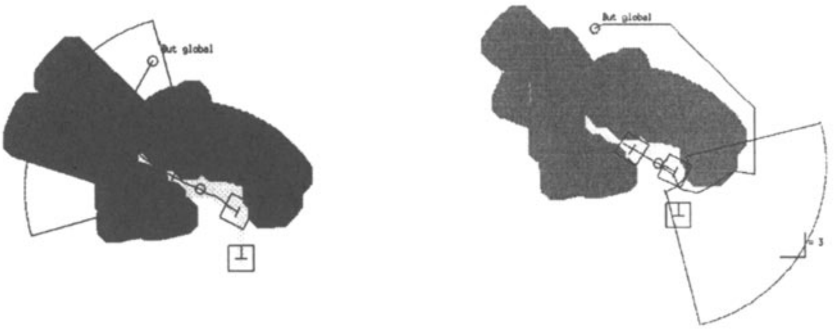


Figure 16: *The navigation planner explores a dead-end: it first tries to go through the bottom of the dead-end, which is modelled as an obstacle region, but with a low confidence level (top); after having perceived this region and confirmed that is must be labelled as obstacle, the planner decides to go back (bottom)*

Figure 16 presents two typical behaviours of the navigation algorithm in a dead-end, and figure 17 shows the trajectory followed by the robot to avoid this dead-end, on the terrain model built after 10 data acquisitions.

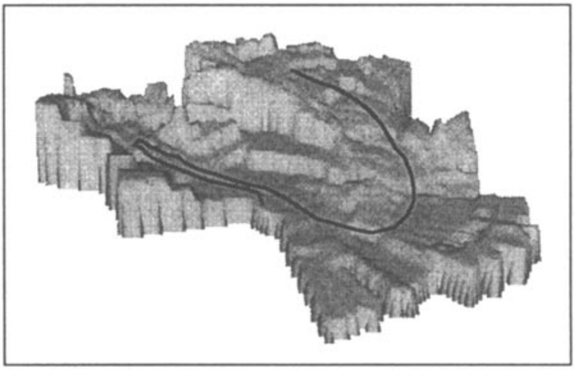


Figure 17: *A trajectory that avoids a dead-end (80 meters - 10 perceptions)*

The navigation planner proved its efficiency on most of our experiments. The adaptation of the perception and motion tasks to the terrain and the situation enabled the robot to achieve its navigation task efficiently. By possessing several representations and planning functions, the robot was able to take the adequate decisions. However, some problems raised when the planned classification task did not bring any new information: this happened in some very particular cases where the laser range finder could not return any measure, because of a very small incidence angle with the terrain. In these cases, the terrain model is not modified by the new perception, and the navigation planner re-planned the same perception task. This shows clearly the need for an explicit sensor model to plan a relevant perception task. And this generalizes to all the actions of the robot: the robot control system should possess a model of the motion or perception actions in order to select them adequately.

References

[1] S. Betge-Brezetz, R. Chatila, and M.Devy. Natural scene understanding for mobile robot navigation. In *IEEE International Conference on Robotics and Automation, San Diego, California*, 1994.

[2] R. Chatila, S. Fleury, M. Herrb, S. Lacroix, and C. Proust. Autonomous navigation in natural environment. In *Third International Symposium on Experimental Robotics, Kyoto, Japan, Oct. 28-30*, 1993.

[3] P. Fillatreau, M. Devy, and R. Prajoux. Modelling of unstructured terrain and feature extraction using b-spline surface. In *International Conference on Advanced Robotics, Tokyo(Japan)*, July 1993.

[4] E. Krotkov, M. Hebert, M. Buffa, F. Cozman, and L. Robert. Stereo friving and position estimation for autonomous planetary rovers. In *IARP 2nd Workshop on Robotics in Space, Montreal, Canada*, 1994.

[5] S. Lacroix, R. Chatila, S. Fleury, M. Herrb, and T. Simeon. Autonomous navigation in outdoor environment : Adaptative approach and experiment. In *IEEE International Conference on Robotics and Automation, San Diego, California*, 1994.

[6] F. Nashashibi, P. Fillatreau, B. Dacre-Wright, and T. Simeon. 3d autonomous navigation in a natural environment. In *IEEE International Conference on Robotics and Automation, San Diego, California*, 1994.

[7] T. Siméon and B. Dacre-Wright. A practical motion planner for all-terrain mobile robots. In *IEEE International Conference on Intelligent Robots and Systems, Yokohama (Japan)*, 1995.

[8] C. Thorpe, M. Hebert, T. Kanade, and S. Shafer. Toward autonomous driving : the cmu navlab. part i : Perception. *IEEE Expert*, 6(4), August 1991.

[9] C.R. Weisbin, M. Montenerlo, and W. Whittaker. Evolving directions in nasa's planetary rover requirements end technology. In *Missions, Technologies and Design of Planetary Mobile Vehicules. Centre National d'Etudes Spatiales, France*, Sept 1992.

[10] B. Wilcox and D. Gennery. A mars rover for the 1990's. *Journal of the British Interplanetary Society*, 40:484–488, 1987.

Acknowledgments. Many persons participated in the development of the concepts, algorithms, systems, robots, and experiments presented in this paper: R. Alami, G. Bauzil, S. Betgé-Brezetz, B. Dacre-wright, B. Degallaix, P. Fillatreau, S. Fleury, G. Giralt, M. Herrb, F. Ingrand, M. Khatib, C. Lemaire, P. Moutarlier, F. Nashashibi, C. Proust, G. Vialaret.

Active Vision for Autonomous Systems

Helder J. Araújo, J. Dias, J. Batista, P. Peixoto
Institute of Systems and Robotics–Dept. of Electrical Engineering
University of Coimbra
3030 Coimbra–Portugal
{helder, jorge, batista, peixoto}@isr.uc.pt

Abstract: In this paper we discuss the use of *active vision* for the development of autonomous systems. Active vision systems are essentially based on biological motivations. Two systems with potential application to surveillance are described. Both systems behave as "watchrobots". One of them involves the integration of an active vision system in a mobile platform. The second system can track non-rigid objects in real-time by using differential flow.

1. Introduction

A number of recent research results in computer vision and robotics suggest that image understanding should also include the process of selective acquisition of data in space and time [1, 2, 3]. In contrast the classical theory of computer vision is based on a reconstruction process, leading to the creation of representations at increasingly high levels of abstraction [4]. Since vision interacts with the environment such formalization requires modelling of all aspects of reality. Such modelling is very difficult, and therefore, only simple problems can be solved within the framework of classical vision theory. In active vision systems only the information required to achieve a specific task or behavior is recovered. By extracting only task-specific information and avoiding 3D reconstructions (by tightly coupling perception and action) these systems are able to operate in realistic conditions.

Autonomy requires the ability of adjusting to changes in the environment. Systems operating in different environments should not use the same vision and motor control algorithms. The structure and algorithms should be designed taking into account the purpose/goal of the system/agent. Since different agents, working with different purposes in different environments, do not sense and act in the same manner, we should not seek a general methodology for designing autonomous systems.

The development of autonomous systems by avoiding general purpose solutions, has two main advantages: it enables a more effective implementation of the system in a real environment (in terms of its performance) while at the same time decreasing the the computational burden of the algorithms. A strong motivation for this approach are the biological organisms [5]. In nature there are no general perception systems. We can not consider the Human visual system as general. As a proof of this fact are the illusions to which it is

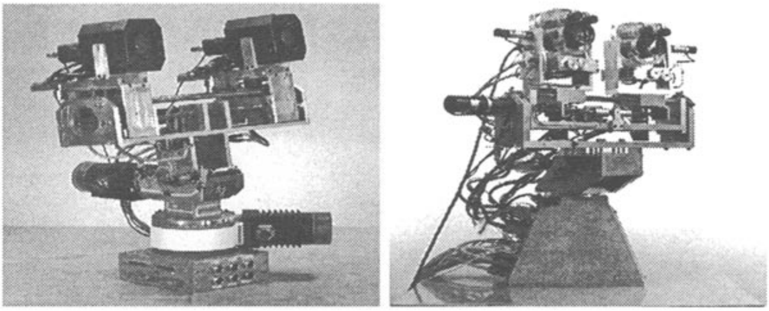


Figure 1. a)Active vision system used on the mobile robot; b)Non-mobile active vision system

subject and the visual tasks it can not perform, while other animals can [4]. Therefore the development of an autonomous system using vision as its main sensing modality should be guided by the tasks the system has to perform, taking into account the environment. From this analysis the behaviors required to implement the tasks should be identified and, as a result, the corresponding motor actions and the relevant visual information.

To demonstrate these concepts we chose to implement autonomous systems for surveillance applications. Two different systems addressing different tasks and problems in surveillance applications were designed and built.

2. Active Vision Systems for Surveillance

Surveillance is one important field for robotics and computer vision applications. The scenarios of surveillance applications are also extremely varied [6, 7, 8, 9]. Some applications are related to traffic monitoring and surveillance [8, 10], others are related to surveillance in large regions for human activity [11], and there are also applications (related to security) that may imply behavior modelling and analysis [12, 13, 14]. For security applications in man-made environments video images are the most important type of data. Currently most commercial systems are not automated, and require human attention to interpret the data. Images of the environment are acquired either with static cameras with wide-angle lenses (to cover all the space), or with cameras mounted on pan and tilt devices (so that all the space is covered by using good resolution images). Computer vision systems described in the literature are also based either on images from wide-angle static cameras, or on images acquired by active cameras. Wide-angle images have the advantage that each single image is usually enough to cover all the environment. Therefore any potential intrusion is more easily detected since no scanning is required. Systems based on active cameras usually employ longer focal length cameras and therefore provide better resolution images. Some of the systems are active and binocular [15]. These enable the recovery of 3D trajectories by tracking stereoscopically. Proprioceptive data from camera platform can be used to recover depth by triangulation. Trajectories in 3D can also be recovered monocularly by imposing the scene constraint that motion occurs in a plane, typically the ground plane [16]. One of the advantages of an active system is that, in general, the tracked target is kept in the fovea. This implies a higher resolution image and a simpler geometry. Within the framework of security applications we implemented two active and autonomous systems that perform different but complementary tasks: one of them pursues the intruder keeping distance and orientation approximately constant (a kind of a "mobile watchrobot"), while the other detects and tracks the intruder reconstructing its 3D trajectory (a "fixed watchrobot"). The first of these systems is based on a mobile robot fitted with a binocular active vision system while the latter is based only on a binocular active vision system (see Figure 1). The vision processing and the design principles used on both are completely different, for they address different tasks. Since the first one has to keep distance and orientation relative to the target approximately constant it has to *translate*. In this case all vision processing is based on correlation (it correlates target templates that are updated periodically to compensate for shape changes). The second system does not translate and in this case almost all the visual processing is based on differential optic flow. With this approach it is easier to cope with changes of the target shape. We will now describe in detail both systems.

3. The "Mobile Watchrobot"

The pursuit of moving objects with machines such as a mobile robot equipped with an active vision system deals with the problem of integration and cooperation between different systems. This integration has two distinct aspects: the interaction and cooperation between different control systems and the use of a common feedback information provided by the vision system. The system is controlled to keep constant the distance and the orientation of the robot and the vision system. The solution for this problem deals implies the interaction of different control systems using visual feedback while performing real-time tracking of objects by using a vision system. This problem has been addressed in different fields such as surveillance, automated guidance systems and robotics in general. Several works addressed the problems of visual servoing but they are mainly concerned with object tracking by using vision and manipulators [17, 18, 19] and only some address problems related with ours [20, 3, 21]. Papanikolopoulos also proposed a tracking process by using a camera mounted on a manipulator for tracking objects with a trajectory parallel to the image plane [19]. A control process is also reported by Allen for tracking moving objects in 3D [17]. These studies have connection with the solution for pursuit proposed in this article, since they deal with the tracking problem by using visual information. However in our system we explore the concept of visual fixation to develop the application. The computational solution for visual fixation uses motion detection to initiate the *fixation process* and to define a pattern that will be tracked. During *pursuit* the system uses image correlation to continuously track the target in the images [22]. More recently several laboratories have been engaged in a large European project (the *Vision as Process* project) for the development of systems, based on active vision principles [21]. Some of the systems described above have similarities with ours but in our system we control the system to keep the distance and orientation of the mobile robot with respect to a *target*. The solution used includes the control of the *gaze* of the active vision system. Furthermore, our hierarchical control scheme establishes a pursuit process using different degrees of freedom on the active vision system and the movement of the mobile robot. To simplify the solution several assumptions were made. These assumptions are based on the type of movements and targets that we designed the system to cope with and the system's physical constraints such as: maximum robot velocity, possibility of adjustment of the optical parameters for focusing, maximum computational power for image processing and, the non-holonomic structure of the mobile robot. We assume that the

- target and the robot move on a plane (horizontal plane);
- the difference between the velocities of the target and of the robot does not exceed $1.2m/s$;
- the distance between the target and the mobile robot will be in the interval of $[2.5m, 5m]$ and the focal length of both lenses is set to $12.5mm.$;
- the target is detected only when it appears inside the cameras' field of view.
- the system is initialized by setting the vision system aligned with the vehicle (the cameras are oriented to see the vehicle's front).

These assumptions bound the problem and only two variables are used to control the system. One is the angle in the horizontal plane defined by the target position relative to the mobile robot referential. The other is the distance between the robot and the target.

3.1. Pursuit of Moving Objects

The problem of pursuing a moving object is essentially a motion matching problem. The robot must be controlled to reach the same motion as the *target*. In practice this is equivalent to keep constant the distance and orientation from the robot to the *target*. However, the solution for this problem has some particular aspects that must be emphasized. If the target is a person walking, its trajectory can be suddenly modified and consequently its velocity. Any solution proposed must cope with these situations and perform the control of the system in $real - time$. Since the machines have physical limitations in their velocity and maneuvering capabilities, it is essential to classify the different sub-systems used according to their velocity characteristics. In our experiments we use a mobile robot and an active vision system, and these two systems have different movement characteristics. The active vision system presents greater velocity than the mobile robot and also has less mass. However, it is the mobile robot (the body of the system) that must follow the *target* - see figure 2.

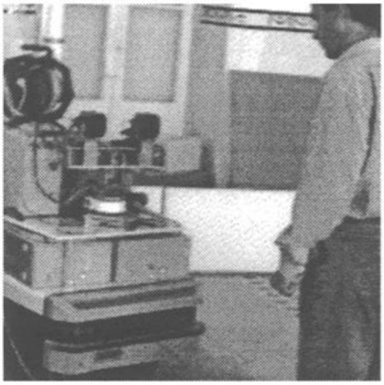


Figure 2. The information provided by the *active vision system* is used to control the mobile robot to *pursuit* a person in *real – time.*

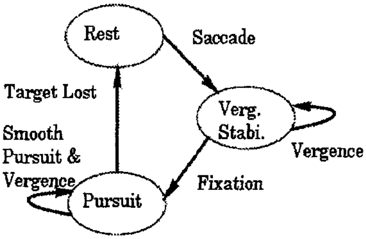


Figure 3. State diagram of the *pursuit* process.

To perform the pursuit of a moving *target* we use two basic control schemes: a visual *fixation* control of the active vision system and the trajectory control of the robot. The visual *fixation* control guarantees that the *target* is continuously tracked by the vision system, and gives information about its position to the robot control. The robot control uses that information as a feedback to maintain the distance and orientation to the *target*. The visual *fixation* control must be one visual process that runs in the active vision system and has capabilities to define a *target*, to concentrate the vision system on the *target* and follow it. A process with these characteristics has similarities with the visual gaze-shifting mechanism in the humans. The gaze-shifting mechanism generates movements in the vision system to put a new object of interest in the center of the image and hold it there. The movement used to put the object in the center is called *saccade*, it is fast and it is performed by the two eyes simultaneously. If the *target* of interest is moving relative to the world, the vision system must perform movements to hold the *target* in the image center. These movements are composed by two types of motions called *smooth pursuit* and *vergence* . These motions are the consequence of the control performed by the process that we designate as *fixation*. The *fixation* process centers and holds the orientation of the vision system on a point in the environment. *Fixation* gives a useful mechanism to maintain the relative orientation and translation between the referential in the vehicle and the *target* that is followed. This results from the advantages of the *fixation* process, where the selected *target* is always in the image center (foveal region in the mammals). This avoids the segmentation of all the image to select the *target* and allows the use of relative coordinate systems which simplifies the spatial description of the *target* (relationship between the observer reference system and the object reference system). The pursuit process can be described graphically by the state diagram in figure 3. The process has three states: *Rest*, *Vergence Stabilization*, and *Pursuit*. The *pursuit* process must be initialized before starting. During this initialization, a *target* is chosen and several movements are performed by the active vision system: the gaze is shifted by a *saccade* movement and the vergence stabilized. In our system the *target* is chosen based on the visual motion stimulus. The selection corresponds to a region in the images that generates a large visual motion in the two images. If a *target* is selected, a *saccade* movement is performed to put the *target* in the image center, and the system changes from the state *Rest* to *Vergence Stabilization*. During the *saccade* movement no visual information is used to feedback the movement. In the *Vergence Stabilization* state the system adjusts its *fixation* in the *target*. This is equivalent to establishing the correct correspondence between the centers of the two images, and defining a *fixation* point in the *target*. When the vergence is stabilized, the system is maintained in the *Pursuit* state.

3.2. Building a System to Simulate Pursuit

3.2.1. System Architecture

The main hardware components of the system are the mobile robot and the active vision system. These two basic units are interconnected by a computer designated *Master Processing Unit*. This unit controls the movements of the active vision system, communicates with the robot's on-board computer and is connected to two other computers designated *Processing Units*. These units are responsible for processing the images provided by the active vision system. The connections between different processing units are represented in the diagram shown in figure 4 and a photograph of the system is presented in figure 5. The *Right* and the *Left Slave Processing Units* are two PCs. Each contains a frame grabber connected to each one of the cameras. The *Slave Processing Units* process the images and communicate their results to the *Master Processing Unit* (another PC). These communications use a 10 MBits connection provided by Ethernet boards (one board on each computer). The active vision system has two CCD monochromatic video cameras with motorized lenses (allowing for the control of the iris, focus and zoom) and five step motors that confer an equal number of degrees of freedom to the system (vergence of each camera, baseline shifting, head tilt and neck pan). The *Master Processing Unit* is responsible for the control of the degrees of freedom of the active vision system (using step motor controllers) and for the communication with the mobile platform (using a serial link). The actual control of the mobile platform is done by a multiprocessor system, installed on the platform. The management and the interface

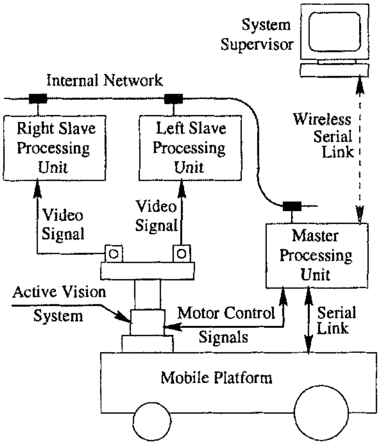


Figure 4. System Architecture.

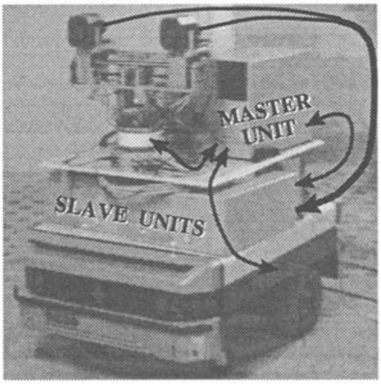


Figure 5. The active vision system and the mobile robot.

with the system is done by a computer, connected to the *Master Processing Unit* using the serial link and a wireless modem.

3.2.2. Camera Model

To find the relation between a 2D point in one image obtained by either camera with its corresponding 3D point in that camera's referential $\{\mathbf{CAM}\}$, we use the perspective model. The projection of the 3D point **P** in plane **I** is a point **p**$=(u, v)$, that results from the intersection of the projective line of **P** with the plane **I**. The perpendicular projection of the point **O** in the plane **I** is defined as the center of the image, with coordinates (u_o, v_o). The distance f between the point **O** and its projection is called the focal length. If (x, y, z) are the 3D coordinates of the point **P** in the $\{\mathbf{CAM}\}$ referential, the 2D coordinates of the projection (x_u, y_v) of it on a continuous image plane is given by the perspective relationships:

$$x_u = f\frac{x}{z} \quad y_v = f\frac{y}{z} \tag{1}$$

Since the image for processing is a sampled version of the continuous image, the relation between the units (millimeters) used in the $\{\mathbf{CAM}\}$ referential and the image points (u, v) are related with (x_u, y_v) by:

$$u = S_x x_u + u_0 \quad v = S_y y_v + v_0 \tag{2}$$

That relation is obtained with a calibration process that gives the scale factors for both the x and the $y - axis$ (S_x and S_y respectively) [23].The image center (u_o, v_o), the focal length f and the scale factors S_x and S_y are called the intrinsic parameters of the camera.

3.2.3. System Models and Geometric Relations

The information of the *target* position in the images is used to control the position and orientation of the vision system and of the mobile robot in order to maintain the relative distance and orientation to the *target*. Essentially the system must control the position of each actuator to maintain this goal. This implies to control the actuators of the vision system and also of the mobile robot. In the case of the vision system the actuators used are step motors. These motors are controlled by dedicated units supervised by the *Master Processing Unit*. These motors rotate a specific number of degrees for each pulse sent to their power driver unit. The pulses are generated by the dedicated control units. These units generate different profiles for the pulse rate curve which must be adjusted for each motor. This adjustment is equivalent to a step motor *identification* procedure. This procedure was performed for each motor used in the active vision system. With this procedure the correct curve profile was adapted for a precise position control. The mobile robot has also its own on-board computer that controls the motors used to move it. The on-board computer is responsible for the correct execution of the movements, and it accepts commands for movements that can be modified during their execution. This possibility is explored in our system to correct the path during the movement execution. The commands sent to the mobile robot reflect the position that the robot must reach to maintain the distance to the *target*. If the commands sent do not exceed the possibilities of the system, the command will be sent to the robot to be executed with accuracy. This detail is verified before sending a command to the mobile robot. The movements executed by the mobile robot are based on two direct current motors associated with each of the driving wheels (rear axle). The movements permitted with this type of configuration are represented in figure 6, and are used to make the compensation for the active vision system. The control of the motors is done by a multi-processing, installed on the mobile platform. Therefore, the only responsibility of the *Master Processing Unit* is to send to the platform the parameters of the required movement. The robot's movements represented in figure 6 can be divided into three groups: translational (no angular velocity),

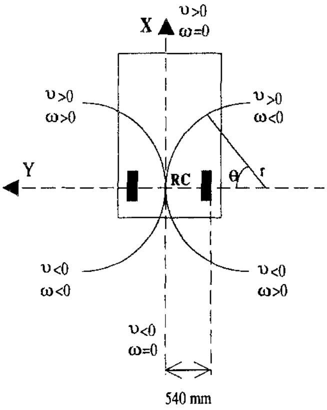


Figure 6. Possible movements of the mobile robot: ω is the angular velocity, v is the linear velocity, r is the radius and θ is the orientation angle.

rotation around the center of the driving axle represented in figure 6 by RC (no linear velocity) and compositions of both movements. To define each one of these three movements, it is necessary to supply not only the values for the linear and angular velocities (v, ω), but also the duration time of the movement (T).

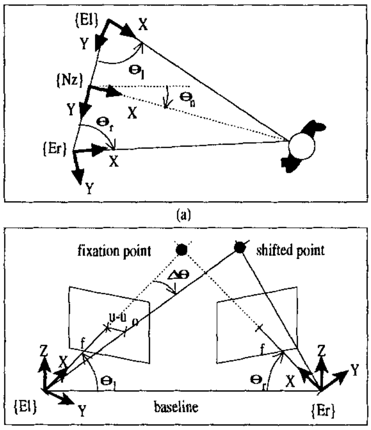
The following lines are examples of commands that can be issued to the platform to launch velocity controlled movements:

- issuing a composed movement: "MOTV LA V=100 W=100 T=50";
- issuing a pure rotation movement: "MOTV LA V=0 W=100 T=50";
- issuing a linear movement: "MOTV LA V=100 T=50".

Another type of movements are those based on the control of the platform position. In this case, the specified parameters define the distance that each one of the driving wheels must cover, and the time that they should take to do it (in $40ms$ units). The following example shows a command that gives rise to a rotation of the platform based on the controlled position movements:

- "MOVE P RC=-200,200 P=100".

Since the *target* changes its position in space, in most of the time its image position will also change. The goal is to control the system in such a way that the object's image projects into the center of both images, maintaining at the same time the distance to the object. The control can be performed by controlling the robot position, the neck orientation and the vergence of both cameras. The control implies the use of these degrees of freedom to reach the goal of *pursuing* a target. It is possible to obtain expressions relating the several degrees of freedom, useful for their control, based on the geometric relationships. The goal is to change the cameras' angles θ_l and θ_r, by the amount necessary to keep the projection of the *target* in the center of the image (see figure 7). Since we assume that the target moves on the same plane as the



(a)

(b)

Figure 7. Cameras' vergence angle control.

mobile robot we will consider only the horizontal disparity $\Delta u = (u - u_o)$. Let u be the coordinate in pixels of the reference point along the $x - axis$ of either frame. The angle that each camera must turn is given by:

$$\Delta\theta = \arctan \frac{u - u_o}{S_x f} \tag{3}$$

This relation is easily derived from the equations 2 and from the representation in figure 7. To provide the system with the ability to react to the movements of the object's and with the ability to keep the distance and attitude between the two bodies, it is necessary to evaluate the distance of the object with respect to the robot. The position of the object to track is defined in terms of its distance D and the angle θ_n with respect to the $\{\mathrm{C}\}$ referential, and using the *fixation* point as reference (both parameters are represented in figure 8). To obtain the equations that give the values of D and θ_n, we start by defining the following relations, taken directly from figure 9 (equivalent to figure 8, but with some auxiliary parameters):

$$\begin{aligned} h &= \tan(\theta_r) D_r \qquad & h &= \tan(\theta_l) D_l \\ B &= D_l + D_r \qquad & p &= D_l - \frac{B}{2} \end{aligned} \tag{4}$$

The distance D and the angle θ_n of the *fixation* point with respect to the $\{\mathrm{C}\}$ referential can be obtained by the following equations (recall that the angle θ_n is positive clockwise - see figure 9):

$$\theta_n = 90^\circ - \arctan\left(\frac{h}{p}\right) \qquad D = \sqrt{h^2 + p^2} \tag{5}$$

Note that, when θ_l equals θ_r, the above relations are not valid. In that case, the angle θ_n is zero, and the distance D is equal to:

$$D = \frac{B}{2} \tan(\theta_l)$$

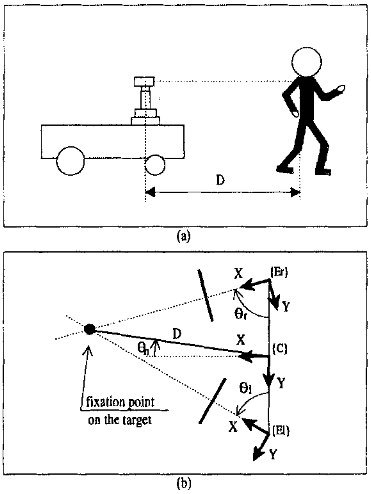


Figure 8. Distance and angle to the object defined in the plane parallel to the $xy - plane$ of the $\{C\}$ referential.

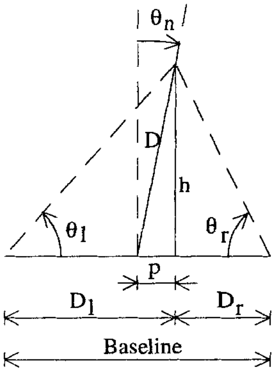


Figure 9. Auxiliary parameters.

As described above, the motion and feature detection algorithms generate the position in both images of the object to follow. From that position, only the value along the $x - axis$ will be used, since we assume that the object *moves in the horizontal plane*, and therefore *without significant vertical shifts.* The trajectories of the moving platform are planned by the *Master Processing Unit* based on the values of D and θ_n given by equations 5. The values D and θ_n define a 2D point in the $\{C\}$ referential. These values can be related to the $\{\mathbf{B}\}$ referential since all the relationships between referentials are known. The result is a point $\mathbf{P}$ with coordinates x and y as shown in figure 10. This figure is useful to establish the conditions for a mobile robot's trajectory when we want that the mobile robot reaches a point $\mathbf{P}(x, y)$. To clarify the situation we

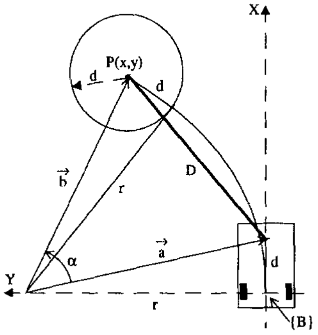


Figure 10. Robot trajectory planning.

suppose that the object appeared in vehicle's front with an initial orientation $\theta = 0$. (the solution is be similar for an angle $\theta < 90°$). We know that several trajectories are possible to reach a specific point but, the trajectories' parameters are chosen according to the following:

- The point **P** is assumed to be in front of the vehicle and the angle α is always greater than zero 10. This is a condition derived from the system initialization and the correct execution of the *pursuit* process (see Section I). Additionally, that condition helps to deal with the non-holonomic structure of the mobile robot.

- The platform must stop at a given distance from the object. This condition is represented in figure 10 by the circle around the point **P** (the center of the platform's driving axle, point **RC**, must stop somewhere over this circle).

- The platform must be facing the object at the end of the trajectory. In other words, the object must be at the $x - axis$ of the $\{\mathbf{B}\}$ referential when the platform stops.

The trajectory that results from the application of those two conditions is a combination of a translational and a rotational movement. Two parameters are needed to define the trajectory, as shown in figure 10: the radius r and the angle α. The analysis of the figure allows the derivation of the following relations:

$$\begin{aligned}
\vec{\mathbf{b}} &= \begin{bmatrix} x & (y-r) \end{bmatrix} \\
\vec{\mathbf{a}} &= \begin{bmatrix} d & -r \end{bmatrix} \\
\|\vec{\mathbf{b}}\| &= \|\vec{\mathbf{a}}\| \\
r^2 + d^2 &= x^2 + (y-r)^2 \\
\vec{\mathbf{b}}\,\vec{\mathbf{a}} &= \left(\sqrt{x^2+(y-r)^2}\right)\left(\sqrt{r^2+d^2}\right)\cos(\alpha) \\
&= xd + r(r-y) \qquad (6)
\end{aligned}$$

After simplification we get:

$$r = \frac{x^2 + y^2 - d^2}{2y} \qquad \alpha = \arccos\left(\frac{xd + r(r - y)}{r^2 + d^2}\right) \tag{7}$$

The equations 7 are not defined when the y coordinate is equal to zero. In that case, the trajectory is linear, and the distance r that the mobile platform must cover is given by:

$$r = x - d \tag{8}$$

3.3. Vision Processing and State Estimation

3.3.1. Image Processing

The *Slave Processing Units* analyze, independently, the images captured by the frame grabbers connected to the cameras. Therefore, the motion and feature detection algorithms described here are intended to work with the sequence of images obtained by each camera. The *Slave Units* are responsible by processing the sequence of images during all states illustrated in figure 3. When the system is initialized, the *Rest* phase starts and the *Master Processing Unit* commands the *Slave Units* to begin a searching phase. This phase implies the detection of any movement that satisfies a set of constraints described below. At a certain point during this phase, and based on the evolution of the process in both *Slave Units*, the *Master Unit* decides if there is a *target* to follow. After this decision the *Master Unit* sends a *saccade* command to the *Slave Units* to begin the vergence stabilization phase. During this phase, the system will only follow a specific pattern corresponding to the *target* previously defined and ignoring any other movements that may appear. This phase proceeds until the *vergence* is considered stable and after that it changes to the *pursuit* state. The system remains in this state until the pattern can no longer be found in the images.

3.3.2. Gaussian Pyramid

In order to speedup the computing process, the algorithms are based on the construction of a Gaussian pyramid [24]. The images are captured with 512x512 pixels but are reduced by using this technique. Generally speaking, using a pyramid allows us to work with smaller images without missing significant information. Climbing one level on the pyramid results in an image with half the dimensions and one quarter of the size. Level 0 corresponds to 512x512 pixels and level 2 to 128x128. Each pixel in one level is obtained by applying a mask to the group of pixels of the image directly below it. The applied mask is basically a low pass filter, that helps in reducing the noise and smoothing the images.

3.3.3. Image Processing for Saccade

The *saccade* is preceded by searching for a large movement in the images. As described above, the searching phase is concerned with the detection of any type of movements, within certain limits. For that purpose, two consecutive images $t(k)$ and $t(k+1)$ separated by a few milliseconds are captured. These images are analyzed at the pyramid level 2. The analysis consists in two steps described graphically by the blocks diagram shown in figure 11:

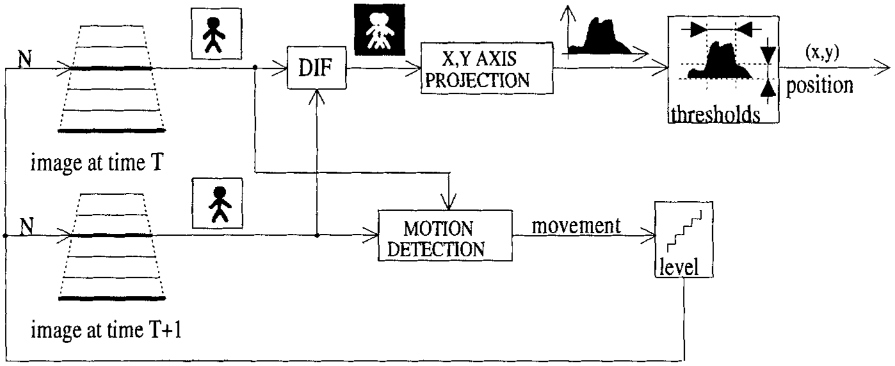


Figure 11. Illustration of the image processing used for saccade. The *saccade* is preceded by searching for a large movement in the images. The searching phase is concerned with the detection of image movements, within certain limits.

- Computation of the area of motion using the images acquired at time $t(k)$ and $t(k+1)$. This calculation measures the amount of shift that occurred from one frame to the other, and is used to decide when to climb or to descend levels on the pyramid.

- Absolute value subtraction of both images, pixel by pixel, generating an image of differences, followed by the computation of the projections of the image in the x and $y-axis$. Since we assume that the target will have a negligible vertical motion component, only the image projection on the horizontal axis is considered for the *saccade* movement. Two thresholds are then imposed: one defining the lowest value of the projection that can be considered to be a movement, and the other limiting the minimum size of the image shift that will be assumed as a valid moving *target*. If both thresholds are validated, the object is assumed to be in the center of the moving area. If the movement is sensed by both cameras and it satisfies these two thresholds, a *saccade* movement will be generated.

3.3.4. Image Processing for Fixation

The goal of *fixation* is to keep the *target* image steady and centered. This presumes that the *target* is the same for the two cameras. Vergence is dependent on this assumption and, in this work, it is assumed that the vergence is driven by the position of the three-dimensional *fixation* point. This point corresponds to the three-dimensional position of the *target* that must be followed. This is equivalent to the problem of finding the correspondence between *target* zones in the two images. In this work this process is called correspondence. Since the system is continuously controlled to keep the images centered on the fixated *target*, the correspondence zone is defined around the image center and the search process becomes easy. The correspondence used for *fixation* starts by receiving the pattern from the other *Slave Unit*. The pattern that is needed to follow in one image (left/right) is passed to the other (right/left) to find

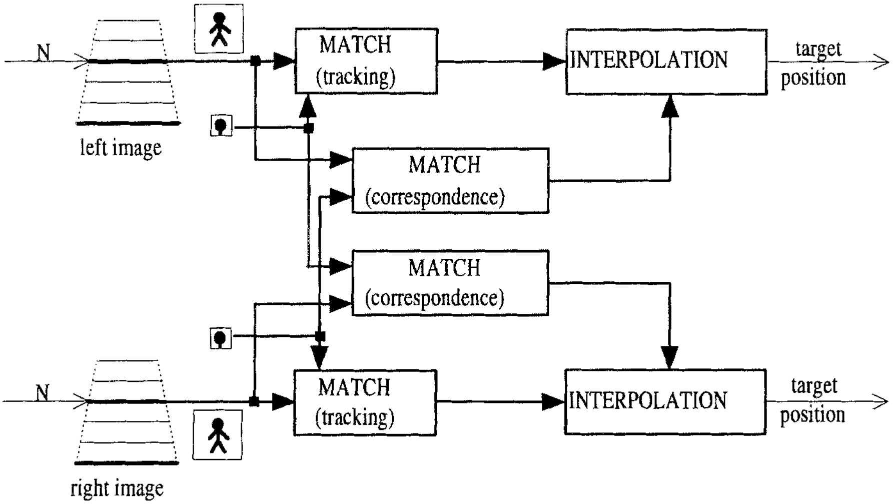


Figure 12. Illustration of the image processing used for *Fixation*. To maintain the *fixation* dynamically, the process uses two phases: pattern tracking and correspondence phase. The pattern is defined as an area around the center of the image acquired after the execution of the *saccade* movement. The tracking phase locates the *target* and measures the variables necessary to control the system and keep the object centered in the image. The correspondence phase confirms the results of the searching phase and estimates the images disparity.

the position of the correspondent pattern. The search starts around the image center and tries to find an image that matches the pattern received. The test uses the operator described by equation (9) and the image area with lowest difference will be considered if the value of $\delta(x,y)$ is less than a maximum threshold. The similarity operator applied to the position (x,y) is defined as follows:

$$\delta(x,y) = \sum_{i=-n}^{n} \sum_{j=-m}^{m} (P(i,j) - I(x-i, y-j))^2 \tag{9}$$

where **P** is the pattern to search and **I** is the image. In the definition, $2n+1$ and $2m+1$ represent the width and height of the pattern in pixels. The process to keep the *fixation* dynamically is equivalent to a process of *gaze holding*. The *gaze holding* uses similar image processing as used for *fixation*. The *gaze holding* is implemented through *smooth pursuit* and it vergence movements of the active vision system. These movements are based on the tracking of a pattern in each image. The pattern is defined as an area around the center of the image acquired after the execution of the *saccade* movement. The *gaze holding* uses two image processing phases: the tracking of the patterns in the left and right images, and the correspondence of the left and right patterns. The tracking phase uses the pattern defined at the end of the *saccade* movement. That pattern will be searched in each image captured, at a certain pyramid level,

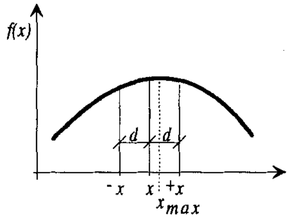


Figure 13. Polynomial representation.

as described by the blocks diagram shown in figure 11. The search consists in the computation of a similarity measure between the pattern and the areas with equivalent size taken from the image - see equation 9. The image area that resulted in the lowest difference is considered to be a match, but only if the value of the difference is less than a maximum threshold, above which no matches are accepted. This process is finished with the verification of correspondence between *left* and *right* patterns in similar manner as the *fixation* phase. Since the images used during *fixation* and *gaze holding* are sub-sampled, the target position is defined with more precision with by means of a quadratic sub-pixel interpolation. Since we are using images, the samples are discrete values of a function and we use interpolation to find the function's maximum with sub pixel precision. Suppose that the function is parameterized by the polynomial:

$$y = ax^2 + bx + c \tag{10}$$

with the extreme point at $x_{max} = -\frac{b}{2a}$. If the discrete points are at ${}^{-}x = x - d$ and ${}^{+}x = x + d$ with d the distance between samples, their values for the function $f(x)$ are given by:

$$\begin{aligned} f({}^{-}x) &= ax^2 - 2adx + ad^2 + bx - db + c \\ f({}^{+}x) &= ax^2 + 2adx + ad^2 + bx + db + c \\ f(x) &= ax^2 + bx + c \end{aligned} \tag{11}$$

Combining these equations we obtain the x_{max} position is given by:

$$x_{max} = x - \tfrac{d}{2} \tfrac{[f({}^{+}x)+f({}^{-}x)]}{[f({}^{-}x)-2f(x)+f({}^{+}x)]} \tag{12}$$

The final position is combined with the position obtained by the *fixation* and *gaze holding* phases. Actually the final position corresponds to the middle point between these two positions.

3.3.5. System State

The system state used for control is described by two variables: the distance D from the system to the *target* and the angle θ_n of the neck pan, defined when the vision system is fixated on the *target*. Both variables have values that can be obtained by the geometry of the system and are described by equations (5). These two quantities are continuously estimated by using a fixed coefficient filter: the (α-β-γ) tracker. Filtering and prediction methods can be used to estimate the present and the future of the *target* kinematics quantities such as position, velocity, and acceleration. In the current work we use the prediction capabilities of the filters to compensate for the delays in the system. There are two common approaches to filtering and prediction. The first is to use fixed coefficients (α-β and α-β-γ filters), and the second, Kalman filtering, which generates time-variable coefficients that are determined by an a priori model for the statistics of the measured noise and *target* dynamics. The first approach has computational advantages, but Kalman filtering performs a high-accuracy tracking. The filter used in this system is of the first type, and can be used in linear dynamic systems with time-invariant coefficients in their state transition matrix and measurement equations. For these systems the filter gain achieves steady-state values that can often be computed in advance. This advantage is important to save some computational time and in practice both approaches are valid in our system, since we do not expect sudden changes in our model's parameters. Both of these filters can be implemented recursively and the data received in the past is included in the present estimates. Therefore, all data is used but forgotten at an exponential rate. The estimate of the variable value at time k is $\hat{x}(k|k)$ and will be denoted by the smoothed estimate x_s. With these filters we can predict the values for the next step. The one-step prediction is denoted by $x_p = x(k+1|k)$, and signifies the estimate at time $k+1$ given data through time k. Fixed coefficients filters have the advantage of simple implementation using fixed parameters for the filter's gains. The most extensively applied of these filters is the α-β tracker. This filter is used with constant velocity models when only position measurements are available. The α-β tracker is defined by the following equations:

$$x_s(k) = \hat{x}(k|k) = x_p(k) + \alpha[x_0(k) - x_p(k)] \tag{13}$$

$$v_s(k) = \hat{\dot{x}}(k|k) = v_s(k-1) + \frac{\beta}{qT}[x_0(k) - x_p(k)] \tag{14}$$

$$x_p(k+1) = \hat{x}(k+1|k) = x_s(k) + Tv_s(k) \tag{15}$$

The variable T is the sampling interval, $x_0(k)$ is the measurement at time k and the α and β are the fixed filter gain coefficients. The quantity q is normally defined as one, but in the case where missing observations occur its value may be taken as the number of scan steps (interactions) since the last measurement. The initialization process can be defined by:

$$x_s(1) = x_p(2) = x_0(1) \qquad v_s(1) = 0 \qquad v_s(2) = \frac{[x_0(2) - x_0(1)]}{T}$$

The equation 13 is used directly when an observation is received at time k. The optimal values for α and β are derived in [25] and depend only on the ratio of the process noise standard deviation and the measurement noise standard deviation. The logical extension of the α-β filter is the α-β-γ filter, which includes an estimate for the acceleration and can be used with the assumption of uniform acceleration. This filter makes a quadratic prediction instead of a linear one, and tends to be more sensitive to noise but better able to predict smoothly varying velocities. The equations for this filter are defined as:

$$x_s(k) = \hat{x}(k|k) = x_p(k) + \alpha[x_0(k) - x_p(k)] \tag{16}$$

$$v_s(k) = \hat{\dot{x}}(k|k) = v_s(k-1) + Ta_s(k-1) + \frac{\beta}{qT}[x_0(k) - x_p(k)] \tag{17}$$

$$a_s(k) = \hat{\ddot{x}}(k|k) = a_s(k-1) + \frac{\gamma}{(qT)^2}[x_0(k) - x_p(k)] \tag{18}$$

$$x_p(k+1) = \hat{x}(k+1|k) = x_s(k) + Tv_s(k) + \frac{T^2}{2}a_s(k) \tag{19}$$

The usual initialization is:

$$x_s(1) = x_p(2) = x_0(1)$$

$$v_s(1) = a_s(1) = a_s(2) = 0$$

$$v_s(2) = \frac{[x_0(2) - x_0(1)]}{T}$$

$$a_3(3) = \frac{[x_0(3) - 2x_0(2) + x_0(1)]}{T^2}$$

The optimal values for α and β are defined as [25] and the optimal value for γ is given by:

$$\gamma = \frac{\beta^2}{\alpha} \tag{20}$$

In our case the values estimated by this filter are described by:

$$\hat{\mathbf{x}}_D = \begin{bmatrix} D \\ \dot{D} \\ \ddot{D} \end{bmatrix} \quad \mathit{and} \quad \hat{\mathbf{x}}_\theta = \begin{bmatrix} \theta \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} \tag{21}$$

As it was already said this filter is specifically useful for estimating the state variables assuming that the *target* has uniform accelerations. This enables the compensation for delays in the system as will be described below. The values described by (21) are computed by the *Master Processing Unit* and are used to control the degrees of freedom of the active vision system and the mobile robot, as will be described in the next section.

3.4. System Control

3.4.1. Introduction

The implementation proposed in this work is based on control loops working in parallel and based on the visual *pursuit* process. Since the the neck inertia and mobile robot inertia are greater than the vergence mechanism inertia, this type of control simulates a control system with different levels - see figure 14. The in-most level comprises cameras and the vergence motors of the active vision system, responsible for tracking the *target* in real-time. This sub-system controls the cameras' position to maintain the visual system fixed in the *target*.

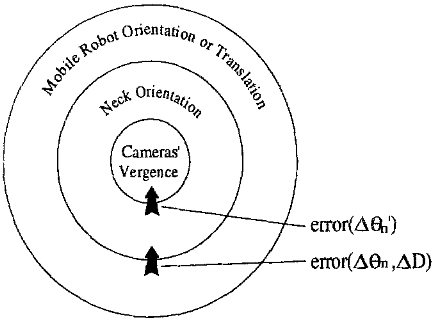

Mobile Robot Orientation or Translation
Neck Orientation
Cameras' Vergence
error($\Delta\theta_n$')
error($\Delta\theta_n$,ΔD)

Figure 14. Graphic scheme of the principle used for the control of the system. Since the mobile robot has more inertial mass than the active vision system, it will receive commands to cover the error. Very often the robot does not have possibility to eliminate all the errors. In that case it is the other fastest degree of freedom that tries to do it. If in the in-most level the object's movement can not be compensated for then it will be assumed that the *target* was lost and the pursuit process will re-start again.

At the intermediate level there is the neck sub-system, that provides the control of the orientation of the vision system and compensates for the cameras' vergence movements. At the out-most level is the mobile robot sub-system that provides the compensation for the orientation of the active vision system and also controls the orientation and distance to the *target*. Conceptually, the error between the actual distance and orientation of the *target* and the system is propagated from the out-most level to the in-most level. This concept is graphically described in figure 14. In each level the error is compensated for by the sub-system associated to each level. This error must be such that the maximum characteristic values of each sub-system are not exceeded. In the cases where the error exceeds these maximum values, the difference of error that can not be compensated for in that level is passed to the next in-most level. This scheme establishes a mechanism to propagate the error through the different control systems, giving more priority to the mobile robot, followed by the neck and eyes at the end. This gives the effect of compensation for the *target* movements, simulating its *pursuit*. The control scheme used is illustrated by the global block diagram in figure 15. If in the in-most level the object's movement can not be compensated for then it will be assumed that the *target* was lost and the pursuit process will re-start again. That is equivalent to the transition between the *Pursuit* and *Rest* state described in figure 3.

3.4.2. Timing Considerations

The pursuit control loop consists basically of three stages: image acquisition, error estimation (the orientation and distance) and error correction. These steps are realized by the *Slave* and *Master Processing Units* at cycles syn-

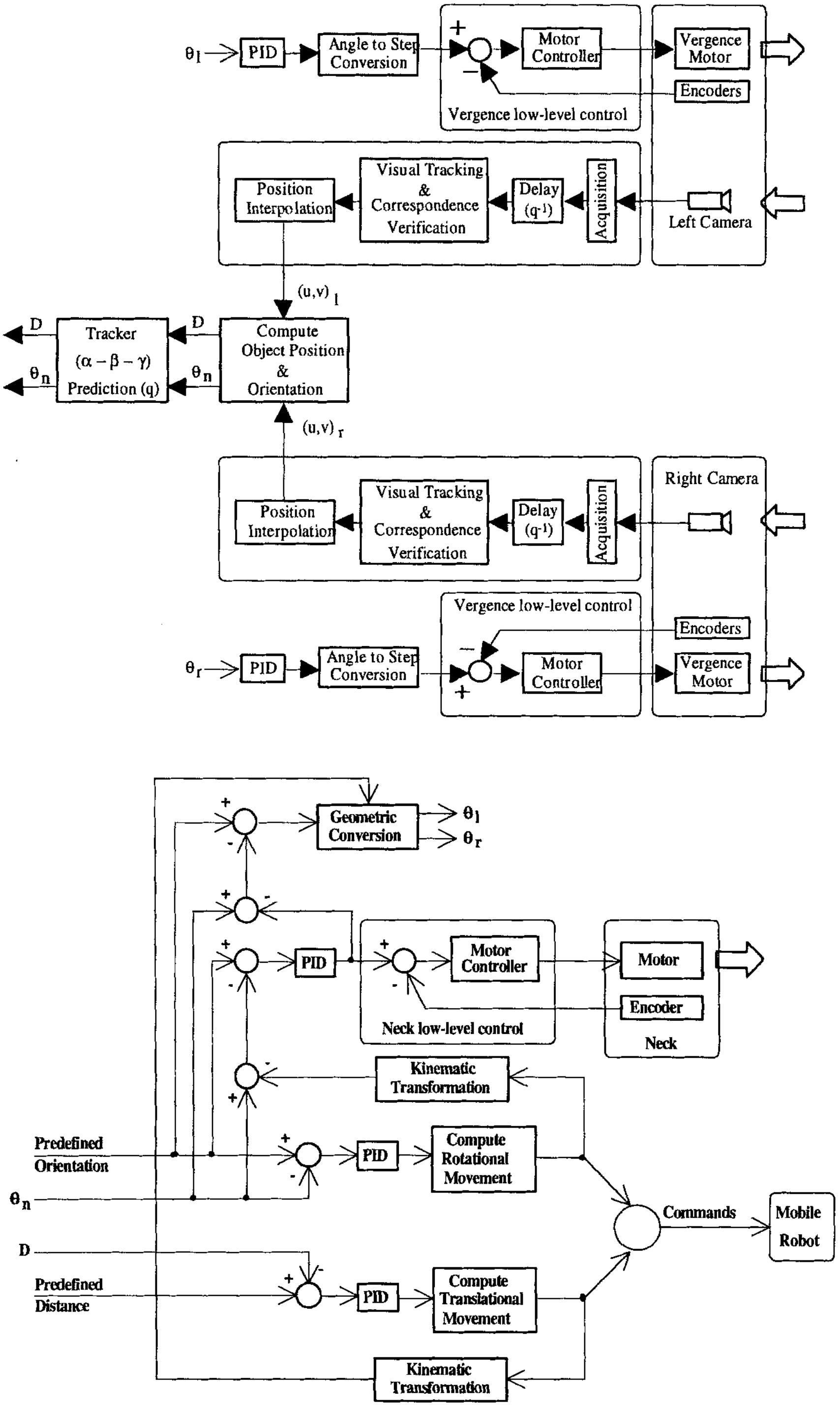


Figure 15. Global control scheme for vergence control and object's position calculation *(top)* and mobile and neck control *(bottom)*.

chronized by a general clock in the system. In the experimental site used to develop the system the clock has a 200$msec$ cycle and the system's parameters are adjusted for that cycle. During this cycle the system performs different computations, depending on the state of the system. The different states of the system are illustrated by figure 3. The images generated by each camera are acquired and analyzed by the *Slave Processing Units.* These units analyze the images and give the position of the *target* in each image. That position gives the necessary information to compute the system state D and θ_n. This state is passed to the (α-β-γ) tracker. The information provided by the *Slave Units* is delayed by one cycle of 200$msec$. To avoid the lateral effects of this delay we use the prediction capabilities of the (α-β-γ) filter to estimate a value for the system state. The *Master Processing Unit* repeatedly performs the control algorithm by using the error between the predicted system state and the desired system state. This error is passed to the different sub-systems according to the illustration in figure 15. This error is passed to the different PID discrete time algorithms implemented in each subsystem. The results will be changes in the positions of the step motors associated with the vision system and the commands for the mobile robot to maintain the desired system state D and θ_n. As described previously the movements executed by the mobile platform are based on two motors associated with each of the driving wheels (rear axle) and are essential to make the compensation for the error in the distance D. The movements permitted with this type of configuration are represented in figure 6 and can be divided into three groups: translational (no angular velocity), rotations around the center of the driving axle represented in figure 6 by **RC** (no linear velocity) and compositions of both movements. The values of the velocities are dependent on the type of movement and the duration time. In our experiments the period of time is a multiple integer of the system cycle (200$msec$).

4. The "Fixed Watchrobot"

The second surveillance system is made up of a binocular active system. If one intruder approaches a critical area the active system starts its tracking. For that purpose several visual routines were implemented [26, 27]. By using optical flow the system is able to track binocularly non-rigid objects in real-time. Simultaneously motion segmentation is performed, which enables the extraction of high quality target images (since they are foveated images). The 3D trajectory of the target is also recovered by using the proprioceptive data. This system does not require fixed lighting conditions since it adjusts its aperture and focus to the current lighting level. Next we will describe the implementation of non-rigid motion tracking and segmentation.

4.1. Visual Routines

4.1.1. Fixation

The fixation process is realized using gross fixation solutions, defined as saccades, followed by fine adjustments in both eyes in order to achieve vergence. Depending on the information available, the fixation process performs different

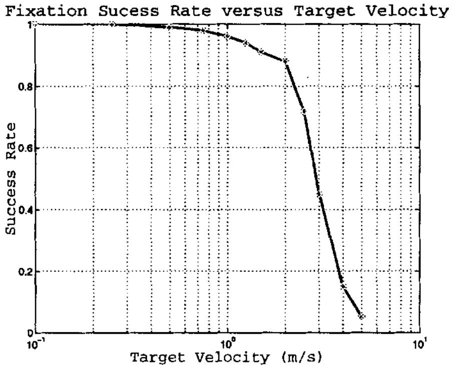


Figure 16. Performance of the fixation process as function of the target velocity.

tasks. If the target is detected in both retinas, a neck (pan and tilt) and eyes saccadic movement is started to gaze the head into the target. However, if the moving target is only detected in one of the retinas, a two-stage fixation process is used (see [26]). The saccades are planned so that the head is at the best state for the next action. Neck and eyes are moved in order to achieve symmetric vergence. The detection of motion is performed computing the image motion flow. In order to be able to detect the moving target during the search target procedure, the image motion induced by the eye egomotion must be subtracted from the overall image motion flow. The center of mass of the moving target is computed and its pixel coordinates converted into rotation angles that the neck or the eye must rotate to foveate on the origin of motion. Due to the latency of the saccade movement (200ms for neck-saccade and 100ms for eye-saccade) an $\alpha - \beta$ filter is used to predict the image position of the target assuming that the target is moving with constant velocity.

Saccade motion is performed by means of position control of all degrees of freedom involved. During the saccade period, and since the camera is moving at high velocity no visual feedback information is processed. In order to achieve perfect gaze of both eyes in the moving target, and since the center of mass is probably not the same in both retinas for non rigid objects, after the saccade a fine fixation adjustment is performed. A grey level cross-correlation tracker is used to achieve perfect fixation of both eyes. The performance of the proposed fixation process as function of the target velocity is presented on figure 16. For a target moving at a cyclopean depth of 5 meters, this fixation process performs well for a maximum target velocity of around $2 meters/sec$.

4.1.2. Smooth Pursuit Using Optical Flow

During this process, the motion of the head must satisfy two basic requirements:

1. stabilize the images of the target on both retinas;

2. maintain fixation on the target.

A prerequisite to use pursuit planning is that the target is not far from the

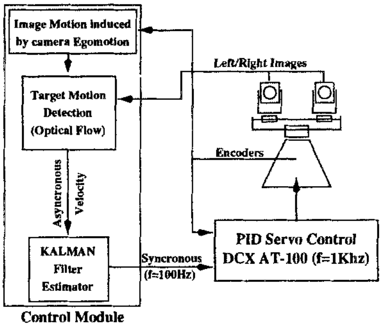


Image Motion induced by camera Egomotion
Target Motion Detection (Optical Flow)
Asyncronous
Velocity
KALMAN Filter Estimator
Control Module
Left/Right Images
Encoders
PID Servo Control DCX AT-100 (f=1Khz)
Syncronous (f=100Hz)

Figure 17. The MDOF High Level Gaze Controller

fixation point of the head (the images of the target on the retinas must not be far from the center of the foveal window). Otherwise, a saccade must be started prior to the smooth-pursuit process. This means that saccades have higher priority than pursuit. After fixating on the target the pursuit process is started by computing the optical flow. During the pursuit process velocity control of the degrees of freedom is used instead of position control as in the case of the saccade. Assuming that the moving object is inside the fovea after a saccade, the smooth pursuit process starts a Kalman filter estimator (using a constant acceleration model), which takes the estimated image motion velocity of the target as an input. Using this approach, the smooth pursuit controller generates a new prediction of the current image target velocity, and this information is sent to the motion servo controller every 10ms (see fig. 17). The smooth-pursuit controller assumes that the moving target is always located on the horopter and the cyclopean eye is pointing straight to the target. With this assumption, the motion induced on the retina by the moving target is almost the same on both eyes (see fig. 18). Two Kalman filters were used to filter the estimated image motion velocities and the velocity used to control the neck pan and tilt joints was considered as the average value of both velocities (cyclopean eye velocity). Maintaining the target on the horopter is accomplished by the vergence process. To maintain tracking, the desired velocity of each of the neck joints should be $V_{des} = V + K_p \cdot \Delta_i$ where K_p is a gain matrix and Δ_i is the position error of the tracking point. The primary function of the first term is to stabilize the images of the target on the retinas. The second term is used to enforce the positional constraint for dynamic fixation.

4.1.3. Vergence Control

To fixate and verge on a target means to keep the images of the target on the image center (center of the fovea), that is, the positions of the target on the image plane must be ${}^{l/r}P_i = (0,0)$. If the image positions of the target on both eyes are known, the 3D position of the target can be recovered using the inverse kinematics of the head. Considering the constraint that both eyes must have equal vergence angles, and the positions of the target on the image plane must be ${}^{l/r}P_i = (0,0)$, the neck and eyes joints rotation angles can be computed based on ${}^{l/r}P_i ={}^{l/r} T_b \cdot P_b$, where P_b represents the target position in the head base coordinate system, and ${}^{l/r}T_b$ represents the transformation matrix between the head base coordinate system and each one of the retina coordinate systems. Consider the existence of a point P_c with coordinates (X_c, Y_c, Z_c) in the cyclopean eye coordinate system, moving with velocity $V_c = -\Omega_c \times P_c - t_c$, being $\Omega_c = [\Omega_1, \Omega_2, \Omega_3]^T$ the angular velocity and $t_c = [t_x, t_y, t_z]^T$ the translational velocity.

After some mathematical manipulation, and considering that the target is verged with equal vergence angles ($\theta = \theta_l = \theta_r$), which means that its coordinates are $P_c = [0, 0, Z_c]$ and its image projection are $[0, 0]_{l/r}$, the retinal image motion flow disparity is given by

$$\Delta_v = \begin{bmatrix} \Delta v_x \\ \Delta v_y \end{bmatrix} = \begin{bmatrix} \frac{t_z f \sin 2\theta}{Z_c} \\ 0.0 \end{bmatrix} \tag{22}$$

being f the focal length of both lenses.

For this particular geometry, the horizontal retinal motion disparity allows the computation of *time-to-contact*

$$\frac{Z_c}{t_z} = \frac{f \sin 2\theta}{\Delta v_x}. \tag{23}$$

Assuming that the target is verged with equal vergence angles on both retinas, $Z_c = \frac{B}{2} \tan\theta$, results for the horizontal retinal motion disparity

$$\Delta v_x = 4 \frac{t_z f \cos^2\theta}{B}, \tag{24}$$

that allows the computation of the Z component of the translational velocity perform by the target, t_z.

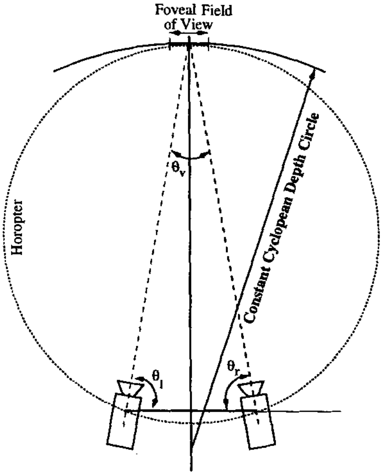


Figure 18. Fixation geometry for vergence and smooth-pursuit

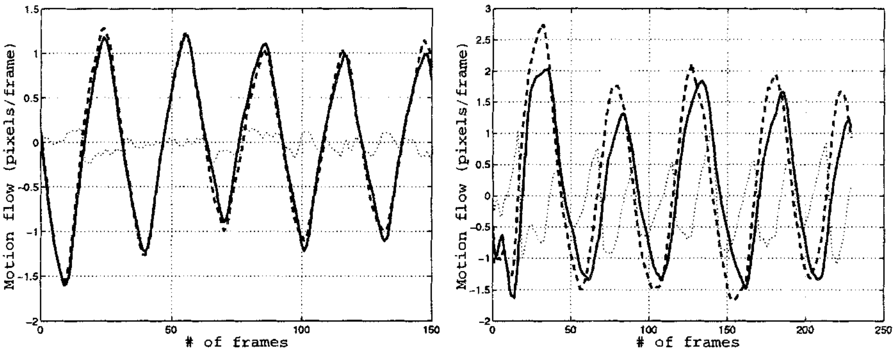


Figure 19. Motion flow and retinal disparity. left) targets moving along the horopter. right) targets moving outside the horopter.

Differentiating Z_c with respect to time, results

$$\frac{\partial Z_c}{\partial t} = \frac{B}{2\cos^2\theta}\frac{\partial \theta}{\partial t}. \tag{25}$$

Considering that

$$\frac{\partial Z_c}{\partial t} = t_z = \frac{B\Delta v_x}{4f\cos^2\theta} \tag{26}$$

and replacing $\frac{\partial Z_c}{\partial t}$ in equation 25 results

$$\frac{\partial \theta}{\partial t} = \frac{\Delta v_x}{2f} \tag{27}$$

that represents the angular velocity of the vergence joints to maintain vergence on the moving target. For this particular geometry for vergence, only the horizontal motion flow disparity is required to control the joints vergence velocity of both retinas.

Figures 19 show the filtered motion vector on both retinas and the retinal disparity, respectively for the case of a target moving along and outside the horopter[1].

The solution adopted to control the vergence of the MDOF robot head is based on two procedures running at different frequencies. The main process is optical flow based and the target motion detected between both retinas (retinal disparity) is used to control the eyes vergence.

Since this process only guarantees that the target center of mass is located in the center of the fovea, we use a grey level cross correlation to adjust vergence and adjust the target depth, at a sample rate 10 times smaller (every 10th frame). The target depth is used to control the auto-focusing of both eyes, taking advantage of the pre-calibration of the focusing depth. Figure 20 shows the target depth obtained by triangulation using the proposed binocular vergence process.

[1] On both figures, full line and dashed line correspond respectively to left and right retina motion, and dotted line represents the retinal disparity

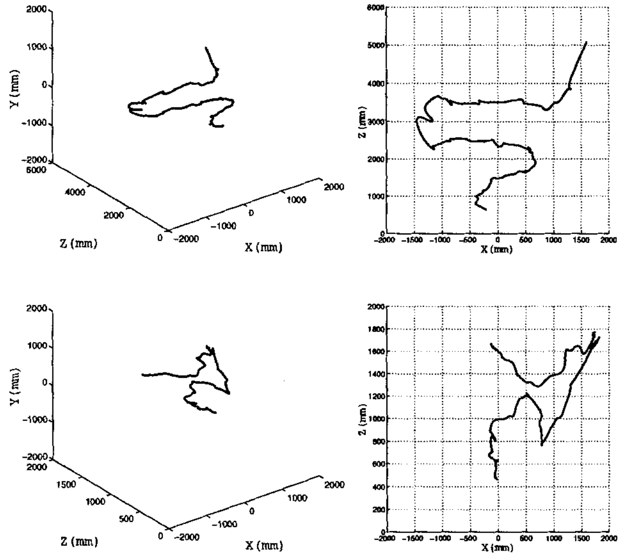


Figure 20. 3D target motion obtained with the proposed vergence and smooth-pursuit process. The right plots represent the top-view of the volume plot, showing the depth movement of the target.

4.2. Target Motion on the Retinas

Unlike the motion of the target in 3D space, which is independent of the head joint motions, the target motions on the retinas are related to the motion of the head joints. Taking into account that the target motions on the retinas caused by the joint motion are generally faster than that caused by the target motion in space, we can not ignore the effects of joint motions on the target motions on the retinas. We considered the analysis of motion described by the two-component model proposed in [24]. Two different motions must be considered to exist in the scene: one caused by the motion being undertaken by the head and the second one coming from the object. Since the first one is known (through inverse kinematics of the head), we only have to compute the latter which we do by using the method described in [24].

To compute the optical flow we model image formation by means of the scaled orthographic projection. Even if we model image formation as a perspective projection this is a reasonable assumption since motion will be computed near the origin of the image coordinate system (in a small area around the center x and y are close to zero). We can therefore assume that the optical flow vector is approximately constant throughout all the image, i.e., $u = p_x$ and $v = p_y$.

The flow is computed on a multiresolution structure. Three different resolutions are used: $16 * 16, 32 * 32, 64 * 64$. These are sub-sampled images. The optical flow computed this way is used to control the angular velocity of the motors.

4.3. Background Image Motion Estimation

The image motion induced by the camera egomotion can be computed using the following equations

$$v_u = \frac{v_x \cdot f_x}{z} - \frac{v_z \cdot f_x \cdot x}{z^2} \qquad v_v = \frac{v_y \cdot f_y}{z} - \frac{v_z \cdot f_y \cdot y}{z^2} \tag{28}$$

where $(f_x,\ f_y)$ represents the focal length of the lens in pixels and $V = \begin{bmatrix} v_x & v_y & v_z \end{bmatrix}^T$ represent the velocity of the point $P = \begin{bmatrix} x & y & z \end{bmatrix}^T$ in the camera coordinate system due to the egomotion of the head.

This velocity V results from the combination of several joints rotation (eye, tilt, swing and pan) and is defined as: $V = V_{eye} + V_{tilt} + V_{swing} + V_{pan}$.

Representing the velocity induced by each joint by $V_{ref} = -V_{Trans} - \Omega \wedge P$ and since each joint only performs rotations ($V_{Trans} = 0$), the velocity induced by each joint can be expressed as $V_{ref} = -\Omega \wedge P$.

Assuming the following angular velocities for each of the rotation joints $\Omega_{eye} = \begin{bmatrix} \Omega_e & \Omega_v & 0 \end{bmatrix}^T$, $\Omega_{tilt} = \begin{bmatrix} \Omega_t & 0 & 0 \end{bmatrix}^T$, $\Omega_{swing} = \begin{bmatrix} 0 & 0 & \Omega_s \end{bmatrix}^T$ and $\Omega_{pan} = \begin{bmatrix} 0 & \Omega_p & 0 \end{bmatrix}^T$ the velocities induced by the rotation of each joint are

$$\begin{aligned} V_{eye} &= -{}^{cam}T_{eye} \cdot (-\Omega_{eye} \wedge P_{eye}) && (29)\\ V_{tilt} &= -{}^{cam}T_{tilt} \cdot (-\Omega_{tilt} \wedge P_{tilt}) && (30)\\ V_{swing} &= -{}^{cam}T_{swing} \cdot (-\Omega_{swing} \wedge P_{swing}) && (31)\\ V_{pan} &= -{}^{cam}T_{pan} \cdot (-\Omega_{pan} \wedge P_{pan}) && (32) \end{aligned}$$

where P_{eye}, P_{tilt}, P_{swing} and P_{pan} represents the coordinates of the point P in each one of the joints coordinate systems.

4.4. The Global Gaze Controller Strategy

The strategy adopted by the gaze controller to combine saccade, smooth pursuit and vergence to track moving objects using the MDOF robot head was based on a *State Transition System.* This controller defines five different states : *Waiting, Fixation, Pursuit, Vergence and μSaccade.* Each one of these states receives control commands from the previous state, and triggers the next state in a closed loop transition system.

The distances of the target to the center of the foveal windows on the retinas are well described by the view direction difference between head fixation point and the target in space. Suppose the minimum view angle difference is α_{min} to guarantee that foveal image processing can still yield reliable results. If $\alpha - \alpha_{min} > 0$ then a μsaccade motion planning is activated. Otherwise, smooth-pursuit is used. During the μsaccade motion planning, no visual information is processed, and we use the Kalman filter estimator to predict the position and velocity of the target after a μsaccade.

4.5. Optical Accommodation

In a real world environment the range of conditions that a camera may need to image under, be it focused distance, spatial detail, lighting conditions or

	Precision	Range	Velocity
Zoom	$Range/90000$	$[12.5\ldots 75]mm$	$\sim 1.2 * range/s$
Aperture	$Range/50000$	$[1.2\ldots 16]$	$\sim 2.2 * range/s$
Focus	$Range/90000$	$[1\ldots \infty]m$	$\sim 1.2 * range/s$

Table 1. Optical structure characteristics of the MDOF active vision system

radiometric sensitivity, can often exceed the capabilities of a camera with a fixed parameters lens.

New motorized lenses have been developed to enable this head to accommodate the optical system in real time (25 images per second), with very good precision. These lenses have controllable zoom, focus and iris and they use small harmonic drive DC motors with encoder feedback information. With such performances (see table 1), the lens is able to make continuous, small optical adjustments required by many algorithms in near real time with excellent precision.

4.6. The Motorized Lens Aperture Control

The aperture control of the motorized lens was designed to work as a background process, running in parallel with the visual behaviors developed. By optimum lens aperture we mean the aperture that enables sharp images acquired by the retinas. Several statistical measures have been made with gray level images such has mean, variance and median. Simultaneosly we measure the focus parameter in order to confirm the correct behavior with the aperture control.

The behavior of these statistical measures were obtained using several lens aperture and focus positions, and we observed that the grey level image variance was the best parameter to be used for the aperture control (see [27]). Maximum grey level variance occurs simultaneously with the maximum of the focus parameter, and in fact sharp images were obtained for that situation. This behavior was shown to be independent of the focus and aperture of the lens. After selecting the grey level variance as the aperture control parameter, we developed a background process that always try to achieve the best aperture position using an *hill climbing* grey level variance maximum search.

4.7. Auto-Focusing Mechanism

The auto-focusing mechanism is mainly based on a previous calibration of the focused target depth D. During the smooth-pursuit process both eyes are verged on the moving target. Taking advantage of the precise information provided as feedback by all MDOF robot head degrees of freedom, a rough estimate of the target depth related to the robot head can be obtained through triangulation of fixated foveated images.

Taking the lens-target depth and the zoom motor setting (that was considered fixed during the smooth-pursuit process), the focus motor position can be computed using the bivariate polynomial that modeled the relationship between the focus and zoom motors settings and the target depth D. Figure 21 presents the behavior of the computed target depth and focus motor setting during a smooth-pursuit process.

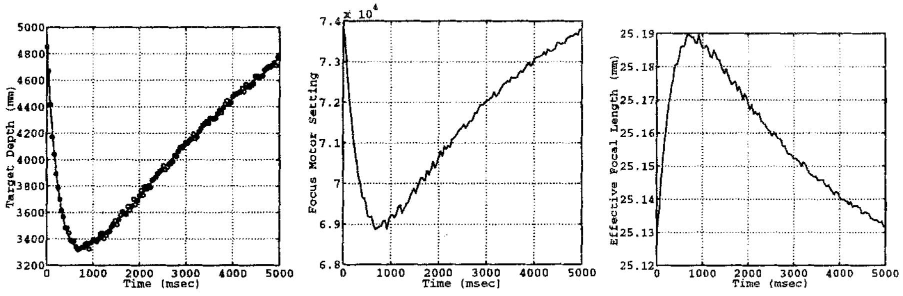


Figure 21. Target Depth and auto-focusing focus motor setting during a 5*sec* smooth-pursuit process.

5. Conclusions

This article describes two autonomous surveillance systems that can be used in complementary ways. Both systems behave as "watchrobots". One of them involves the integration of an active vision system in a mobile platform. For that purpose a control scheme for real-time *pursuit* of objects moving in front of the vehicle was developed. The *pursuit* process controls the system to maintain the initial orientation and distance to the object. The control is based on multiple independent processes, controlling different degrees of freedom of the vision system and the mobile robot's position and orientation. The system is able to operate at approximately human walking rates. The system has limitations and some of them were established as assumptions.

The second system can track non-rigid objects in real-time by using differential flow. Since no shape information is used to track the objects, non-rigidity is not a problem. Also the system can cope with occlusion. The degree of occlusion tolerance depends upon the distance at which the target is located. The closer to the system, the higher the degree of occlusion that can be handled. Since the target is always foveated and focused by the active vision system, good quality images can be extracted for further processing. The 3D trajectories are reconstructed easily by using the proprioceptive data.

References

[1] Y. Aloimonos I W, Bandopadhay A 1988 Active vision. *Intern J Computer Vision*, 7

[2] Aloimonos Y 1990 Purposive and qualitative active vision. In: *Proc. Image Understanding Workshop*, pp 816–828

[3] Blake A, Yuille A (eds) 1992 *Active Vision*. The MIT Press

[4] Aloimonos Y (ed) 1993 *Active Perception*. Computer Vision, Lawrence Erlbaum Associates

[5] Aloimonos Y (ed) 1997 *Visual Navigation: From Biological Systems to Unmanned Ground Vehicles*. Computer Vision, Lawrence Erlbaum Associates

[6] Crowley J 1987 Coordination of action and perception in a surveillance robot. *IEEE Expert*, 2

[7] Kanade T, Collins R T, Lipton A, Anandan P, Burt P, Wixson L 1997 Coopera-

tive multisensor video surveillance. In: *Proc. of the DARPA Image Understanding Workshop*, pp 3–10

[8] Howarth R, Buxton H 1996 Visual surveillance monitoring and watching. In: *ECCV96-II*, pp 321–334

[9] Yedanapudi M, Bar-Shalom Y, Pattipati K R January 1997 Imm estimation for nultitarget-multisensor air-traffic surveillance. *Proc IEEE*, 1:80–94

[10] Davis L, Chellapa R, Yacoob Y, Zheng Q 1997 Visual surveillance and monitoring of human and vehicular activity. In: *DARPA97*, pp 19–27

[11] Rao K 1996 Shape description of curved 3d objects for aerial surveillance. In: *ARPA96*, pp 1065–1076

[12] Shen X, Hogg D 1994 Shape models from image sequences. In: *Proc. ECCV94*, pp 225–230

[13] Baumberg A, Hogg D 1994 Learning flexible models from image sequences. In: *Proc. ECCV94*, pp 299–308

[14] Kollnig H, Nagel H H, Otte M 1994 Association of motion verbs with vehicle movements extracted from dense optical flow fields. In: *Proc. ECCV94*, pp 338–350

[15] Fairley S, Reid I, Murray D 1995 Transfer of fixation for an active stereo platform via affine structure recovery. In: *Proc. ICCV95-I*, pp 100–105

[16] Bradshaw K J, Reid I, Murray D March 1997 The active recovery of 3d motion trajectories and their use in prediction. *IEEE Trans on PAMI*, 19:219–234

[17] Allen P, Timcenko A, Yoshimi, Michelman P April 1993 Automated tracking and grasping of a moving object with a robotic hand-eye system. *IEEE Trans on Robotics and Automation*, 9

[18] Grosso E, Ballard D October 1992 Head-centered orientation strategies in animate vision. Tech. Rep. TR-442, Dept. of Computer Science, University of Rochester, Rochester, N.Y., 1992

[19] Papanikolopoulos N, Khosla P, Kanade T February 1993 Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision. *IEEE Trans on Robotics and Automation*, 9:14–34

[20] Brown C. T D (ed) 1994 *Real-time Computer Vision*. Cambridge University Press

[21] Crowley J L, Christensen H I (eds) 1995 *Vision as a Process*. Springer-Verlag

[22] Burt P, Bergen J, Hingorani R, et al. 1989 Object tracking with a moving camera. In: *Proc. IEEE Workshop on Visual Motion*, Irvine, NATO ASI Series

[23] Batista J, Dias J, Araujo H, de Almeida A July 1993 Monoplanar camera calibration- iterative multi-step approach. In: *Proc. of the British Machine Vision Conference*, Surrey, UK

[24] J. Bergen P. Burt R H S P April 1990 Computing two motions from three frames. Tech. rep., David Sarnoff Research Center

[25] Blackman S 1986 *Multiple-Target Tracking with Radar Applications*. Artech House Inc.

[26] Batista J, Peixoto P, Araújo H September 1997 Real-time vergence and binocular gaze control. In: *IROS97-IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Grenoble, France

[27] Batista J, Peixoto P, Araujo H October 1997 Visual behaviors for real-time control of a binocular active vision system. *Control Engineering Practice*, 5:1451–1461

Sensors for Mobile Robot Navigation

Jorge Lobo, Lino Marques, Jorge Dias, Urbano Nunes, Aníbal T. de Almeida,
Institute of Systems and Robotics
Department of Electrical Engineering
University of Coimbra, 3030 COIMBRA, Portugal
{jlobo, lino, jorge, urbano, adealmeida}@isr.uc.pt

Abstract: The article describes a set of sensors relevant for mobile robot navigation. The article describes their sensing principles and includes examples of robust navigation systems for outdoor/indoor autonomous vehicles, applying different low-cost sensors, exploring high integrity and multiple sensorial modalities. There are many applications, from different sectors that could profit from this type of technology: autonomous mobile platforms for materials handling in industry, warehouses, hospitals, etc.; forestry cutting and undergrowth management equipment; autonomous fire-fighting machines; mining machinery; advanced electrical wheel chairs; autonomous cleaning machines; security and surveillance robots. Advanced sensor systems which are now emerging in different activities from the health care services to the transportation sector and domestic services, will significantly increase the capabilities of autonomous vehicles and will enlarge their application potential.

1. Introduction to Navigation Systems

The level of automation and complexity of modern society is making evermore demands on the technology. Automation is slowly, but surely, getting to the market place. This includes use of mobile platform for many different purposes like materials handling in industry, floor cleaning, semi-autonomous wheel chairs, semi-automatic de-mining, mining trucks, semi-autonomous cars, etc. As these tasks are being automated, a corresponding set of sensor systems is being developed to enable (semi-) autonomous operation. Navigation, in particular, plays an important role in a great variety of tasks, by allowing autonomous operation to go beyond fixed and structured environments.

Navigation [1] may be defined as the process of directing the movements of a vehicle from one point to another. A remarkable variety of physical principles has been utilised in navigational equipment. Some depend on receipt of information from somewhere outside the vehicle itself, either on or the earth or in the sky. They are therefore subject to error or in-operativeness, when such information is erroneous or is lacking, whether from natural or artificial induced sources [1]. Others are based on internal sensing that is dead-reckoning [2], and do not depend on external references, overcoming some of their associated

[1] The word is derived ultimately from the Latin *navis*, ship, and *agere*, to move or direct.
[2] The origin of the term is "deduced reckoning" from sailing days.

problems, but having others, such as drift. These last systems provide relative position measurements and not absolute positioning. The present location of a vehicle is determined by advancing some previous position through known course and velocity information over a given length of time [2].

Traditional dead-reckoning is not truly self-contained, relying on water speed sensors or wheel encoders that interact with the vehicles environment. Land vehicle dead-reckoning is usually associated with odometry, where encoders are used to measure wheel rotation and steering orientation. What in principle might seem a simple and elegant solution, turns out to be prone to errors, the most common source being wheel-slippage and different or irregular floors. Inertial navigation systems depend on measurements carried out entirely within the vehicle, in accordance with the Newtonian laws of motion and gravitation. Therefore, by relying only on inertial sensor measurements, inertial systems are not affected by the vehicles environment, making them non-jammable and self-contained.

If outdoor mobile robots are to be flexible, they have to navigate in unstructured environments, in which some navigation systems are inoperative or have their performance degraded, but where the self-contained inertial navigation system maintains its performance.

For outdoor vehicles the satellite based Global Positioning System (GPS) is available. The inertial system can provide short-term accurate relative positioning and GPS gives absolute positioning, bounding the error. The deduced reckoning of the inertial system is therefore combined with external reference absolute positioning provided by the GPS.

Range sensors have been also used for mobile robot navigation, and a wide set of applications could be found on [3] and [2]. Their information is very important for collision avoidance, path finding and navigation. Section 6 presents the most common range sensors used on mobile robotics, namely optical range sensors and ultrasonic sensors.

2. Inertial Navigation and Inertial Sensors

The principle of generalised relativity of Einstein states that only the specific force on one point and the angular instantaneous velocity, but no other quantity concerning motion and orientation with respect to the rest of the universe, can be measured from physical experiments inside an isolated closed system. Therefore from inertial measurements one can only determine an estimate for linear accelerations and angular velocities. Linear velocity and position, and angular position, can be obtained by integration [4].

Inertial navigation systems implement this process of obtaining velocity and position information from inertial sensor measurements. The basic principle employed in inertial navigation is therefore deduced reckoning. A set of three accelerometers are used to measure acceleration along three orthogonal axes, and their outputs are integrated twice to determine position. To compensate body rotation, three gyroscopes are used to measure rotation rates about three orthogonal axis. In gimballed systems the accelerometers are kept on a gyro-stabilised platform with a high-speed rotor keeping the spatial orientation constant. In strap-down systems all sensors are rigidly fixed to the vehicle and the gyro data is used to transform the accelerometer data to navigation frame of reference. This can be seen as computationally stabilised accelerometer platforms, as opposed to the physically stabilised platforms used in gimballed systems.

Early versions of INS (Inertial Navigation Systems) were used by the Peenumüde group in Germany, in World War II, to guide the V2 rocket. This was one of the first examples of inertial guidance, relying on a gyro assembly to control the missile's attitude and an integrating accelerometer to sense accelerations along the thrust axis.

INS have since become widespread used in avionics and naval applications. High-grade INS were usually gimballed systems, relying on expensive mechanical components and require high-grade sensors to overcome the severe drift problems due to the double integration of acceleration measurements to determine position. Although the cost of gimballed systems has lowered due to technological developments, it is still rather high for robotic applications.

With new sensor development and more computation power, strap-down systems are becoming more accurate and suitable for high-end applications. They provide high performance and reliability at a lower cost, consume less power and are more compact and lightweight [5].

Recent development in accelerometer and gyroscope technology has lead to some new low-cost sensors, as described in the following section. Strap-down systems based on these low-cost inertial sensors offer performance suitable for mobile robotic applications. The inertial system can be used to provide short-term accurate relative positioning, which combined with some other external reference absolute positioning system, to limit the INS absolute position drift error, will provide a suitable navigation system. Complete INS systems have to consider several factors such as the earth's rotation, and compensate for it in the calculations. But for mobile robotic applications, not travelling long distances along the earth's surface, some simplifications can be made [4].

Gyroscopes and accelerometers are known as inertial sensors since they exploit the property of inertia, i.e. resistance to a change in momentum, to sense angular motion in the case of the gyro, and changes in linear motion in the case of the accelerometer. Inclinometers (also known as clinometers, tilt sensors or level sensors) are also inertial sensors. They measure the orientation of the gravity vector, or to be more exact, the resultant acceleration vector acting upon the sensor. In the following sections we will describe a few of these currently available low-cost sensors.

2.1. Accelerometers

A basic accelerometer may be conceived as a basic mass-spring system as shown in figure 1. A proof mass is suspended by an elastic spring (i.e. obeying Hooke's law), the damper is included to control ringing. Upon acceleration of the base frame, the spring must supply a force to make the proof mass keep up, and spring deflection is taken as measure of acceleration. The device is thus a force-measuring instruments which solves the equation

$$F = ma \tag{1}$$

where m is the mass and a acceleration of the sensor, including gravity.

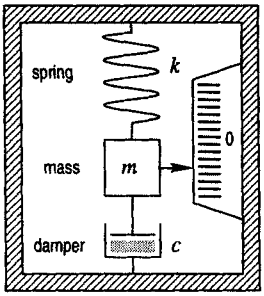


Figure 1. Basic accelerometer.

This damped mass-spring system with applied force constitutes a classical second order mechanical system and the the system's response will depend on the amount of damping. If under-damped, overshoot and oscillation will occur. The system will have the shortest rise time without overshoot when it is critically damped. When the system is over-damped, there will be no overshoot but the rise time will be slow [6].

Practical accelerometers vary in design and technology, but all are based in the equation $F = ma$ in some way. They can be electromagnetic, vibrating string, gyro-pendulum, optical, piezoresistive, piezoelectric, capacitive, amongst others. See [7] and [8] for an overview of some of the older accelerometer technologies.

2.1.1. Silicon Accelerometers

In recent years micro-machined accelerometers have become widely available, largely due to the ability to produce them at low cost. The needs of the automotive industry, namely for airbag deployment systems, encouraged silicon sensor development, enabling the batch-fabrication of the integrated accelerometer sensors. The current commercially available silicon accelerometers incorporate amplification, signal conditioning and temperature compensation. There are presently three main types of micro-machined low cost accelerometers. These are the capacitive, piezoelectric and piezo-resistive types. The piezoelectric sensors have no DC response, making them unsuitable for inertial navigation systems. In the piezo-resistive sensors the acceleration causes a sensing mass to move with respect to a frame, creating stress in a piezo-resistor, which changes its resistor value. The capacitive sensors rely on the displacement of capacitive plates due to the acceleration, creating a mismatch in the capacitive coupling. This change is used to generate a signal proportional to the acceleration applied to the sensor. Some recent devices are open loop sensors, others have a force balancing feedback loop that keeps the sensing element at its central position, gaining improved linearity. These devices are built so as to have a sensing axis and reduced off-axis sensitivity. Some are three-axial, incorporating three accelerometers in one sensor, simplifying mounting and alignment. These sensors present different measurement ranges from ± 2 g up to ± 500 g.

Typical applications of such devices in the automotive industry include frontal impact airbag systems, suspension control, braking control and crash testing. They also find applications in industrial vibration monitoring, transportation shock monitoring and motion control. This large market will push the development of the technology further, and improved performance and lower cost sensors are to be expected.

A silicon accelerometer typically has a silicon spring and a silicon mass. In open loop configurations the acceleration is computed by measuring the displacement of the mass. Typical errors include: non-linearity of the spring; off-axis sensitivity; hysteresis due to the springs or hinges; rotation-induced errors (i.e. when body rotation adds rotational acceleration to the linear acceleration we intend to measure); and accelerometer signal noise.

For higher precision, force balancing closed loop configurations are implemented. Forces are applied to the mass to make it track the frame motion perfectly, and thus zero-balance the mass. Typical restoring forces used in silicon accelerometers include magnetic, piezoelectric and electrostatic. The sensor output will be given by the amount of force necessary to zero-balance the mass. By zero-balancing the mass, errors due to distortions and spring nonlinearity are minimised. The input dynamic range and bandwidth is increased. Weaker hinges can be used, reducing hysteresis effects, and mechanical fatigue is minimised. No damping fluid is required, allowing operation in vacuum, and mechanical resonance avoided. Improved precision is thus accomplished.

In order to sense the proof mass displacement, either to directly give the output signal or control the zero-balancing loop, a number of sensing techniques is available. These include piezo-resistive, piezoelectric, capacitive and optical. The piezoelectric accelerometers rely on the deposition of a piezoelectric layer onto the silicon springs. They have a high output at relatively low current, but have high impedance and no DC response. Optical silicon accelerometers rely on the changing characteristics of an optical cavity, due to mass displacement. Radiation penetrating the cavity is band-pass dependent of the mass displacement. This technology has been used in high-resolution, but rather high cost, pressure sensors [9]. Piezo-resistive and capacitive both have DC response and relatively low cost, making them suitable for low-grade inertial navigation systems.

Piezo-resistive Accelerometers

The first silicon accelerometer prototype was built in 1976 [9]. This device had a single cantilever structure, was fragile and had to be damped with a liquid. Despite its limitations, it represented a significant step from the attachment of silicon strain sensors onto metal diaphragms, to having the resistor diffused onto single-crystal silicon. The basic design structures that have evolved for silicon are shown in figure below 2.

The single cantilever has, in theory, the highest sensitivity, but has more off-axis errors and is rather fragile. The double cantilever provides good off-axis

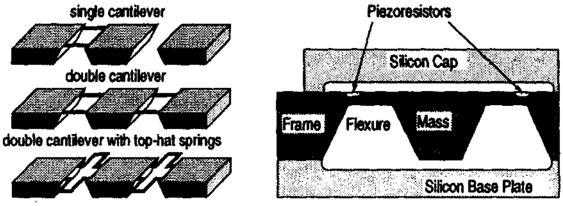


Figure 2. Design structures for the piezo-resistive accelerometer and cross-section of double cantilever sensor (adapted from [10]).

cancellation and is more robust. The folded springs of the top-hat configuration allow for large displacements in a smaller area, thus reducing the cost of the sensor.

Capacitive Accelerometers

In capacitive accelerometers, proof mass displacement alters the geometry of capacitive sensing elements.

One design of capacitive silicon accelerometers uses a main beam that constitutes the proof mass, with springs at each end. The beam has multiple centre plates at right angles to the main beam that interleave with fixed plates attached to the frame on each side, forming a comb-like symmetric structure. This design allows sensing of positive and negative acceleration along the axis of the main beam in the sensor plane.

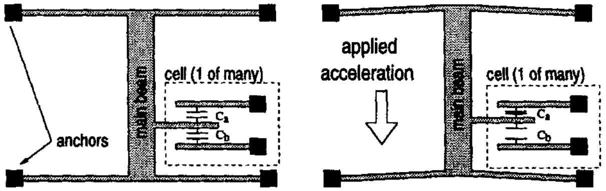


Figure 3. Capacitive comb finger array accelerometer working principle (adapted from [11]).

Each of the centre plates fits between two adjacent fixed plates, forming a capacitive divider, as shown in figure 3. The two fixed plates are driven with an equal amplitude but opposite polarity square wave signals, typically 1 MHz.

With no acceleration, the two capacitances are approximately equal and the centre plate will be at approximately zero volts. Any applied acceleration causes a mismatch in plate separation which results in greater capacitive coupling from the nearest fixed plate; a voltage output can thus be detected on the centre plate. The acceleration signal is contained in the phase relative to the driving signal, thus a synchronous demodulator technique is actually used to extract the relatively low frequency acceleration signal.

The resulting acceleration signal is used in a feedback loop to force balance the sensor, impeding the deflection and servoing the sensor back to its 0 g position. The balancing force is obtained electrostatically, caused by driving the centre plates to a voltage proportional to the acceleration signal. The force-balancing servo loop response has to be fast enough and flat enough to track fast level changes, keeping the sensor nearly motionless, and minimising the errors.

2.2. Inclinometers

Though not strictly accelerometers, inclinometers or clinometers, measure the orientation of the resultant acceleration vector acting upon the vehicle. If the vehicle is at rest, this means its orientation with respect to level ground.

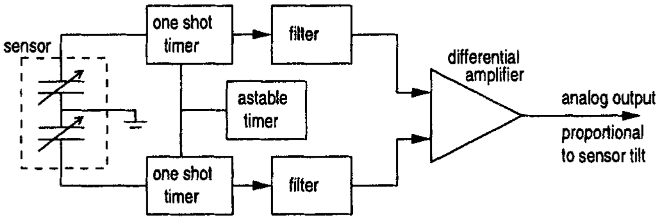


Figure 4. AccuStar inclinometer block diagram.

The concept of the sensor is based on a dielectric fluid, with an air bubble, inside a capacitive sensor. When the sensor is tilted the bubble, moving under the force of gravity, changes the capacitance of the sensor elements. The resulting differential generates an output signal which reflects the relative tilt in the sensing axis as shown in figure 4. Due to the fluids inertia and settling time, and sometimes the measurement method, inclinometers tend to have a delayed response.

The concept of the sensor is based on a dielectric fluid with an air bubble inside a dome shaped capacitive sensor. The sensing dome is divided into four quadrants. When the sensor is tilted, the bubble, moving under the force of gravity, changes the capacitance of the sensor elements in each quadrant. The resulting differential generates an output signal which reflects the relative tilt of the device in either x- or y-axis.

Other designs, still using the principle of the spirit level, measure resistance to obtain the tilt. These sensors have a suitably curved tube, with an electrically conducting liquid and gas bubble inside, and three electrodes. When the sensor is tilted the bubble's position relative to the electrodes changes, causing a difference in the electrical resistance between electrodes proportional to the tilt.

When using inclinometers care should be taken, when accelerations other than gravity are present, since the tilt will be measured relative to the resultant vector. If the sensor is tilted by an angle α to the horizontal and is subject to an acceleration a in a direction normal to the sensor's measuring axis in the horizontal plane, the tilt sensor will not measure α. The measured angle will be

$$\alpha_{measured} = \alpha + \tan^{-1}\left(\frac{a}{g}\right) \tag{2}$$

where g is the modulus of the gravity vector [12].

2.3. Gyroscopes

The mechanical gyroscope [3], a well known and reliable but expensive rotation sensor, based on the inertial properties of a rapidly spinning rotor, has been around since the early 1800s. The spinning rotor or flywheel type of gyroscope uses the fundamental characteristic of the angular momentum of the rotor to resist changing its direction to either provide a spatial reference or to measure the rate of angular rotation [5]. Many different designs have been built, and different methods used to suspend the spinning wheel. See [7] for some examples of such devices.

Optical gyroscopes measure angular rate of rotation by sensing the resulting difference in the transit times for laser light waves travelling around a closed path in opposite directions - see figure 5. This time difference is proportional to the input rotation rate, and the effect is known as the 'Sagnac effect', after the French physicist G. Sagnac. Sagnac, in fact, demonstrated that rotation rate could be sensed optically with the Sagnac interferometer as long ago as 1913 [5].

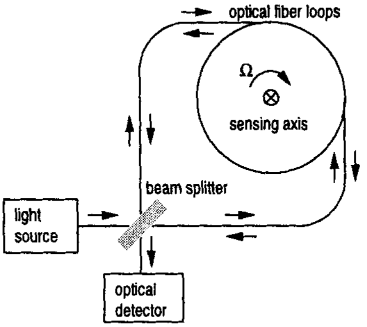


Figure 5. Simplified diagram of optical fibre gyroscope (adapted from [13]).

The communications industry has made optical fibres increasingly available, enabling the construction of low-cost fibre optic gyroscopes. These devices, named FOG or OFG for short, use multiple loops of optical fibre to construct the closed loop path, and semiconductor laser diodes for the light source. A simplified diagram is shown in figure 5. The beam splitter divides the laser beam into two coherent components. The difference of travelling time between the two beams, caused by the difference in optical path lengths, is detected as the interference between the two beams by an optical detector. Several manufactures have produced relatively inexpensive optical fiber gyros for car navigation systems.

But even lower cost, and becoming increasingly compact are the vibrating structure gyroscopes. These use the Coriolis effect whereby an object with linear motion in a rotating frame of reference, relative to inertial space, will experience a so called Coriolis acceleration given by

[3] from the Greek word *gyros* meaning rotation and *skopein* meaning view.

$$\vec{a}_{coriolis} = 2\vec{\varpi} \times \vec{v} \tag{3}$$

where $\vec{\varpi}$ is the angular velocity of the rotating frame and the object's velocity $\vec{v}$ is given in the rotating frame of reference. Imagine a ball rolling across a rotating table. An outside observer would see it moving along a straight line. But an observer on the table would see the ball following a non-linear trajectory, as if a mysterious force was driving it. This apparent force is called the Coriolis force. You can see from equation 3 that the Coriolis force will be perpendicular to both the rotation axis and the objects linear motion.

2.3.1. Vibrating Structure Gyroscopes

The basic principle of Vibrating Structure Gyroscopes (VSG), is to have radial linear motion and measure the Coriolis effect. If a sensing element is made to vibrate in a certain direction, say along the x-axis, rotating the sensor around the z-axis will produce vibration in the y direction with the same frequency. The amplitude of this vibration is determined by the rotation rate. The geometry used takes into account, amongst other factors, the cancelling out of unwanted accelerations.

The common house fly, in fact, uses a miniature vibrating structure gyro to control its flight. A pair of small stalks with a swelling at their ends constitute radially oscillating masses that will be subject to Coriolis forces when yaw is experienced. These forces will generate muscular signals that assist the acrobatic fly [3].

The Vibrating Prism Gyroscope

A piezoelectric vibrating prism sensor can be used for sensing angular velocity. The device' s output is a voltage proportional to the angular velocity. The principle of the sensor is outlined in figure 6. Inside the device there is an equilateral triangle prism made from *elinvar*, elastic invariable metal, which is fixed at two points. Three piezoelectric ceramic elements are attached to the faces of prism, one on each side. The prism is forced to vibrate by two of the piezoelectric elements, whilst the other is used for feedback to the drive oscillator. These two elements are also used for detection. When there is no rotation they detect equally large signals. When the prism is turned, Coriolis forces will affect the prism vibration and the sensing piezoelectric elements will receive different signals. The difference between the signals is processed by the internal analogue circuits to provide an output voltage proportional to the angular velocity [14].

The Tuning Fork Gyroscope

A micro-miniature double-ended piezoelectric quartz tuning fork element can be used to sense angular velocity. The sensor element and supporting structure are fabricated chemically from a single wafer of mono-crystalline piezoelectric quartz.

The drive tines, being the active portion of the sensor, are driven by a high frequency oscillator circuit at a precise amplitude, producing the radial

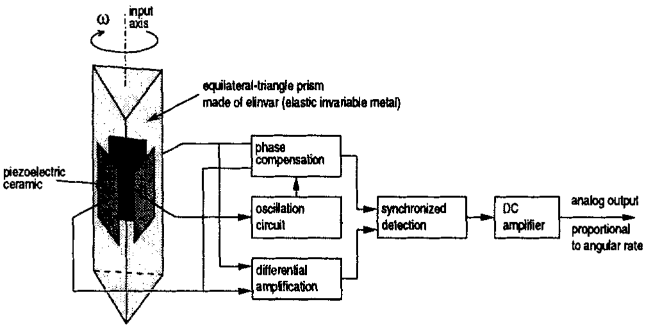

ω
input axis
equilateral-triangle prism made of elinvar (elastic invariable metal)
piezoelectric ceramic
phase compensation
oscillation circuit
differential amplification
synchronized detection
DC amplifier
analog output
proportional to angular rate

Figure 6. Piezoelectric vibrating prism gyroscope (adapted from [14]).

oscillation of the tines along the sensor plane, as shown in figure 7. A rotational motion about the sensor's longitudinal axis produces a DC voltage proportional to the rate of rotation due to the Coriolis forces acting on the sensing tines. Each tine will have a Coriolis force acting on it of:

$$F = 2m\varpi_i \times V_r \tag{4}$$

where m is the tine mass, V_r the instantaneous radial velocity and ϖ_i the input angular rate. This force is perpendicular to both the input angular rate and the instantaneous radial velocity.

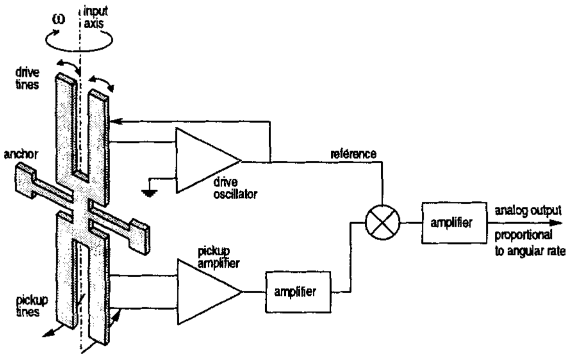


Figure 7. Example of tuning fork gyroscope (adapted from [15]).

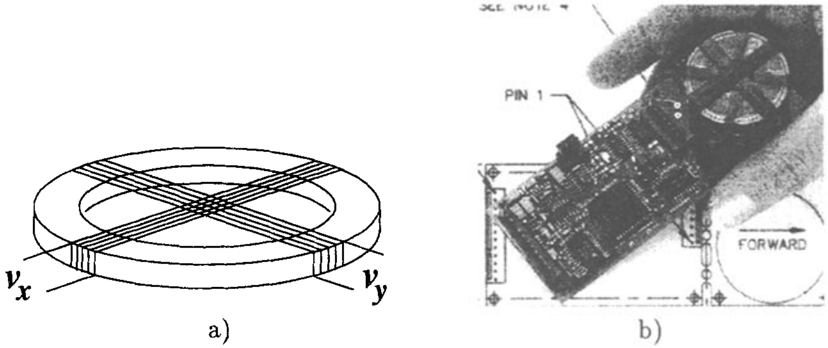
The two drive tines move in opposite directions, and the resultant forces are perpendicular to the plane of the fork assembly, and also in opposite directions. This produces a torque which is proportional to the input rotational rate. Since the radial velocity is sinusoidal, the torque produced is also sinusoidal at the same frequency of the drive tines, and in-phase with the radial velocity of the tine.

The pickup tines respond to the oscillating torque by moving in and out of plane, producing a signal at the pickup amplifier. The sensed pickup signal is then synchronously demodulated to produce the output signal proportional to the angular velocity along the sensor input axis.

3. Fluxgate Compass

One good source for absolute heading of outdoor mobile robots is the earth's magnetic field. The magnetic compass has long been used in navigation. Mechanical magnetic compasses have evolved from the simple magnetised needle floating in water, to the more sophisticated and time proven systems in use today. Much more practical and suitable for mobile outdoor robots are the fluxgate compasses. These saturable-core magnetometers use a gating action on AC-driven excitation coils to induce a time varying permeability in the sensor core, hence the name fluxgate. Highly permeable materials present a lower magnetic resistance path and will draw in the lines of flux of an external uniform magnetic field. If the material is forced into saturation by an additional magnetising force, the material will no longer affect the lines of flux of the external field. The fluxgate sensor uses this saturation phenomenon by driving the core element into and out of saturation, producing a time varying magnetic flux density that will induce e.m.f. changes in properly oriented sensing coils. These variations will provide a measurement of the external DC magnetic field. See [2] for a more detailed description.

One example of such a device is the C100 model from KVH Industries, Inc. This fluxgate sensor uses a saturable ring core element, free floating in an inert fluid within a cylindrical lexan housing. The lexan housing is surrounded by windings which electrically drive the coil into and out of saturation. Pulses, whose amplitude is proportional to the sensed horizontal component of the earth's magnetic field, are detected by two secondary windings. The secondary windings are at right angles, as can be seen in figure 8, thereby providing data on the x and y horizontal components of the earth's magnetic field.



a) b)

Figure 8. a) Flux-gate sensor element (adapted from [16]); b) KVH-C100 Compass engine (photo adapted from [17])

These signals are then converted to a DC level, digitised and sent to a microprocessor that calculates the azimuth angle as

$$\Phi = \tan^{-1} \frac{v_x}{v_y} \tag{5}$$

The microprocessor also performs compensations based on previous calibrations that substantially increase the sensor's accuracy. Several output modes are available, including a serial RS-232 port to provide heading information and also perform compass configuration.

4. Global Positioning System - GPS

4.1. Introduction

One of the most relevant external sensors, for outdoor applications, is the Global Positioning System (GPS). Navigation employing GPS and inertial sensors in a synergistic relationship and the integration of these two types of sensors not only overcomes performance issues found in each individual sensor, but could produce a system whose performance exceeds that of the individual the sensors.

The inertial systems accuracy degrades with time, but GPS provides bounded accuracy. GPS and INS complement each other, and their information can be combined to provide an overall better system. The GPS enables calibration and correction of the INS drift errors by means of a Kalman filter. The INS can smooth out the step changes in the GPS position output, which can occur when switching to another satellite or due to other errors.

4.2. Overview of the GPS system

The GPS system was designed for, and is operated by, the U. S. military system. Its scope for military missions has been far outgrown with civilian applications, both commercial and scientific. The U. S. Department of Defence funds and controls the system, and civilian users world-wide can use the system free of charge and restrictions. However the accuracy is intentionally degraded for the non-military applications. The satellite-based systems can provide service to an unlimited number of users since the user receivers operate passively (i.e. receive only). The system provides continuous, high accuracy positioning anywhere on the surface of the planet and near space region, 24 hours a day, under all weather conditions. GPS also provides a form of co-ordinated universal time. The users receivers are small and lightweight, making hand-held global positioning systems a reality. See [18] for a brief history and description of the system or [19] for a more detailed description and underlying principles.

The GPS system is composed of three segments. The space segment consists of the GPS operational constellation of satellites. The constellation consists of 24 earth satellites, including 3 active spares, in 12 hour orbits. They are arranged in six orbital planes, separated by 60° in longitude, and inclined at about 55° to the equatorial plane. The satellites' nearly circular orbit, with an altitude of around 20 000 km, is such that they repeat exactly twice per sideral day. This implies that they repeat their ground track 4 minutes later each day. This constellation provides the user with between 5 and 8 satellites visible from any point on earth. GPS operation requires a clear line of sight, and since the signals cannot penetrate water, soil, or walls very well, satellite visibility can be affected by those types of obstacles. The control segment consists of a world-wide system of tracking stations. A Master Control Station tracks the position of all satellites and maintains the overall system time standard. The other monitor stations measure signals from the satellites, allowing the Master Station to compute the satellites exact orbital parameters (ephemeris) and clock corrections, and upload them to the satellites, at least once a day. The satellite then sends subsets of this information to the user receivers. Satellites have redundant clocks, allowing them to maintain synchronous GPS system time. The user segment consists of the GPS receivers. They convert the satellite signals into position, velocity, and time estimates.

Position measurement is based on the principle of range triangulation. The receiver needs to know the range to the satellites and the positions of these satellites. The satellites positions can be determined by the ephemeris data broadcast from each satellite.

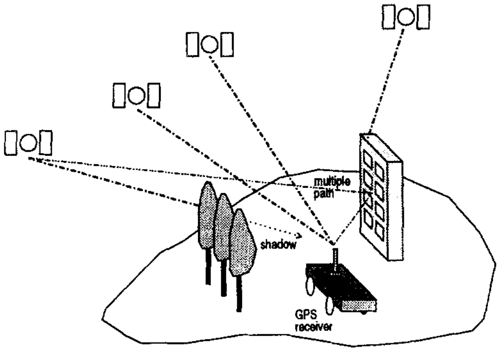


Figure 9. GPS basic idea.

The ranges are determined by measuring the signal propagation time from each satellite to the receiver. The receiver needs a local clock synchronised with the GPS system time. The atomic clock used in the satellites are impractical for the user receivers, and cheap crystal oscillators are used instead. These introduce a user clock bias that effectively adds a fourth unknown in the triangulation. The computed range to each satellite will be equally affected by the same clock bias dependent variable. These erroneous ranges are called pseudoranges. To determine position in three dimensions, four equations are needed to determine the four unknowns. For each satellite the following equation holds:

$$pseudorange_{sat_i} = \sqrt{(x - x_{sat_i})^2 + (y - y_{sat_i})^2 + (z - z_{sat_i})^2} + c\Delta t \qquad (6)$$

where receiver and satellite positions are expressed in Cartesian geocentric coordinates, c is a constant, and Δt is the user clock bias, which it the same for every satellite, since the satellite clocks are synchronous [20]. Four satellites will be needed, and the three dimensional position will be given by the simultaneous solution(s) of the four equations. This is done in practice with a standard Newton-Raphson method for solving simultaneous non-linear equations. When more satellites are used, or some prior knowledge is available, a least squares technique is used. When altitude is known, navigation in two dimensions can be done with only three satellites.

All satellites broadcast two microwave carrier signals, L1 (1575.42 MHz)

and L2 (1227.60 MHz), as well as UHF intra-satellite communications link, and S-band links to ground stations. The dual frequency approach allows estimation of ionospheric propagation delay at the receiver since the delay is frequency dependent. Satellites use unique Pseudo Random Noise (PRN) codes to modulate the signals, enabling satellite identification at the receiver end. The use of a particular type of PRN codes allows receivers with antenna only a few inches across to extract very low power signals from background noise by correlating them with expectations. The PRN codes of the different satellites are nearly uncorrelated with respect to each other, allowing receivers to "tune in" to different satellites by generating the appropriate PRN code and correlating with the received signal. The receiver computes satellite signal propagation time by shifting the self generated PRN code sequence in time, until the correlation function peaks. The time shift introduced gives the signal propagation time, including clock bias.

4.3. GPS errors

Selective Availability (SA) is a deliberate error introduced to degrade system performance for non-U.S. military and government users. The system clocks and ephemeris data is degraded, adding uncertainty to the pseudo-range estimates. Since the SA bias, specific for each satellite, has low frequency terms in excess of a few hours, averaging pseudo-ranges estimates over short periods of time is not effective [21]. The potential accuracy of 30 meters for C/A code receivers is reduced to 100 meters.

Satellites are subject to deviations from their planned ephemeris, introducing ephemeris errors. The satellite clocks degrade over time, and if the ground control leaves then uncorrected, unwanted clock errors are introduced.

The troposphere (sea-level to 50 km) introduces propagation errors that are hard to model, unless local atmospheric data are available. The ionosphere (50 km to 5000 km) also introduces delays, and some compensation can be made with modelling based on almanac data. Dual frequency receivers allow direct estimation of ionospheric propagation delay since the delay is frequency dependent.

Shadows and multiple paths, as seen in figure 9, can also introduce errors. Shadows reduce the number of visible satellites available for positioning. Multiple path error is caused by reflected signals from surfaces near the receiver and can be difficult to detect and hard to avoid. The reflected signal can either interfere, or be mistaken for, the straight line path signal form the satellite.

4.4. Differential GPS (DGPS)

The basic idea behind differential positioning is to correct bias errors at the receiver with measured bias errors at a known nearby position. The reference receiver, knowing the satellites' ephemeris and the expected signal propagation delay, can calculate the corrections for the measured transit times. This correction is computed for each visible satellite signal, and sent to the user receiver. These pseudo-range corrections can be radio broadcast to multiple user receivers. A more simplistic approach would be to simply correct the user position with the known position offset of the reference receiver. But this would only provide good corrections if both receivers where using the same set of satellites.

Another differential technique is the carrier-phase DGPS, also known as interferometric GPS, which bypasses the pseudo-random code and uses the high resolution carriers. The phase shift between signals received at the base and mobile units gives the signal path difference. It is also called code-less DGPS, as opposed to the coded DGPS where the pseudo-random noise code sequence is used to estimate signal path differences for each satellite. This technique is typically used in surveying applications, where accuracy of a few centimetres can be achieved. Besides the high cost, code-less DGPS requires a long set-up time, is subject to cycle slip, and unsuitable for fast moving vehicles.

5. Visual and Inertial Sensing Integration

In human and other animals the ear vestibular system gives inertial information essential for navigation, orientation or equilibrium of the body. In humans this sensorial system is located in the inner ear and it is crucial for several visual tasks and head stabilisation. The human vestibular system appears as a special sensorial modality, that co-operates with other sensorial systems and gives essential information for everyday tasks.

One example of co-operation is between the vestibular sensorial system and the visual system. It is well known that the inertial information plays an important role in some eye and head movements [22].

The information provided by the vestibular system is used during the execution of these movements, as described by Carpenter [22]. However the inertial information is also important for head-stabilisation behaviours, including the control of posture and equilibrium of the body.

The inertial information can also be useful on applications with autonomous systems and artificial vision. In the case of active vision systems, the inertial information gives a second modality of sensing that gives useful information for image stabilisation, control of pursuit movements, or ego-motion determination when the active vision system is used with a mobile platform. This kind of sensorial information is also crucial for the development of tasks with artificial autonomous systems where the notion of horizontal or vertical is important, see Viéville for one example [4].

The inertial system prototype described in this section is used in a mobile robot with an active vision system as illustrated in figure 10. The following sections describe the mobile system used and a first approach of inertial and vision data fusion, namely in identifying the ground plane.

5.1. An Inertial System based on Solid-State Devices

To study the integration of the inertial information in artificial autonomous systems that include active vision systems it was decided to develop an inertial system prototype composed of low-cost inertial sensors. Their mechanical mounting and the necessary electronics for processing were also designed. The sensors used in the prototype system include a three-axial accelerometer, three gyroscopes and a dual-axis inclinometer.

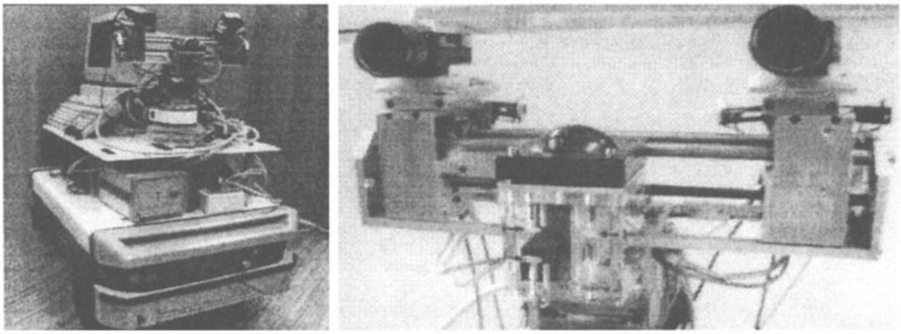


Figure 10. The mobile system with the active vision system. The inertial system prototype was designed to use with this system.

The three-axial accelerometer chosen for the system, while minimising eventual alignment problems, did not add much to the equivalent cost of three separate single-axis sensors. The device used was Summit Instruments' 34103A three-axial capacitive accelerometer. In order to keep track of rotation on the x-, y- and z-axis three gyroscopes were used. The piezoelectric vibrating prism gyroscope Gyrostar ENV-011D built by Murata was chosen. Since orientation is obtained by integration of the angular velocity over time, drift errors will occur. In the prototype, a magnetic flux-gate compass was used to overcome this problem, providing an external reference - see [23] for details. To measure tilt about the x and y-axis a dual axis AccuStar electronic inclinometer, built by Lucas Sensing Systems, was used.

To handle the inertial data acquisition, and also enable some processing, a micro-controller based card was built. This card has analogue filters, an A/D converter as is based on Intel‘s 80C196KC micro-controller. The robot's master processing unit has an EISA bus interface, where the card is connected along with another for image acquisition and processing. This card is a frame grabber module with two Texas Instruments TMS320C40 DSPs that handles the video processing.

Figure 11 shows the architecture of the system and the computer that supervises the active vision, moving platform and inertial system. The inertial sensors were mounted inside an acrylic cube, enabling the correct alignment of the gyros, inclinometer (mounted on the outside) and accelerometer, as can be seen in the close up of figure 10. This cube is connected to, and continuously monitored by, the micro-controller card in the host computer. The inertial unit is placed at the middle of the stereo camera baseline. The head co-ordinate frame referential, or Cyclop $\{C_y\}$is defined as having the origin at the centre of the baseline of the stereo cameras.

The inclinometer data can be used to determine the orientation of the ground plane. In order to locate this plane in space at least one point belonging to the ground plane must be known. When the vehicle is stationary or subject to approximately constant speed the inclinometer gives the direction of $\vec{g}$ relative to the Cyclop referential $\{C_y\}$. Assuming the ground is levelled,

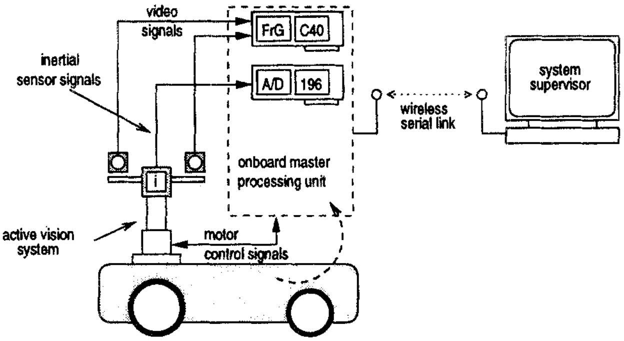


Figure 11. System Architecture. The inertial system processing board uses the Master processing unit as host computer.

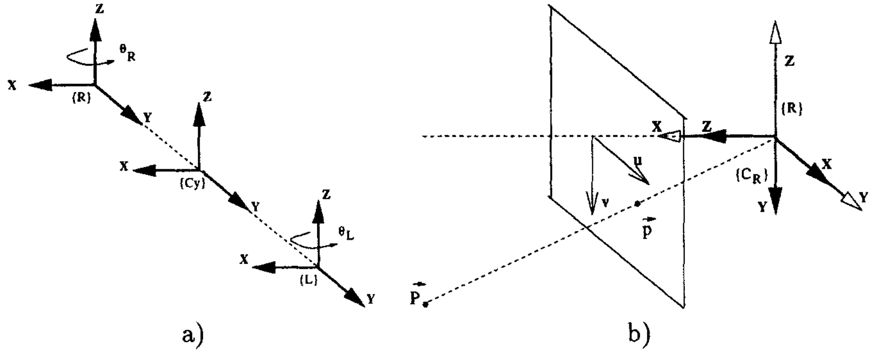


Figure 12. a) System Geometry; b) The camera referential and picture co-ordinates.

and with α_x and α_y being the sensed angles along the x and y-axis, the normal to the ground plane will be

$$\hat{n} = -\frac{\vec{g}}{\|\vec{g}\|} = \frac{1}{\sqrt{1 - \sin^2\alpha_x \sin^2\alpha_y}} \begin{bmatrix} -\cos\alpha_x \sin\alpha_y \\ -\cos\alpha_y \sin\alpha_x \\ \cos\alpha_y \cos\alpha_x \end{bmatrix} \tag{7}$$

given in the Cyclop frame of reference. Using this inertial information the equation for the ground plane will be given by

$$\hat{n}.\vec{P}_f + h = 0 \tag{8}$$

where $\vec{P}_f$ is a point in the plane and h is the distance from the origin of $\{C_y\}$ down to the ground plane.

To obtain a point belonging to the ground plane it will be necessary to find the correspondence between points in the image. Establishing this correspondence will give us enough equations to determine the 3D co-ordinates, if a few vision system parameters are known. However if visual fixation is used, the geometry is simple and the reconstruction of the 3D fixated point is simplified, as can be seen in figure 13. Notice that the visual fixation can be achieved by controlling the active vision system and the process allows a fast and robust 3D reconstruction of the fixation point. This mechanism was developed in the ISR laboratory and is described in [24] and [25]. If the active vision system fixates in a point that belongs to the ground plane, the ground plane could be determined in the Cyclop referential $\{C_y\}$ using the reconstructed point $\vec{P}_f$ and the inclinometer data. Hence, any other correspondent point in the image can be identified as belonging or not to the ground plane.

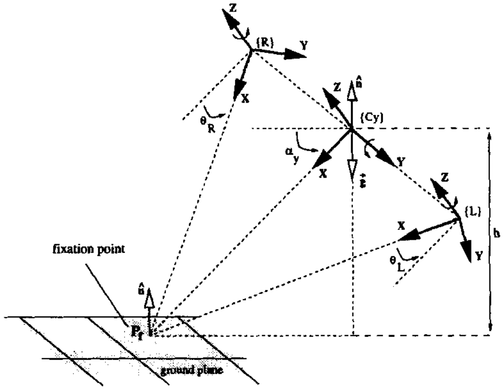


Figure 13. Ground plane point fixated. The point $\vec{P}_f$ in the ground plane is visualised by the active vision system. The geometry of this visualisation corresponds to a state, named visual fixation.

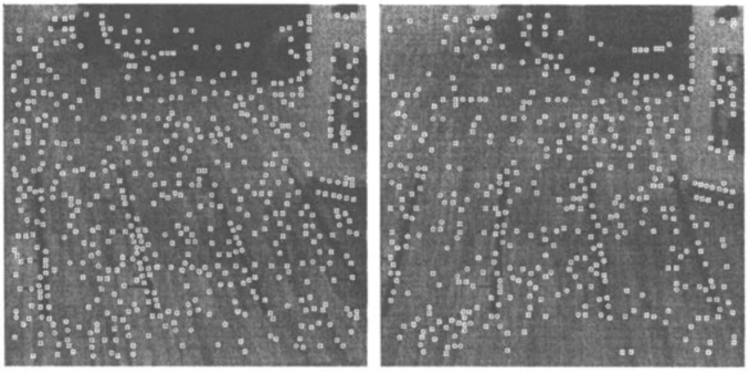


Figure 14. Stereo images with a set of initial points.

Figure 14 shows a pair of stereo images where fixation was obtained for a ground plane point. In this example α_x is null and $\alpha_y = 16.05^o$, $\theta = 2.88^o$ and $b = 29.6cm$. Making $h = b.\cot(\theta)sin(\alpha_y)/2 \simeq 81,3cm$. The points shown in figure 14 were interest points obtained using SUSAN [26] corner detector. These points are tested by an algorithm that classifies them as belong or not, to the ground plane. Any point in the ground plane verifies a geometric constraint that could between established between two stereo images, easily obtained from the stereo vision geometry [27]. The ground plane is thus determined. Figure 15 shows the matched ground plane points of interest.

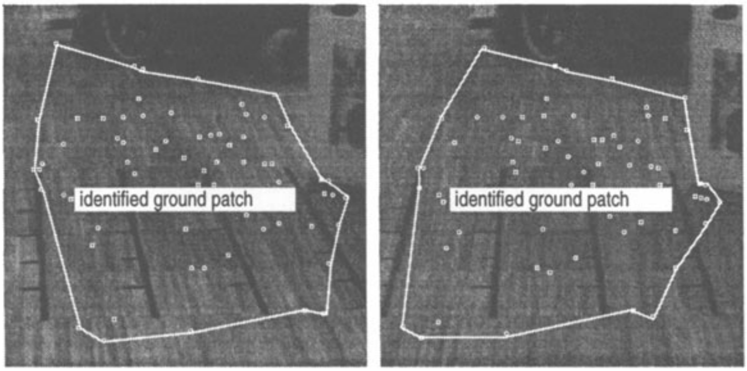


Figure 15. Detected ground points.

6. Range Sensors

Range sensors in mobile robots are useful for navigation in unstructured and unknown environments, allowing it to avoid obstacles, detect landmarks or identify navigable routes.

Humans use stereo vision for range sensing and environment perception, but those kinds of techniques are not very adequate for real time control of mobile platforms because they are computing intensive and unreliable.

There are several techniques that can be used for mobile robotics range sensing, namely: magnetic, inductive, capacitive, ultrasound, microwave and optical techniques [3]. Magnetic range sensors can only be used to detect surfaces that generate magnetic fields so its utilization on mobile robotics is very limited. Inductive sensors can be used to measure distance to metallic surfaces, but its range is very short (more or less the diameter of the sensor coil) and its response depends on surface magnetic and conductive properties. On a similar way, capacitive sensors can be used to measure distances up to some centimetres to dielectric surfaces, but its response also depends from the surface dielectric properties.

The most commonly used range sensors in mobile robotics are ultrasound sensors and optical range sensors. Ultrasound sensors are inexpensive and can measure distances up to several meters. Active optical range sensors based on the projection of optical radiation onto a scene can give fast and accurate results. The following section describes the main characteristics of ultrasound sensors and section 6.2 describes the most used optical range sensing methods for mobile robotics.

6.1. Ultrasound Sensors

Microwave and ultrasound sensors measure the distance based on the round-trip time of an energy wave between the sensor and a surface. Some ultrasound systems use two transducers, one to transmit and the other to receive the

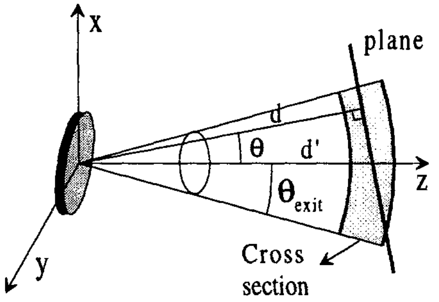


Figure 16. Ultrasonic TOF sensor model.

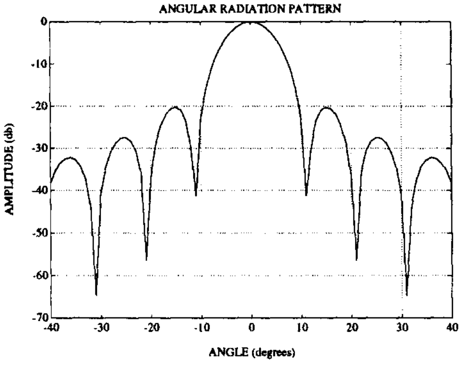


Figure 17. Beam pattern of a Polaroid ultrasonic module.

returned wave.

The propagation velocity (v) of an acoustic wave is given by;

$$v = \sqrt{\frac{K_m}{\rho}} \tag{9}$$

where K_m is the modulus of elasticity and ρ is the density of the medium. The relatively low velocity of sound in the air ($331.6 + 0.6 \times {}^{o}C\ m/s$), makes possible the ranging with just one transducer that acts both as a transmitter and as a receiver. The transducer emits a short pulse of a longitudinal wave in the ultrasonic spectrum (typically from 20 to 200 kHz). If an object intercepts the acoustic field and reflects enough energy to the receiver, the system will provide the range to the object. The maximum detection range depends on the emitted power, on the target cross-sectional area, reflectivity and orientation.

Although these sensors are very independent from surface characteristics, they are affected by the conditions of the propagating medium, namely temperature and humidity. These sensors are very linear, but they have some uncertainty that comes from the Time-Of-Flight (TOF) model. This model predicts that an echo comes from a curved shape volume, so that there will be uncertainty related to the divergence angle, both in depth and in orientation (see Figure 16). The divergence angle is a function of the transducer radius and resonant frequency, the larger the radius and the frequency, the narrow the divergence angle.

The most popular ultrasonic ranging systems are the Polaroid modules. These modules measure distances from 15 *cm* to about 10 *m* and have an angular beam width of about 12 degrees (see Figure 17). Because ultrasound sensors use acoustic waves, they suffer from low spatial resolution, crosstalk, depth uncertainty, errors due to multiple specular reflections, and low acquisition rates. In spite of these problems, their low cost and easy interface makes them some of the most used range sensors for indoors mobile robotics. Outdoor environments pose some additional problems like noise sources, dust and moisture.

6.2. Optical Range Sensors

Optical range sensors are particularly attractive for mobile robotics because they offer real-time accurate measurements with very high spatial resolution. These good properties are possible through the use of good quality optical components, namely coherent light sources (e.g. laser diodes) and very sensitive and high-resolution optical detectors (e.g. CCD cameras and avalanche photodiodes).

This section presents some of the most used optical range sensing methods in robotics, namely intensity reflection, triangulation, telemetry and lens focusing. Complementing surveys on optical range sensors can be found in the references [28, 3, 29, 30].

6.2.1. Reflective

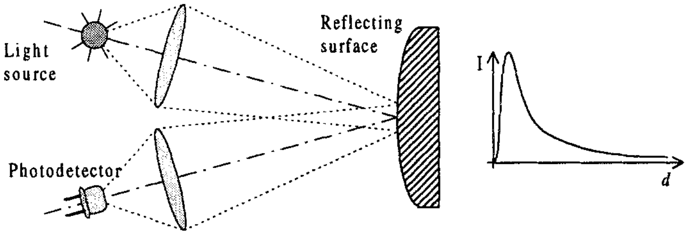


Figure 18. Reflective sensor principle. Some of the light emitted by the source is reflected on the surface and captured by the photodetector. The amount of captured optical light (I) depends on the distance (d) between the surface and the system.

Return signal intensity sensors are composed by a light emitter and a photodetector whose optical axes can be parallel for long range detection or convergent for shorter range detection (see Figure 18). These sensors measure the distance to an object by the amplitude of light reflected from its surface. The amount of detected light reflected from the object surface can be expressed by the following equation:

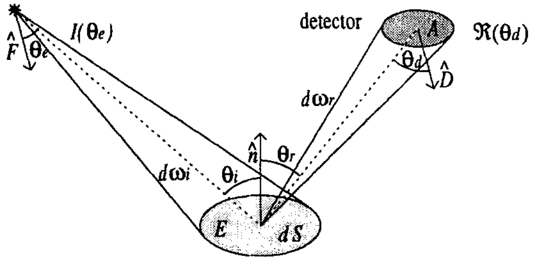


Figure 19. Reflection geometry used to calculate the detected intensity relative to an incremental surface patch.

$$\Phi = \int_S d\phi_r \tag{10}$$

where $d\phi_r$ represents the flux received from an incremental object surface patch (see Figure 19). This flux can be expressed by:

$$d\phi_r = I(\theta_e) \cdot d\omega_i \cdot R \cdot d\omega_r \cdot \Re(\theta_d) \tag{11}$$

where $I(\theta_e)$ is the emitted intensity of light from the solid angle $d\omega_i$, R is the function that characterises the reflectivity pattern of the surface, and $\Re(\theta_d)$ represents the acceptance of the photodetector at an off-axis angle θ_d. All common surfaces have a specular and a lambertian reflective component, so that using a simple model, like Phong's model, the following expression for the surface reflectivity can be obtained [31]:

$$R = K_s.\delta(\theta_r - \theta_i) + K_d.\cos(\theta_r) \tag{12}$$

where K_s and K_d are the coefficients for the specular and lambertian components respectively.

For a sensor with parallel emitter and detector optical axes, the detected intensity varies approximately with the bi-quadratic inverse of the distance [32]. Because the reflected intensity depends heavily from the surface optical characteristics and from its orientation, these sensors suffer from low repeatability. The most commonly found commercial devices are not intended to be used as full range measuring devices, but as simple non-contact presence detectors.

Several manufacturers (e.g. SunX, Banner, Honeywell) provide photoelectric detectors that detect surfaces up to about 1 meter. A common strategy to eliminate the influence of background light, is the utilization of modulated infrared energy and the appropriated optical and electrical filters.

Reflective sensors can be easily homebuilt with increased functionality around a LED and a photodiode. For example some researchers use the IR proximity system to establish data links between a community of mobile robots.

Some mobile robots use IR proximity sensors for short range, narrow beam sensing, together with ultrasound range sensors for medium range, wide beam

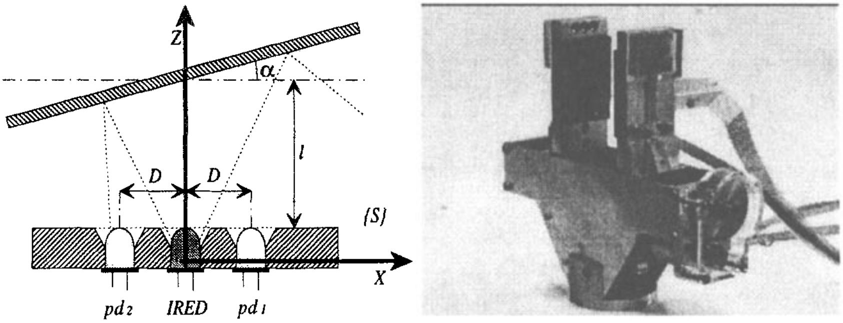


Figure 20. a) Geometry and elements of the proximity sensor. In the picture, pd_x is photo-detector x and *IRED* is infrared light emitting diode. S represents the co-ordinated system associated to the sensor. b) *Gripper* with a reflective sensor on the extremity of each finger.

sensing. The fusion of both kinds of information improves the results [33, 34] obtained with just one type of sensor.

2-D Reflective Sensor

The ISR Reflex sensor is a small reflective sensor designed to be used on a parallel jaw electrical gripper. This sensor is composed by two photodiodes on opposite sides of a light emitting diode (see Figure 20a). This configuration allows measuring the orientation to a planar homogeneous surface by the difference between the detected intensities in the two photodiodes.

A more detailed description and the sensor modelling can be reached in [32].

Two prototypes of this sensor were integrated on a parallel jaw gripper presented in Figure 20b) for object detection and pre-prehension gripper control. The presence of objects between the gripper fingers is detected by occlusion between the emitter of a finger and the two detectors of the other finger. With this sensorial information, the gripper can align we the object to be grasped and made a smooth transition between position control and contact force control.

When used with planar homogeneous calibrated surfaces, the sensor can measure distances with an accuracy of 0.1 *mm* on a range from 5 to 100 *mm*. The orientation accuracy is about 0.1°.

A cooperative array of emitters and detectors around a mobile platform can be used in order to estimate surface orientation and profiles based on the several detected intensities. This kind of sensor can help for example on docking tasks.

6.2.2. Telemetry

Laser radar sensors or laser range-finders measure the distance d between the sensor and a target surface based on the round-trip time Δt of a laser beam (see Figure 21). Considering v the velocity of the propagated wave on the medium, the distance d can be calculated by the following formula:

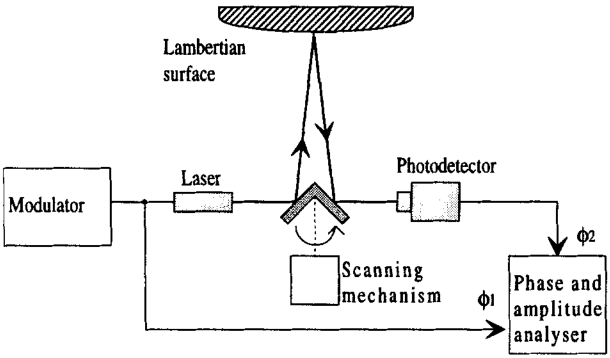


Figure 21. Laser radar. The distance is proportional to the round-trip time of a light pulse between the measuring system and a target surface.

$$2d = v \cdot \Delta t \tag{13}$$

Although these sensors can use three different methods: pulse based time-of-flight (TOF), amplitude modulated (AMCW) and frequency modulated (FMCW), the first two are the most common ones. Pulsed rangefinders emit a short pulse of light and count the time to receive the reflected signal whilst AMCW range finders send an amplitude modulated continuous wave and use the phase shift between the emitted and the received wave to calculate the distance. Range accuracy of AMCW sensors depends upon the modulation wavelength and the accuracy with which the phase shift is measured. For round-trip distances longer than the modulation wavelength there will be ambiguity on the phase shift measurement.

Laser range-finders can measure not only the distance but also the amplitude of the reflected signal (intensity). The fusion of range and intensity images provided by scanning systems, can be helpful for image recognition tasks. These systems are fast, linear and very accurate over a long range of distances, but they are also the most expensive range sensors [35, 36, 37]. Table 1 presents the main characteristics of some currently available scanning systems [38, 39].

6.2.3. Triangulation

Triangulation sensors are based on the following trigonometric principle: *if the length of one side along with two interior angles of a triangle are known, then we can determine the length of the two remaining sides along with the other angle.*

An optical triangulation system can be either passive (use only the ambient light of the scene) or active (use an energy source to illuminate the target). Passive triangulation or stereoscopic systems use two cameras oriented to the same scene. The lens central points of each camera along with each point on the scene, define triangles with a fixed baseline (the distance between the central point of each camera lens) and variable interior angles. If the focal distance of each camera is known, these two interior angles can be calculated by the

Manufacturer	Model	Modulation	field of view	pixels	Accuracy	Min. dist.	Max. distance
Acuity Res.	AccuRange 4000	–	360°/line	2000	2.5mm	0	50 $feet$
ALST[a]	ELRF-3-M	Pulsed/1Hz	–	1	2 m	50 m	30 km
ERIM	Adaptive Suspension	AMCW/14MHz	80° Az 60° El	128 × 128	3.9 cm	–	19.5 m
ERIM	Autonomous Land	AMCW/7MHz	80° Az 30° El	256 × 64	7.8 cm	–	39 m
ERIM	Intelligent Task	AMCW/720MHz	38° Az 38° El	256 × 256	2.5 mm	–	0.91 m
ERIM	US Postal Service	AMCW/700MHz	35° Az 35° El	300 × 300	0.05 mm	–	1 m
HDOS	Imaging Laser Radar	AMCW/15MHz	80° Az 60° El	~ 3000	0.15 m	–	460 m
IMRA	Laser Radar Sensor	–	90° /line	128	< 1 m	–	25 m
Leica	ODIN	–	6°sr	–	0.1 m	–	150 m
Odetics	GMS	AMCW/16MHz	60° Az 60° El	128 × 128	18 mm	–	> 10 m
Perceptron	LASAR	AMCW/varies	60° Az 60° El	1024 × 1024	2 mm	–	2-40 m
Riegl	LRI 20x5	Pulsed	5° Az 20° El	10.000	±3 cm	2 m	80 m
Riegl	LMS-Q140-80	Pulsed/12kHz	80°/line	400	±2.5 cm	2 m	350 m
Riegl	LD90-3300	Pulsed	–	1	±5 cm	3 m	400 m
Sandia	Scannerless R. Imager	AMCW/5.5MHz	ajustable	256 × 256	0.3 m	–	varies
Schwartz EO	LRF-200	Pulsed	360°/line	–	±30 cm	1 m	100 m
Schwartz EO	LADAR	Pulsed/9KHz	4° Az 10° El	25 × 65	0.3 m	–	500 m
Schwartz EO	Treesense	Pulsed/18KHz	2 × 90°	–	3 in	–	100 $feet$
Schwartz EO	Autosense II	Pulsed/15KHz	10° Az 30° El	30/line	3 in	2 $feet$	50 $feet$
Sick Optic	PLS100	–	180°/line		7 cm	–	50 m

Table 1. Main characteristics of some commercial laser rangefinders.

[a] Very long range. Used normally on military applications.

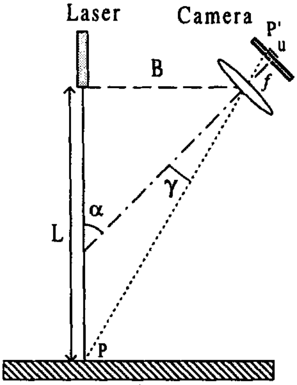


Figure 22. Triangulation system based on a laser beam and some kind of an imaging camera.

position of each point on both images. The main problem of these systems is due to the identification of corresponding points on both images (feature matching). To obtain a solution for this problem, active triangulation systems replace the second camera by a light source that projects a pattern of light on to the scene. The simplest case of such a sensor, like the one represented in Figure 22, use a laser beam and a one-dimensional camera. The distance (L) between the sensor and the surface can be measured by the image position (u) of the bright spot formed on the intersection point (P) between the laser beam and the surface.

$$L = \frac{B}{tan(\alpha - \gamma)} \tag{14}$$

Where B is the distance between the central point of the lens and the laser beam (baseline) and α is the angle between the camera optical axis and the laser beam. The angle γ is the only unknown value in the equation, but it can be calculated using the position (u) of the imaged spot (provided that the value of the focal distance f is known).

$$\gamma = \arctan\left(\frac{u}{f}\right) \tag{15}$$

If it is required to obtain a range image of a scene, the laser beam can be scanned or one of several techniques based on the projection of structured light patterns, like light strips [40], grids [41, 42, 43, 44], binary coded patterns [45, 46], color coded stripes [47, 48, 49], or random textures [50] can be used. Although these techniques improve the performance of the range imaging system, they may also present some ambiguity problems [51, 52].

Triangulation systems present a good price/performance relation, they are pretty accurate and can measure distances up to several meters. The accuracy of these systems falls with the distance, but usually this is not a great problem on mobile robotics because high accuracy is only required close to the objects,

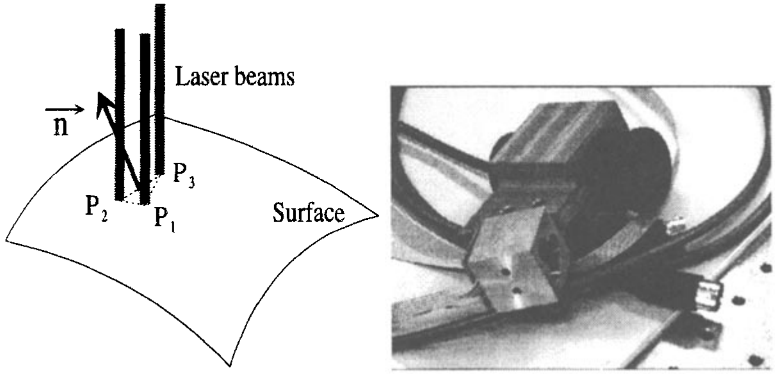


Figure 23. a) Distance and orientation measuring with three laser beams and a Position Sensitive Detector. b) Prototype of the Opto3D measuring head.

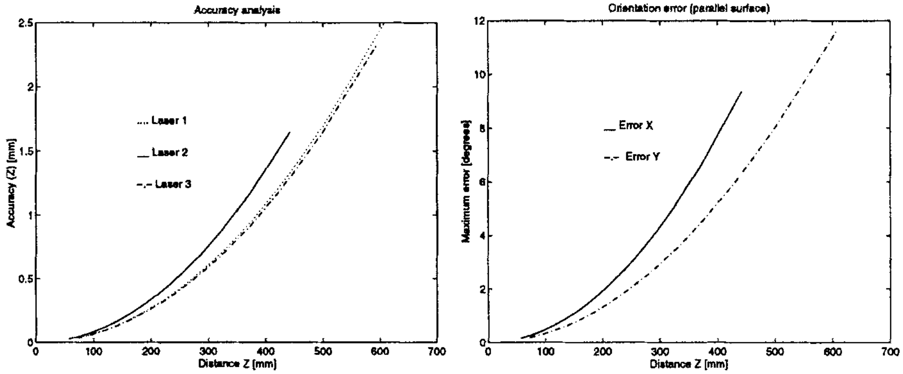


Figure 24. a) Distance accuracy vs. range. b) Angular accuracy for a perpendicular surface.

otherwise it is enough to detect its presence. The main problems of triangulation systems are the possibility of occlusion, and measures on specular surfaces that can blind the sensor or give rise to wrong measures because of multiple reflections [53, 54, 55, 56, 57, 58].

Opto3D

The Opto3D system is a triangulation sensor that uses a PSD[4] camera and three laser beams. Measuring the coordinates of the three intersection points P_1, P_2 and P_3 (see Figure 23a), the sensor can calculate the orientation $\vec{n}$ of the surface by the following expression:

$$\vec{n} = \vec{P_1P_2} \times \vec{P_1P_3} \tag{16}$$

The Opto3D sensor can measure distances up to 75 *cm* with accuracies from 0.05 to 2 *mm* (see Figure 24) [54, 53]. Like every triangulation sensor, the

[4] Position Sensitive Detector

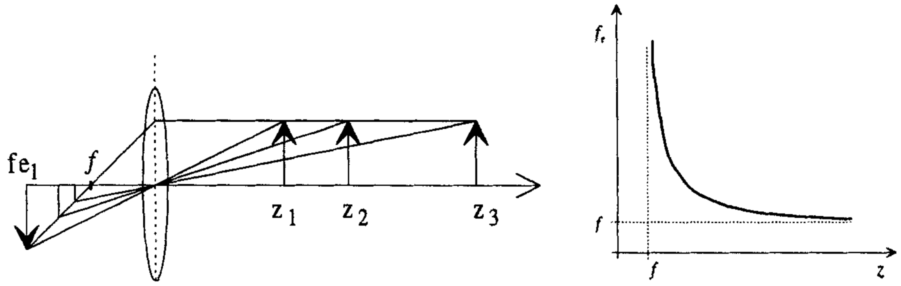


Figure 25. Using the Gauss lens law, it is possible to extract range information from the effective focal distance of an image.

accuracy degrades with the distance. This sensor can measure orientation on a broad range with an accuracy better than 0.1°, and the maximum orientation depends on the reflective properties of the surface (usually only a little amount of light can be detected from light beams that follow over almost tangential surfaces).

6.2.4. Lens Focusing

Focus range sensing relies on Gauss thin lens law (equation 17). If the focal distance (f) of a lens and the actual distance between the focused image plane and the lens center (f_e) is known, the distance (z) between the lens and the imaged object can be calculated using the following equation:

$$\frac{1}{f_e} = \frac{1}{f} - \frac{1}{z} \tag{17}$$

The main techniques exploring this law are **range from focus** (adjust the focal distance f_e till the image is on best focus) and **range from defocus** (determine range from image blur).

These techniques require high frequency textures, otherwise a focused image will look similar to a defocused one. To have some accuracy, it is fundamental to have very precise mathematical models of the image formation process and very precise imaging systems [59].

Image blurring can be caused by the image process or by the scene itself, so depth from defocus technique, requires the processing of at least two images of an object (which may or may not be focused) acquired with different but known camera parameters to determine the depth. A recent system provides the required high-frequency texture projecting an illumination pattern via the same optical path used to acquire the images. This system provide real-time (30 Hz) depth images (512 × 480) with an accuracy of approximately 0.2% [60].

The accuracy of focus range systems is usually worse than stereoscopic ones. Depth from focus systems have a typical accuracy of 1/1000 and depth from defocus systems 1/200 [59]. The main advantage these methods is the lack of correspondence problem (feature matching).

7. Conclusions

The article described several sensor technologies, which allow an improved estimation of the robot position as well as measurements about the robot surroundings by range sensing. Navigation plays an important role in all mobile robot activities and tasks. The integration of inertial systems with other sensors in autonomous systems opens a new field for the development of a substantial number of applications. Range sensors make possible to reconstruct the structure of the environment, avoid static and dynamic obstacles, build maps and find landmarks.

References

[1] Altschuler M, et al. 1962 Introduction. In: Pitman G R (ed), *Inertial Guidance*, John Wiley & Sons, pp 1–15

[2] Feng L, Borenstein J, Everett H December 1994 Where am I? - Sensors and Methods for Autonomous Mobile Robot Positioning. Tech. Rep. UM-MEAM-94-21, University of Michigan

[3] Everett H 1995 *Sensors for Mobile Robotics*. A.K. Peters, ISBN 1-56881-048-2

[4] Viéville T, Faugeras O 1989 Computation of Inertial Information on a Robot. In: Miura H, Arimoto S (eds), *Fifth International Symposium on Robotics Research*, MIT-Press, pp 57–65

[5] Collinson R 1996 *Introduction to Avionics*. Chapman & Hall, ISBN 0-412-48250-9

[6] Ausman J S 1962 *Theory of Inertial Sensing Devices*, George R. Pitman (ed.), John Wiley & Sons, pp 72–91

[7] Slater J M 1962 *Principles of Operation of Inertial Sensing Devices*, George R. Pitman (ed.), John Wiley & Sons, pp 47–71

[8] Kuritsky M M, Goldstein M S 1990 *Inertial Navigation*, T. Lozano-Perez (ed), Springer-Verlag New York, pp 96–116

[9] Allen H V, Terry S C, Knutti J W September 1989 Understanding Silicon Accelerometers. *Sensors*

[10] ICSensors January 1988 *Silicon Accelerometers*. Technical Note TN-008

[11] Summit Instruments September 1994 *34100A Theory of Operation*. Technical Note 402

[12] Barshan B, Durrant-Whyte H June 1995 Inertial Navigation Systems for Mobile Robots. *IEEE Transactions on Robotics and Automation*, 11:328–342

[13] Komoriya K, Oyama E 1994 Position Estimation of a Mobile Robot Using Optical Fiber Gyroscope (OFG). In: *Proceedings of the 1994 IEEE International Conference on Intelligent Robots and Systems*, pp 143–149

[14] Murata 1991 *Piezoelectric Vibrating Gyroscope GYROSTAR*. Cat. No. S34E-1

[15] Systron Donner Inertial Division 1995 *GyroChip*. Product Literature

[16] Peters T May 1986 Automobile Navigation Using a Magnetic Flux-Gate Compass. *IEEE Transactions on Vehicular Technology*, 35:41–47

[17] KVH Industries, Inc May 1993 *C100 Compass Engine Technical Manual*. Revision g edn., KVH Part No. 54-0044

[18] Getting I A December 1993 The Global Positioning System. *IEEE Spectrum*, pp 236–247

[19] Kaplan E D 1996 *Understanding GPS: Principles and Applications*. Artech House, ISBN 0-89006-793-7

[20] Kelly A May 1994 Modern Inertial and Satellite Navigation Systems. Tech. Rep. CMU-RI-TR-94-15, Carnegie Mellon University

[21] Dana P H 1997 Global Positioning System Overview. Tech. Rep. CMU-RI-TR-94-15, Department of Geography, University of Texas at Austin, http://www.utexas.edu/depts/grg/gcraft/notes/gps/gps.html

[22] Carpenter H 1988 *Movements of the Eyes*. London Pion Limited, 2nd edn., ISBN 0-85086-109-8

[23] Lobo J, Lucas P, Dias J, de Almeida A T July 1995 Inertial Navigation System for Mobile Land Vehicles. In: *Proceedings of the 1995 IEEE International Symposium on Industrial Electronics*, Athens, Greece, pp 843–848

[24] Dias J, Paredes C, Fonseca I, de Almeida A T 1995 Simulating Pursuit with Machines. In: *Proceedings of the 1995 IEEE Conference on Robotics and Automation*, Japan, pp 472–477

[25] Paredes C, Dias J, de Almeida A T September 1996 Detecting Movements Using Fixation. In: *Proceedings of the 2nd Portuguese Conference on Automation and Control*, Oporto, Portugal, pp 741–746

[26] Smith S, Brady J May 1997 SUSAN - a new approach to low level image processing. *Int Journal of Computer Vision*, pp 45–78

[27] Silva A, Menezes P, Dias J 1997 Avoiding Obstacles Using a Connectionist Network. In: *Proceedings of the 1997 IEEE International Conference on Intelligent Robots and Systems (IROS'97)*, pp 1236–1242

[28] Besl P 1988 Active Optical Range Imaging Sensors. *Machine Vision and Applications*, 1:127–152

[29] Jarvis R March 1983 A perspective on Range Finding Techniques for Computer Vision. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-5:122–139

[30] Volpe R, Ivlev R 1994 A Survey and Experimental Evaluation of Proximity Sensors for Space Robotics. In: *Proc. IEEE Conf. on Robotics and Automation*, pp 3466–3473

[31] Phong B June 1975 Illumination for computer generated pictures. *Commun of the ACM*, 18:311–317

[32] Marques L, Castro D, Nunes U, de Almeida A 1996 Optoelectronic Proximity Sensor for Robotics Applications. In: *Proc. IEEE 8'th Mediterranean Electrotechnical Conf.*, pp 1351–1354

[33] Flynn A 1985 Redundant Sensors for Mobile Robot Navigation. Tech. Rep. 859, MIT Artificial Intelligence Laboratory

[34] Flynn A December 1988 Combining Sonar and Infrared Sensors for Mobile Robot Navigation. *International Journal of Robotics Research*, 7:5–14

[35] Duda R, Nitzan D, Barret P July 1979 Use of Range and Reflectance Data to Find Planar Surface Regions. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-1:259–271

[36] Nitzan D, Brain A, Duda R February 1977 The Measurement and Use of Registered Reflectance and Range Data in Scene Analysis. *Proceedings of the IEEE*, 65:206–220

[37] Jarvis R September 1983 A Laser Time-of-Flight Range Scanner for Robotic Vision. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-5:505–512

[38] Carmer D, Peterson L February 1996 Laser Radar in Robotics. *Proceedings of the IEEE*, 84:299–320

[39] Conrad D, Sampson R 1990 3D Range Imaging Sensors. In: Henderson T (ed), *Traditional and Non-Traditional Robotic Sensors*, Springer-Verlag, Berlin, vol. F63 of *NATO ASI Series*, pp 35–48

[40] Will P, Pennington K June 1972 Grid Coding: A Novel Technique for Image Processing. *Proceedings of the IEEE*, 60:669–680

[41] Stockman G, Chen S, Hu G, Shrikhande N June 1988 Sensing and Recognition of Rigid Objects Using Structured Light. *IEEE Control Systems Magazine*, 8:14–22

[42] Dunn S, Keizer R, Yu J November 1989 Measuring the Area and Volume of the Human Body with Structured Light. *IEEE Trans Systems Man and Cybernetics*, SMC-19:1350–1364

[43] Wang Y, Mitiche A, Aggarwal J January 1987 Computation of Surface Orientation and Structure of Objects Using Grid Coding. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-9:129–137

[44] Wang Y January 1991 Characterizing Three-Dimensional Surface Structures from Visual Images. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-13:52–60

[45] Altschuler M, et al. 1987 Robot Vision by Encoded Light Beams. In: Kanade T (ed), *Three-dimensional machine vision*, Kluwer Academic Publishers, Boston, pp 97–149

[46] Vuylsteke P, Oosterlinck A Feb 1990 Range Image Acquisitionwith a Single Binary-Encoded Light Pattern. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-12:148–164

[47] Kak A, Boyer K, Safranek R, Yang H 1986 Knowledge-Based Stereo and Structured Light for 3-D Robot Vision. In: Rosenfeld A (ed), *Techniques for 3-D Machine Perception*, Elsevier Science Publishers, pp 185–218

[48] Boyer K, Kak A January 1987 Color-Encoded Structured Light for Rapid Active Ranging. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-9:14–28

[49] Wust C, Capson D 1991 Surface Profile Measurement Using Color Frinje Projection. *Machine Vision and Applications*, 4:193–203

[50] Maruyama, Abe S Jun 1993 Range Sensing by Projecting Multiple Slits with Random Cuts. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-15:647–651

[51] Vuylsteke P, Price C, Oosterlinck A 1990 Image Sensors for Real-Time 3D Acquisition: Part 1. In: Henderson T (ed), *Traditional and Non-Traditional Robotic Sensors*, Springer-Verlag, Berlin, vol. F63 of *NATO ASI Series*, pp 187–210

[52] Mouaddib E, Battle J, Salvi J 1997 Recent Progress in Structured Light in order to Solve the Correspondence Problem in Stereo Vision. In: *Proc. IEEE Conf. on Robotics and Automation*, pp 130–136

[53] Marques L 1997 Desenvolvimento de Sensores Optoelectrónicos para Aplicações de Robótica. Master's thesis, DEE, FCT - Universidade de Coimbra

[54] Marques L, Moita F, Nunes U, de Almeida A 1994 3D Laser-Based Sensor for Robotics. In: *Proc. IEEE 7'th Mediterranean Electro. Conf.*, pp 1328–1331

[55] Lee S 1992 Distributed Optical Proximity Sensor System: HexEYE. In: *Proc. IEEE Conf. on Robotics and Automation*, pp 1567–1572

[56] Lee S, Desai J 1995 Implementation and Evaluation of HexEYE: A Distributed Optical Proximity Sensor System. In: *Proc. IEEE Conf. on Robotics and Automation*, pp 2353–2360

[57] Kanade T, Sommer T 1983 An Optical Proximity Sensor for Measuring Surface Position and Orientation for Robot Manipulation. In: *Proc. 3rd International*

Conference on Robot Vision and Sensory Controls, pp 667–674

[58] Kanade T, Fuhrman M 1987 A Noncontact Optical Proximity Sensor for Measuring Surface Shape. In: Kanade T (ed), *Three-dimensional machine vision*, Kluwer Academic Publishers, Boston, pp 151–192

[59] Xiong Y, Shafer S A 1993 Depth from Focusing and Defocusing. Tech. Rep. 93-07, The Robotics Institute - Carnegie Mellon University

[60] Nayar S, Watanabe M, Noguchi M Dec 1996 Real-Time Focus Range Sensor. *IEEE Trans Pattern Analysis and Machine Intelligence*, PAMI-18:1186–1198

Application of Odor Sensors in Mobile Robotics

Lino Marques and Aníbal T. de Almeida
Institute of Systems and Robotics
Department of Electrical Engineering - University of Coimbra
3030 Coimbra, Portugal
{lino, adealmeida}@isr.uc.pt

Abstract:

Animals that have a rather small number of neurons, like insects, display a diversity of instinctive behaviours strictly correlated with particular sensory information. The diversity of behaviors observed in insects has been shaped by millions of years of biological evolution, so that their strategies must be efficient and adaptive to circumstances which change every moment. Many insects use olfaction as a navigation aid for some vital tasks as searching for sources of food, a sexual partner or a good place for oviposition.

This paper discusses the utilisation of olfactive information as a navigational aid in mobile robots. The main technologies used for chemical sensing and their current utilisation on robotics is presented. The article concludes giving clues for potential utilisation of electronic noses associated to mobile robots.

1. Introduction

Although it is rather common to find robots with sensors that mimic the animal world (particularly the man senses), sensors for taste and smell (chemical sensors) are by far the least found on robotics. The reasons for that are not just the reduced importance of those senses in human motion, but it is also a consequence of the long way for chemical sensors to evolve in order to become similar to their biological counterparts.

Traditional systems for analysis of the gases concentration in the air were bulky, fragile and extremely expensive (spectroscopic systems). The least expensive options based on catalytic or metal oxide sensors had little accuracy, reduced selectivity and short lifetime. Several recent advances in these technologies and the development of new ones, like conducting polymers and optical fibres, lead to the appearance of a new generation of miniature and low cost chemical sensors that can be used to build small and inexpensive electronic noses.

Robots can take advantage from an electronic nose when they need to carry out some chemically related tasks, such as cleaning and finding gas leaks, or when they want to implement a set of animal-like instinctive behaviors based on olfactive sensing.

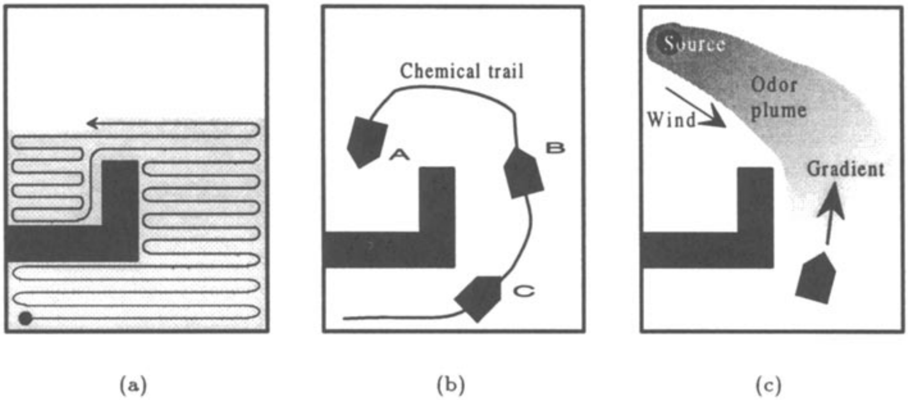


Figure 1. There are several animal behaviors based on olfactory sensing that can be implemented on mobile robots, namelly the following: (a) Repellent behaviors, where a robot goes away from an odor. This behavior can be used on a cleaning robot to detect the pavement already cleaned. (b, c) Attractive behaviors, where a robot can follow a chemical trail or find an odor source.

There are several animal behaviors based on olfactory sensing that can be implemented on mobile robots. Among those behaviors we can emphasize:

1. Find the source of an odor (several animals).
2. Lay down a track to find the way back (ants).
3. Go away from an odor (honeybees).
4. Mark zones of influence with odors.

Small animals, like some insects, can successfully move in changing unstructured environments, thanks to a set of simple instinctive behaviors based on chemically sensed information. Those behaviors, although simple, are very effective because they result from millions of years of biological evolution.

There are two chemically related strategies that can be used for mobile robotics navigation. The searching strategy, where the robot looks for an odor source or a chemical trail, and the repellent strategy, where the robot goes away from an odor.

Insects frequently use the first strategy when they look for food or for a sexual partner. For example, when a male silkworm moth detects the odor liberated by a receptive female it starts a zigzagging search algorithm until it finds the odor source [1, 2].

Ants establish and maintain an odor trail between the source of food and their nest. All that they have to do in order to move the food to the nest, is to follow the laid chemical trail [3].

The second strategy, to go away from a specific odor, is used by several animals to mark their territory with odors in order to keep other animals away.

Honeybees also use this strategy, but to improve their efficiency when gathering nectar. They mark the visited flowers with an odor that remains active while the flower creates more nectar. This way they do not need to land on flowers that have no nectar.

Although simple, these kinds of behaviors were successfully used on nature during millions of years. Its implementation on mobile robots can improve their performance without the need for heavy control algorithms. For example, a cleaning robot can use chemical sensorial information to know when a floor is already cleaned (see Figure 1a). A security robot can mark its path with a volatile chemical in order to know when it has recently passed somewhere. In this way, a set of security robots can co-operatively patrol an area without centralised control among them.

When large amounts of material must be transferred from a place to another, an intelligent mobile robot can be used to mark the path with a chemical mark while simple AGV-like transport robots equipped with chemical sensors can follow that path (see Figure 1b).

Hydrogen is among the most widely used gases in industry. A 1995 NASA report refers that undetected leaks are the largest cause of industrial hydrogen accidents. A robot with a sensitive electronic nose could be used to patrol industrial plants and report any abnormal gas concentration on the air. These reports could be a good help for the factory maintenance staff to discover leaks, to detect fires or overheated equipment, and damaged stored material. Such a robot could even use the nose information as a localisation instrument. Usually there are places with typical odors, that could be used as landmarks.

2. Chemical gas sensors

The increasing need for control of industrial processes and environment monitoring, pushed the research in new chemical sensing technologies. The main gas categories to be monitored in common applications are:

1. Oxygen, for the control of combustion processes.

2. Flammable gases in order to protect against fire or explosion.

3. Toxic gases for environmental monitoring.

This section presents several kinds of sensors that can be used to detect chemical gases in the environment. Some more detailed surveys can be found in the references [4, 5, 6, 7, 8, 9, 10].

2.1. Solyd electrolyte sensors

As voltaic cells, solyd electrolyte sensors are based on the voltage generated in the interface between phases having different concentrations.

These sensors have three components, two metallic electrodes (one of which coated with a catalyst), an electrolyte and a membrane. When an electroactive gas diffuses through the membrane and reacts at the electrolyte-catalyst interface, it generates a current proportional to the gas concentration.

These sensors can detect gases in the *ppm* range, but their lifetime depends on the exposition to the reacting gas. Millions of vehicles in the entire world use this type of sensor to monitor the exhausted gases and minimize the toxic emissions.

2.2. Thermal-chemical sensors

Thermal-chemical sensors detect the heat released or absorbed, ΔE_h, when a reaction takes place. This change in enthalpy causes a change in temperature, ΔT, which can be monitored. In the ideal case the system should be thermally isolated. In practice there are heat losses through convection, conduction and radiation, that affect the detected temperature change [11]. The main application of these sensors is the monitoring of combustible gases.

The pellistor is the most common thermal-chemical sensor (other thermal sensors are based on either on thermistors or on thermopiles). This sensor is composed by a platinum coil buried in a probe covered with a thin catalytic layer. The coil serves to heat the sensor to its operating temperature (about $500^\circ C$) and to detect the increase in temperature from the reaction with the gas. The coil resistance changes about $0.4\%/^\circ C$.

Pellistors are produced since the early 70's. They have a low price, but they feature high power dissipation (about 1 W), non-selective response, drift and sensitivity to humidity and temperature. These devices can be irreversibly poisoned by some contaminant vapors that shorten their life.

2.3. Gravimetric chemical sensors

When a chemical species interacts with the sensing material, it often results in a change in the total mass. This small change in mass can be measured by a microbalance using either a piezoelectric Bulk Acoustic Wave (**BAW**) oscillator or a Surface Acoustic Wave (**SAW**) device. These devices are composed by a peace of piezoelectric material (usually Quartz) coated with a thin film of a chemically selective absorbent material [12].

In these sensors the change in mass Δm is converted to a frequency shift Δf by an oscillator circuit.

$$\Delta f = k \cdot \Delta m \tag{1}$$

Where k is a constant. The performance of the chemical sensor depends on the frequency of operation and on the functionality of the chemically-sensitive coating.

SAW devices generally work at much higher frequencies than BAW devices, so for the same sensitivity, they can be made much smaller and less expensive. SAW sensors are also very attractive for the development of sensor arrays because this technology can be applied on two-dimensional structures [5]. These sensors can have a mass resolution down to 1 *pgram*.

2.4. Conducting polymers

Conducting polymers are plastics that change its electrical resistance as they adsorb or desorb specific chemicals. The adsorption of volatile chemicals depends on their polarity (charge) and spatial geometry of the material microstructure (size and shape). The better the fit, the greater the electrical change.

An elevated concentration of a poorly fitting chemical can have the same effect as a low concentration of a good fitting chemical [13, 14].

Conducting polymers are relatively new materials on chemical sensing (they appeared in the early 80's), but because of their good qualities (namely: excellent sensitivity - down to some *ppb*, low price and rapid response time at room temperatures), they have the potential to become the dominant chemical gas sensing technology in the near future.

2.5. IR spectroscopy

Because of their natural molecular vibration, all gases interfere and absorb light at specific wavelengths of the infrared spectrum. This property can be used to detect the concentration of different gases in the environment.

There are monochromatic systems tuned with narrow-band interference filters or laser light sources for a specific gas (like CO_2) and there are spectroscopic systems able to determine the concentration of several gases at once.

These sensors feature slow response, good linearity, low cross-sensitivity to other gases and fairly good accuracy, but they need frequent recalibration, are bulky and very expensive.

2.6. Optical fiber sensors

Optical fiber gas sensors are composed by a fiber bundle with its ends coated with a gas-sensitive fluorescent polymer. When light comes down the fiber, it excites the polymer, which emits at a longer wavelength to the detection system. The amount of returning fluorescent light is related with the concentration of the chemical species of interest at the fiber tips.

Tufts University uses an array of 19 fibers coated with Nile Red dye. To differentiate among various chemical gases, they correlate the amount of returned intensity and the fluorescence lifetime in each fiber [15].

2.7. Metal oxide sensors (MOS)

The adsorption of a gas onto the surface of a semiconducting material can produce a large change in its electrical resistivity. Although this effect was observed since the early 1950s, the non-reproducibility of the results was a major problem and the first commercial devices were only produced in 1968 [16, 17].

The most common metal oxide gas sensor is the Tagushi type. This sensor uses a thin layer of powder tin oxide (SnO_2). When the sensor is heated at a high temperature (usually 300 to $450°C$), some oxygen molecules from the air are adsorbed on the crystal surface and remove electrons from the SnO_2 grains (see Figure 2a). Because tin oxide is a type *n* semiconducting material, this process reduces the charge carrier density, increasing the grain-to-grain contact resistance and consequently increasing the electrical resistivity of the sensor. In the presence of a deoxidizing gas, the reducing reaction decreases the concentration of oxygen molecules in the crystal (see Figure 2b). This effect can be detected by a decrease in the sensor resistance. Typically, a reducing gas concentration of 100 *ppm* can change the resistance of the sensor by a factor of 10.

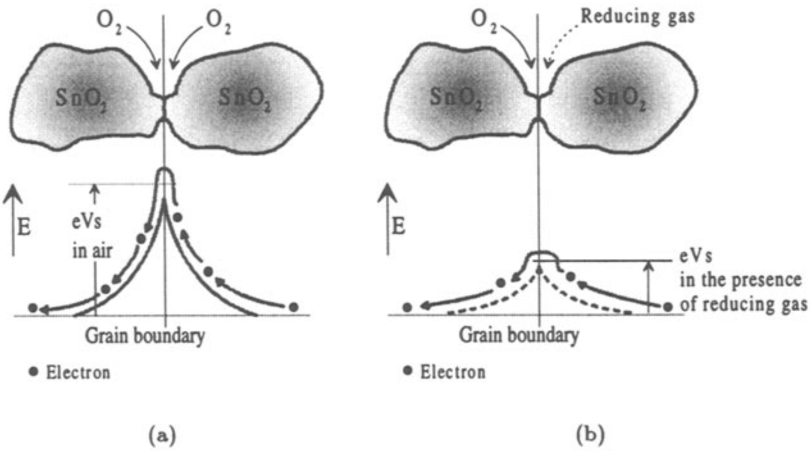


Figure 2. a) Model of inter-grain potential barrier in the absence of gases. b) Model of inter-grain potential barrier in the presence of gases (adapted from Figaro data sheets).

The relationship between sensor resistance and the concentration of the deoxidizing gas can be expressed by the following equation:

$$R = R_0(1 + A \cdot C)^{-\alpha} \tag{2}$$

Where R is resistance of the sensor in the presence of the gas, R_0 is the resistance in the air, A and α are constants and C is the concentration of the deoxidizing gas [18].

MOS sensors are simple and inexpensive gas sensitive resistors that can be used to detect a wide range of reducing gases. The sensitivity of each sensor to a gas depends on the oxide material, on its temperature, on the catalytic properties of the surface and on the humidity and oxygen concentration in the environment. The most common used material is tin oxide, but some researchers are trying other oxides (for example Ga_2O_3) with better stability, reproducibility, selectivity and insensitivity to environmental conditions [19, 20].

3. Electronic noses

Almost all the sensors described in the previous section suffer from some problems; the main ones are lack of selectivity and the inability to give the concentration of gas components. To overcome these problems, Dodd and Persaud proposed a device that tries to mimic the mammalian olfactory system [21]. This device, which they called **electronic nose**, incorporated three broadly-tuned tin oxide gas sensors and used pattern recognition algorithms to discriminate between chemically similar odors.

The two main components of an electronic nose are the chemical sensing

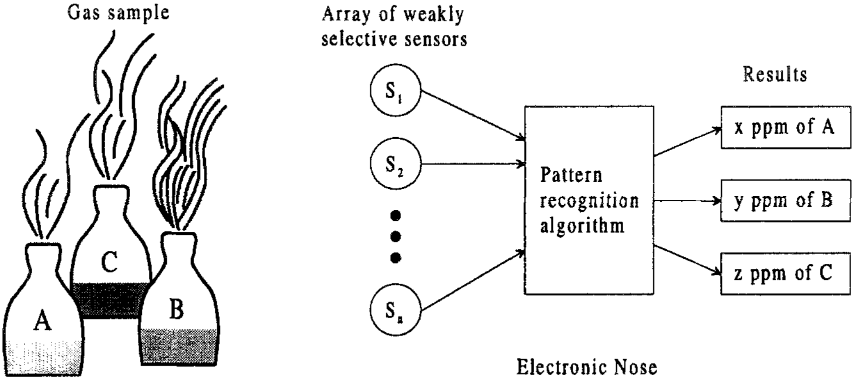


Figure 3. Representation of an electronic nose.

system and the pattern recognition system (see Figure 3). The sensing system is composed by a chemical sensor array, where each element measures a different property of the sensed gas. In this way each vapor presented to the sensing system produces a characteristic signature. The goal of the pattern recognition algorithm is to identify each component of the sensed gas based on the signals from each sensing element [22, 23, 24].

To operate properly, the electronic nose should first be calibrated. In the calibration process the system is placed inside a controllable and isolated environment, and its response is measured for different and known concentrations of the products to be detected.

There are two common methods used to control the concentration of the volatile products inside the isolated recipient. The **static method**, where a fixed volume of the product is injected inside the recipient [25], and the **mass flow method** where a constant flow of a carrier gas with the sample product circulates through the recipient [26, 27, 28, 29, 30, 31]. In the first case the product concentration inside the recipient can be calculated through the injected volume and the volume of the recipient. The second method needs mass flow controllers to mix the sample product in the carrier gas. The main advantage of this method is its velocity.

3.1. Sensor arrays

There are now more then ten companies selling electronic noses. The sensing technologies chosen by almost all of them are a combination of conductive polymers, MOS sensors or piezoelectric elements. In research institutions other sensing technologies like MOSFET chemical sensors and fluorescent polymers associated with optical fibers are also found [32, 15].

Conductive polymers and MOS sensors are simpler to interface because they are resistive elements. Piezoelectric elements need more complicated signal conditioning circuits to convert the frequency to a voltage. It is also common in this case to compensate the frequency drift due to temperature effects with a non-exposed sensing element.

Figure 4 presents the diagram of a typical olfactory sensing system based

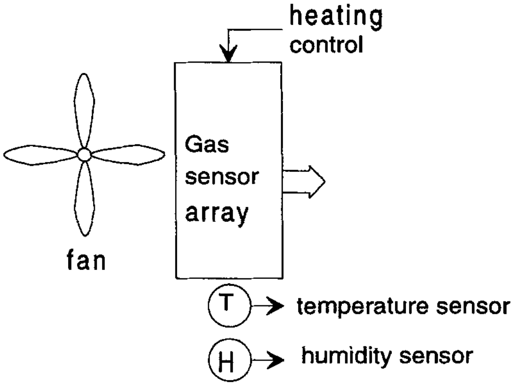


Figure 4. Diagram of a MOS based olfactory sensing system.

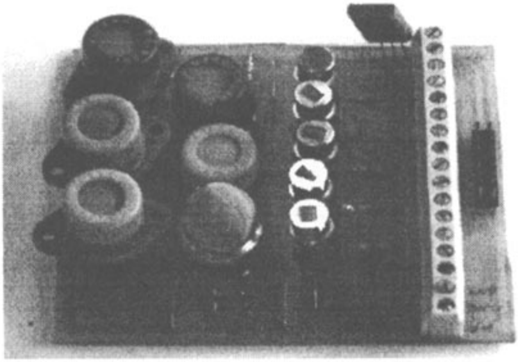


Figure 5. Prototype of the ISR electronic nose. The nose is composed by an array of 11 Tagushi gas sensors, a humidity sensor and a temperature sensor. The gas analysis is based on a fuzzy neural network approach.

on MOS elements. The non-linear dependence of the sensor resistance with the heating temperature can be used to increase the dimension of the data array. This way, for each temperature of the sensors there are linearly independent sets of values. Because MOS gas sensors feature also a high sensitivity to environment humidity and temperature, it is common to feed these variables to the pattern recognition algorithm.

Figure 5 presents a board with several commercial MOS sensors. The resistance change of each element when the array is exposed to alcohol can be seen in Figure 6. Before the sensor output is fed to the pattern recognition algorithm, these values should be pre-processed. Some typical pre-processing methods use the **difference** between the resistance in air and in gas $R_{air} - R_{gas}$ [33]. Other methods use the **relative value** $(R_{gas})/R_{air}$, the **relative difference** $(R_{air} - R_{gas})/R_{air}$, or the logarithm of these relations [18, 33, 34].

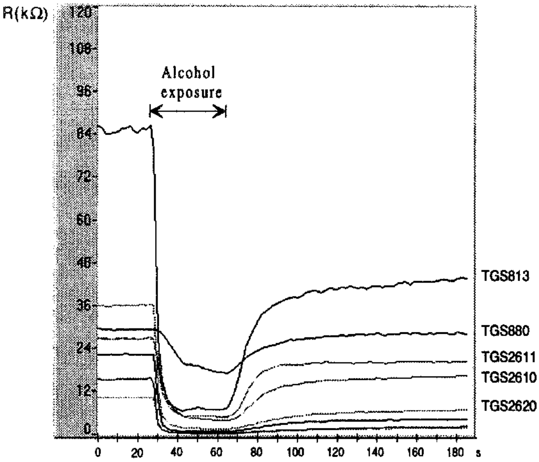


Figure 6. Output from a MOS sensor array to an alcohol exposure. It is visible the low selectivity of each element. The elements in a top down order are the Figaro TGS813, TGS880, TGS2611, TGS2610, TGS2620, TGS822 and TGS800

3.2. Pattern recognition

The number of identifiable patterns from a sensor array is limited by the number of different gas sensors, by their repeatability, by the quantization errors on the acquired signals, and by the calibration accuracy.

There are two conventional methods for extracting information about the gas mixture composition from multiple sensors: one is a statistical technique such as multiple linear regression, and the other is based on artificial neural networks (ANN) [35, 36].

The first pattern recognition methods used on electronic noses were based on vector space methods [37]. These methods model the sensor output as a linear combination of exposed gas's concentration:

$$V_n = a_{1n}C_1 + a_{2n}C_2 + .. + a_{mn}C_m \tag{3}$$

where V_n is the output of sensor n, C_1 to C_m is the concentration of each constituent gas and a_{1n} to a_{mn} are linear constants to be determined by calibration. For simplicity we can write the equation of each sensor on a matrix format:

$$V = A \cdot C \tag{4}$$

Before using the system we should determine the elements of A. These are obtained by exposing the system to known samples of calibration gases.

$$A = V \cdot C^{-1} \tag{5}$$

When a gas is to be tested, then a set of simultaneous equations is obtained and solved to give the value of concentration for each constituent gas.

$$C = A^{-1} \cdot V \tag{6}$$

The main problems with this model are the non-linearity of the sensor elements (see for example equation 2 for the case of MOS elements), limitations on the signal accuracy, difficulties of calibration, and the possible presence of additional constituents at a significant level.

More recent approaches are based on ANN. Many ANN configurations and training algorithms have been used to build electronic noses including: backpropagation-trained, multilayer feed-forward networks, fuzzy ARTmaps, learning vector quantizers (LVQs), Hamming networks, etc [25, 38, 39, 31, 35, 40, 41, 42].

4. Current utilization of odor sensing in robotics

Several authors have suggested the utilization of chemical sensing on mobile robotics. Engelberger for example suggested the utilization of a short-lived chemical mark as an aid for floor cleaning robots [43]. Siegel imagined scenarios for mobile robots self-directed motivation based on chemical senses. He supported his theory on the chemically based navigation of primitive mobile life forms [44, 45].

Russell investigated the use of an odor as a temporal navigational marker. He followed a camphor trail with a mobile robot equipped with two piezoelectric gas sensors. In the beginning the robot was placed with a sensor on the left side of the trail and the other sensor on the right side. The main problems reported by the author are related to the time response of the sensor and the uncertain duration of the mark [46, 47].

Kanzaki and Ishida proposed some odor-source searching strategies without memory and learning, based on the silkworm moth behavior [1, 2]. In their experiences, they used a mobile stage with four pairs of gas and anemometric sensors. They disposed the sensors apart from each other, so that they could measure gas concentration gradients and wind direction. From the experiences carried out, they found out that if the stage is already inside an odor plume and the wind is strong enough, they could move upward the gradient to find the odor-source. Otherwise, it was better to zigzag obliquely upwind across the plume in order to find the odor-source [48, 49].

Cybermotion is an US enterprise that builds security robots equipped with a set of environmental monitoring sensors. Among those sensors there are temperature, humidity, smoke, flame and MOS gas sensors for detection of explosive and toxic gases. With these sensors the robot is able to perform a set of preventive tasks that have already made possible the detection of toxic gases, gas leaks, burning equipment, etc.

5. Future developments and expected utilization

This section identifies some applications of electronic noses associated to mobile robots.

5.1. Security

One of the best tools for narcotics and explosive detection is the dog. It is believed that an Alsatian dog can detect TNT in concentrations as low as five parts per billion.

A robot with a very sensitive electronic nose that detects dangerous (explosives) and illegal substances (drugs) and can be an excellent replacement for police dogs. This robot could be used to patrol public buildings like embassies, airports, train stations, etc. It can move with autonomy, does not become tired and does not need special training.

5.2. Demining

The number of abandoned land mines was estimated to exceed 100 million spread by over 67 countries. At present, it costs about $3 dollars to lay a mine and from $200 to $1000 to find it again and dig it up. For every dug mine, up to 20 more are laid. For example in Angola, there are more mines in the ground than people in the country.

A possible solution for the problem is the development of an autonomous robot that could be placed on the ground to demine dangerous areas [50]. The biggest problem with such a robot is the lack of good sensors for mine detection. Since World War II land mines are essentially plastic, having a minimal metal content. The only way to sense such a mine is to detect the explosive vapors liberated by them. Although it is a difficult task, there are numerous research groups searching for a suitable sensor to detect these mines with some good results already reported [51, 52].

5.3. Agriculture

In recent years several research groups have study the application of robotics in agriculture. The automation of tasks like measurement and control of environmental conditions, plant inspection, and spraying of pesticides, fungicides and other chemical products over the plants, can have significant economic and health impacts, avoiding the workers exposure to insecticides and other dangerous chemical products [53].

Electronic noses have plenty of potential usefulness associated with farming robots, because they provide the necessary sensorial feedback for some of the most common farming tasks. For example the robot can analyze in real-time the volatile compounds released by the soil and fertilize it with the estimated needs. This way the fertilizer is not wasted and the soil does not become contaminated with too much nitrate.

The electronic nose can analyze the environmental conditions to prevent diseases and actuate before the plants become ill. Some diseases release volatile compounds. In these cases, if the plants are already ill, the robot can report the situation in order to prevent the spread of the disease.

A harvesting robot can use aroma information to selectively gather ripe fruits or adult flowers. If the robot knows a map of the plants around the farm, it can use odor information as a rough localization method: near a rose flowerbed, it should smell rose aroma.

5.4. Environmental monitoring

A robot with an electronic nose patrolling a commercial building or an industrial plant, can identify contaminants in the field and make real-time reports about the environmental state. When the robot finds some abnormal situation, like a gas leak or an equipment on fire, it can place a warning for the plant control.

5.5. Cleaning

A cleaning robot with an electronic nose can detect the ammonia odor of an already cleaned floor. This way the robot does not waste time cleaning the floor again.

5.6. Cooperative robotics

Volatile chemicals can be used as temporary marks to coordinate a set of autonomous robots executing a common task [54]. For example, if several cleaning robots clean a huge area, they can use odor information to detect places recently cleaned by other robots because the ammonia odor will be stronger in these places. In a similar way, security robots can mark its path with a volatile trail in order to detect paths recently patrolled.

References

[1] Kanzaki R 1996 Behavioral and neural basis of instictive behavior in insects: Odor-source searching strategies without memory and learning. *Robotics and Autonomous Systems*, 18:33–43

[2] Ishida H, Hayashi H, Takakusaki M, Nakamoto T, Moriizumi T, Kanzaki R 1996 Odour-source localization system mimicking behaviour of silkworm moth. *Sensors and Actuators*, A99:225–230

[3] Sudd J 1967 *An Introduction to the Behavior of Ants*. Arnold Publishers

[4] Moseley P 1997 Solid state gas sensors. *Measurement Science and Technology*, 8:223–237

[5] Gardner J 1994 *Microsensors, Principles and Applications*. John Wiley & Sons

[6] Göpel W, Jones T, Kleitz M, Lundstrom J, Seiyama T (eds) 1991 *Sensors: A Comprehensive Survey*, vol. 2. Weinheim: VCH

[7] Göpel W, Jones T, Kleitz M, Lundstrom J, Seiyama T (eds) 1992 *Sensors: A Comprehensive Survey*, vol. 3. Weinheim: VCH

[8] Moseley P, Tofield B (eds) 1987 *Solid State Gas Sensors*. Adam Hilger

[9] Moseley P, Norris J, Williams D (eds) 1991 *Techniques and Mechanisms in Gas Sensing*. Adam Hilger

[10] Lambrechts M, Sansen W (eds) 1992 *Biosensors: Microelectrochemical Devices*. Adam Hilger

[11] Jones E 1987 The Pellistor Catalytic Gas Detector. In: Moseley P, Tofield B (eds), *Solid State Gas Sensors*, Adam Hilger, pp 17–31

[12] Nieuwenhuizen M, Nederlof A 1992 Silicon Based Surface Acoustic Wave Gas Sensors. In: Gardner J, Bartlett P (eds), *Sensors and Sensory Systems for an Electronic Nose*, Kluwer Academic Publisher, vol. E212 of *NATO ASI Series*, pp 131–145

[13] Bartlett P, Archer P, Ling-Chung S 1989 Conducting polymer gas sensors, Part I: Fabrication and characterization. *Sensors and Actuators*, 19:125–140

[14] Persaud K, Pelosi P 1992 Sensor Arrays using Conducting Polymers for an Artificial Nose. In: Gardner J, Bartlett P (eds), *Sensors and Sensory Systems for an Electronic Nose*, Kluwer Academic Publisher, vol. E212 of *NATO ASI Series*, pp 237–256

[15] Dickinson J, White J, Kauer J, Walt D 1996 A chemical-detecting system based on a cross reactive optical sensor array. *Nature*, 382:697–700

[16] Morrison S March 1991 Semiconducting-Oxide Chemical Sensors. *IEEE Circuits Devices Mag*, 7:32–35

[17] Ihokura K, Watson J 1994 *The Stannic Oxide Gas Sensor*. CRC Press

[18] Horner G, Hierold C 1990 Gas analysis by partial model building. *Sensors and Actuators*, B2:173–184

[19] Bernhardt K, Fleischer M, Meixner H 1995 Innovative sensor materials open up new markets. *Siemens Components*, XXX:35–37

[20] Bernhardt K 1996 Gas sensors of gallium oxide. *Siemens Components*, XXXI:28–30

[21] Persaud K, Dodd G 1982 Analysis of discrimination mechanisms in the mammalian olfactory system using a model nose. *Nature*, 299:352–355

[22] Keller P E, Kangas L J, Liden L H, Hashem S, Kouzes R T 1995 Electronic Noses and Their Applications. In: *IEEE Northcon/Technical Applications Conference*

[23] Hambraeus G 1997 Sensors and electronic noses. *New Nordic Technology*, 3:15–19

[24] Byfield M, May I 1996 Olfactory Sensor Array Systems: The Electronic Nose. *GEC Journal of Research*, 13:17–27

[25] Bourrounet B, Talou T, Gaset A 1995 Application of a multi-gas-sensor device in the meat industry for boar-taint detection. *Sensors and Actuators*, B27:250–254

[26] Shurmer H, Gardner J 1992 Odour discrimination with an electronic nose. *Sensors and Actuators*, B8:1–11

[27] Matsuoka H, Dousaki S, Sera K, Shinohara A, Ishii H 1991 Development of quantitative odor supplier for a plant leaf sensor system. *Sensors and Actuators*, B5:129–133

[28] Wang X, Fang J, Carey P, Yee S 1993 Mixture analysis of organic solvents using nonselective and nonlinear taguchi gas sensors with artificial neural networks. *Sensors and Actuators*, B13:455–457

[29] Matsuoka H, Yamada Y, Dousaki S, Nemoto Y, Shinohara A, Satoh A 1993 Application of teflon particle column to an odor-sensing system. *Sensors and Actuators*, B13:358–361

[30] Ide J, Nakamoto T, Moriizumi T 1993 Development of odour-sensing system using an auto-sampling stage. *Sensors and Actuators*, B13:351–354

[31] Endres H, Jander H D, Göttler W 1995 A test system for gas sensors. *Sensors and Actuators*, B23:163–172

[32] Holmberg M, Winquist F, Lunstr m I, Gardner J, Hines E 1995 Identification of papper quality using a hybrid electronic nose. *Sensors and Actuators*, B27:246–249

[33] Gardner J, Bartlett P 1992 Pattern Recognition in Odour Sensing. In: Gardner J, Bartlett P (eds), *Sensors and Sensory Systems for an Electronic Nose*, Kluwer Academic Publisher, vol. E212 of *NATO ASI Series*, pp 161–179

[34] Moore S, Gardner J, Hines E, Göpel W, Weimar U 1993 A modified multilayer perceptron model for gas mixture analysis. *Sensors and Actuators*, B16:344–348

[35] Di Natale C, Davide F, D'Amico A 1995 Pattern recognition in gas sensing:

well-stated techniques and advances. *Sensors and Actuators*, B23:111–118

[36] Ping W, Jun X 1996 A novel recognition method for electronic nose using artificial neural network and fuzzy recognition. *Sensors and Actuators*, B37:169–174

[37] Shurmer H 1990 Basic limitations for an electronic nose. *Sensors and Actuators*, B1:48–53

[38] Bednarczyk D, DeWeerth S P 1995 Smart chemical sensing arrays using tin oxide sensors and analog winner-take-all signal processing. *Sensors and Actuators*, B27:271–274

[39] Schweizer-Berberich M, Göppert J, Hierlemann A, et al. 1995 Application of neural network-system to the dynamic response of polymer-based sensor arrays. *Sensors and Actuators*, B27:232–236

[40] Niebling G, Schlachter A 1995 Qualitative and quantitative gas analysis with non-linear interdigital sensor arrays and artificial neural networks. *Sensors and Actuators*, B27:289–292

[41] Llobet E, Brezmes J, Vilanova X, Sueiras J, Correig X 1997 Qualitative and quantitative analysis of volatile organic compounds using transient and steady-state response of a thick-film tin oxide gas sensor array. *Sensors and Actuators*, B41:13–21

[42] Ratton L, Kunt T, McAvoy T, Fuja T, Cavicchi R, Semancik S 1997 A comparative study of signal processing techniques for clustering microsensor data (a first step towards an artificial nose). *Sensors and Actuators*, B41:105–120

[43] Engelberger J F 1989 *Robotics in Service*. MIT Press, Cambridge

[44] Wong L, Takemori T, Siegel M 1989 Gas Identification System using Graded Temperature Sensor and Neural Net Interpretation. Tech. Rep. CMU-RI-TR-20-89, The Robotics Institute - Carnegie Mellon University

[45] Siegel M 1990 Olfaction, Metal Oxide Semiconductor Gas Sensors and Neural Networks. In: Henderson T (ed), *Traditional and Non-Traditional Robotic Sensors*, Springer-Verlag, Berlin, vol. F63 of *NATO ASI Series*, pp 143–157

[46] Deveza R, Thiel D, Russell A, Mackay-Sim A June 1994 Odor Sensing for Robot Guidance. *International Journal of Robotics Research*, 13:232–239

[47] Russell A, Thiel D, Mackay-Sim A 1994 Sensing Odour Trails for Mobile Robot Navigation. In: *Int. Conf. on Robotics and Automation*, pp 2672–2677

[48] Ishida H, Suetsugu K, Nakamoto T, Moriizumi T 1994 Study of autonomous mobile sensing system for localization of odor source using gas sensors and anemometric sensors. *Sensors and Actuators*, A45:153–157

[49] Nakamoto T, Hishida H, Moriizumi T 1997 Active odor sensing system. In: *Int. Symposium on Industrial Electronics*, pp SS128–SS133

[50] Walker J 1995 Minerats: Moore's Law in the Minefield. In: *IEEE Asilomar Microprocessor Workshop*

[51] Maechler P 1995 Detection Technologies for Anti-Personnel Mines. In: *Symposium on Autonomous Vehicles in Mine Countermeasures*

[52] McGoldrick P Dec 1996 Creative technologies seeking a solution to abandoned land mines: an impressive spread of hope and dignity. *Electronic Design*, pp 74–78

[53] Mandow A, de Gabriel J G, Martínez J, Muñoz V, Ollero A, García-Cerezo A 1996 The Autonomous Mobile Robot AURORA for Greenhouse Operation. *IEEE Robotics Automation Magazine*, 3:18–28

[54] Russell R 1997 Heat trails as short-lived navigational markers for mobile robots. In: *Int. Conf. on Robotics and Automation*, pp 3534–3539

Part Two

Cooperation and Telerobotics

Advanced Telerobotics
G. Hirzinger, B. Brunner, R. Koeppe, J. Vogel

Cooperative Behaviour Between Autonomous Agents
Toshio Fukuda, Kosuke Sekiyama

Mobile Manipulator Systems
Oussama Khatib

Advanced Telerobotics

G. Hirzinger, B. Brunner, R. Koeppe, J. Vogel

DLR (German Aerospace Research Establishment), Oberpfaffenhofen
Institute of Robotics and System Dynamics
D-82234 Wessling/Germany
Tel. +49 8153 28-2401
Fax: +49 8153 28-1134
email: Gerd.Hirzinger@dlr.de

Abstract: The paper outlines different approaches to directly interfere with a robot manipulator for generating smooth motions and desired forces in a natural way, be it for on-line control (e.g. in teleoperation) or for off-line control (e.g. in a virtual environment as part of an implicit task oriented programming system). Issues of position versus velocity control are discussed and the alternatives of force-reflection and pure force forward commanding are outlined.

These kind of natural man-machine interfaces, combined with local sensorbased (shared) autonomy as well as delay-compensating predictive 3D-graphics have been key issues for the success of ROTEX, the first remotely controlled robot in space. The telerobot techniques applied there are now shifted onto the higher level of implicit task-level-programming and VRML environments in the internet; and in addition we have transferred part of these techniques meanwhile into medicine (teleconsultation by telepresence).

Skill transfer from man to machine including the process forces in machining and assembly is discussed as one of the most interesting research topics in this area.

1 General remarks

As **Tom Sheridan**, a pioneer in teleoperational systems, formulated [1], **telerobots combine the advantages of human remote control with the autonomy of industrial robots**. In the beginning of this kind of technology the terms **teleoperation** and **telemanipulation** essentially meant that an artificial anthropomorphic arm a teleoperator or telemanipulator) is directly and remotely controlled by a human operator just like an extension of his own arm and hand, [11]. Typically such a teleoperator system is supposed to work in hazardous and hostile environment [13], while the human operator in his safe and remote control station may make full use of his intelligence, especially when all relevant information is sent back to him from the remote worksite of the teleoperator, e.g., TV-images, forces and distances as sensed by the manipulator.

More specifically if a teleoperator is prepared to repeat a task once **shown** by the operator and especially if some local autonomy using sensory feedback is provided rendering the arm even more adaptive, we prefer to use the above-mentioned term **telerobot**. Typically then the human operator serves as a supervisory master, not controlling the telerobot's every move, but issuing gross commands refined by the robot or - in its ultimate form - stating objectives which are executed via the local robot sensors and computers.

In the following we will have a closer look into the control structures of these telerobotic systems. ROTEX - the first remotely controlled robot in space - proved the efficiency of the telerobotic concepts available today. And it seems that the close interaction between man and machine - essential in telerobotics but not standard in industrial robotics - becomes more and more popular in robot industry. Skill transfer using force reflection may become a major issue even for industrial robots.

2 Telerobotic control loops

2.1 Bilateral feedback concepts

The original teleoperator systems were closely related to the master slave concept, i.e. more or less independent of the special realization there exist operational modes, where the manipulator slave arm tries to track as precisely as possible the positional (including rotational) or rate (i.e. velocity) commands given by a human master who uses some kind of an input device. In the early days of teleoperation this input device (called **master arm**) was just a 1:1 replica of the slave arm, both connected via mechanical coupling. More advanced systems as presently used in nuclear power plants show up pure electrical coupling between the two arms, thus allowing remote control over great distance, provided that a TV-transmission from slave worksite to the remote control station is used. An example for this concept of **bilateral position control** is depicted in Fig. 1. By the left hand side position control system the slave joints (joint angle vector $\mathbf{q}_S$) are forced to follow the master joints $\mathbf{q}_m$ as closely as possible using kind of a PD servo-controller with gain matrix $\mathbf{S}_S$ and damping matrix **D**. The gain matrix $\mathbf{S}_S$ may be derived from a desired Cartesian stiffness matrix $\mathbf{S}_{X,S}$ following Salisbury's stiffness control concept:

$$\tau_S = \underbrace{\mathbf{J}_S^T \mathbf{S}_{X,S} \mathbf{J}_S}_{\mathbf{S}_s} \Delta \mathbf{q}_s \qquad (1)$$

relating slave joint errors $\Delta\mathbf{q}_S$ and torques τ_S with $\mathbf{J}_S$ as the slave Jacobian.

The right hand side control system in Fig. 1 provides a safety-relevant kind of force reflection into the master arm. Assume that the slave (due to a master motion) moves into a wall or obstacle not realized visually by the operator. The left hand side control system - with the positional error increasing - would force the slave arm to exert increasing forces (eventually arriving at the maximum forces exertable by the slave joints) and thus might destroy the slave arm or the environment. The right hand side loop therefore feeds back the positional joint errors (with a certain gain $\mathbf{S}_m = \mathbf{J}_m^T \mathbf{S}_{x,m} \mathbf{J}_m$ generating a corresponding Cartesian master stiffness $\mathbf{S}_{x,m}$ into the master arm joint motors. In an advanced system these joint controls would in addition compensate for the gravitational friction, inertial and coupling forces using an inverse model (as indicated by the compensating terms $\mathbf{h}_S(\mathbf{q}_S, \dot{\mathbf{q}}_S)$ and $\mathbf{h}_m(\mathbf{q}_m, \dot{\mathbf{q}}_m)$).

The operator in the ideal case would really sense only the positional deviations caused e.g. by external forces and torques (e.g. when colliding, lifting loads etc.). In reality however such a system provides reaction forces to the operator during any kind of motion due to the unavoidable positional errors in such a servo system.

With these observations it comes clear that a system as depicted in Fig. 2 using an external wrist force-torque sensor between the slave's last joint and its endeffector is superior and capable of overcoming these difficulties. The sensed external force-torque vector $\mathbf{f}_S$ at the slave's wrist has to be multiplied by $\mathbf{J}_m^T$, the transpose Jacobian of the master arm to yield the joint torques $\tilde{\tau}_m$, which are necessary to produce this same reaction force-torque vector $\mathbf{f}_m = \mathbf{f}_S$ at the master's wrist. The right hand side positional feedback of Fig. 1 is no longer necessary; systems of this kind are called **bilateral force-reflecting** master-slave systems. The comments made above concerning inverse dynamic model techniques in the slave are still valid here.

Note that the scheme in Fig. 2 is Cartesian based now, i.e. the kinematics of master and slave are independent, provided that the master arm shows up 6 degrees of freedom and has a similarly shaped workspace (possibly down-scaled) compared to the slave arm. The Cartesian errors $\Delta\mathbf{x}, \Delta\dot{\mathbf{x}}$ are transformed via the inverse kinematics into the corresponding joint (and if needed joint velocity) errors of the slave. 6-dof-hand controllers of this type have been developed without and with force-feedback (e.g. the **kinesthetic handcontroller** of **Bejczy** [4]). Although force-reflecting hand controllers may provide a considerable performance improvement [3], systems realized up to now do not always show up the requested **high fidelity**; this has partly to do with high feedback sampling rates needed (1 kHz seems a reasonable value), and with friction problems in the hand-controller. In zero gravity (astronauts as operators) no experience is yet available concerning human's reaction to force reflection.

2.2 Local autonomy concepts

In all cases discussed so far if the operator cannot see the slave robot directly he may make use of a TV-transmission line and look at a monitor image (or better stereo TV image) instead of the real scenery. But if there are signal transmission delays between master and slave (e.g. when teleoperating a space robot from ground) then the bilateral schemes discussed so far implying the human arm in the feedback loop tend to fail. Predictive (i.e. delay-compensating) graphics computation and feedback of forces into the human arm seems feasible when a perfect world model and a high-fidelity force reflection device is available, but is difficult to realize in practice.

Thus in the sequel we are addressing techniques that do not provide any force-sensing in the human arm; these concepts are characterized by feedforward commands and **local**, artificially generated compliance using **impedance control** without a force sensor or **active compliance** with local sensory feedback. **Impedance** control means that the robot's endeffector reacts onto an external force vector **f** just as a desired Cartesian impedance, i.e. [5]

$$\tilde{\mathbf{M}}_x \ddot{\mathbf{x}} + \mathbf{D}_x(\dot{\mathbf{x}}_d - \dot{\mathbf{x}}) + \mathbf{S}_x(\mathbf{x}_d - \mathbf{x}) = \mathbf{f} \tag{2}$$

where the mass, damping and stiffness matrices $\tilde{\mathbf{M}}_x, \mathbf{D}_x, \mathbf{S}_x$ respectively are chosen arbitrarily and $\mathbf{x}_d, \dot{\mathbf{x}}_d$ denotes a desired motion. Fig. 3 shows a corresponding structure and indicates that feedback to the human operator is only visual now, the robot being locally compliant with chosen impedance, so that it does not destroy its environment when the artificial stiffness $\mathbf{S}_x$ is chosen reasonably.

So far all the structures proposed are based on the advanced concept of direct torque control on the joint level. This however is not state of the art until now, so it is justified to look for other practical concepts, especially using the disturbance rejecting positional command interfaces which are offered by all present day robots. **Active compliance** concepts based on local feedback of wrist force sensor data into the positional interface go back to **Whitney** (e.g. [6] and have led to different implementation proposals for telerobots (e.g. [7], [8], [12]). The scheme proposed in [4] may be characterized by Fig. 4, where the wrist forces are added to the positional errors between master and slave via a first order filter, again generating a certain Cartesian stiffness $\mathbf{S}_x$ and damping via the time constant α (s here denotes the Laplace variable). In the stationary case, e.g. when the master position $\mathbf{x}_m$ has been moved **behind** a rigid surface in the environment $\mathbf{x}_{env}$ (Fig. 5), we have $\Delta\mathbf{x} = \Delta\mathbf{x}_{compl}$ so that the robot's motion stops. In fact a scheme like that of Fig. 4 works only if the slave robot has some inherent mechanical compliance $\mathbf{S}_R$ either caused by the inevitable joint compliance (leading to a position-dependent overall compliance) or by a dedicated compliance in the wrist (e.g. Draper Lab's well-known remote center compliance RCC). Now if we ask for the resulting compliance $\tilde{\mathbf{S}}$, relating $\mathbf{x}_m - \mathbf{x}_{env}$ to the sensed force $\mathbf{f}_S$ exerted by the slave, we have to solve the equations (see Fig. 5)

$$\underbrace{(\mathbf{x}_s - \mathbf{x}_{env})\mathbf{S}_R}_{\mathbf{f}_S} = \underbrace{\mathbf{S}_x \Delta\mathbf{x}_{compl}}_{\mathbf{f}_S} = \underbrace{\mathbf{S}_x \Delta\mathbf{x}}_{\mathbf{f}_S} = \mathbf{S}_x(\mathbf{x}_m - \mathbf{x}_S) \tag{3}$$

$$\mathbf{f}_S = \tilde{\mathbf{S}}(\mathbf{x}_m - \mathbf{x}_{env}) \tag{4}$$

After a few elementary calculations we arrive at

$$\tilde{\mathbf{S}} = \mathbf{S}_R(\mathbf{S}_R + \mathbf{S}_x)^{-1}\mathbf{S}_x \tag{5}$$

i.e. an extremely stiff slave robot ($\mathbf{S}_R$ very dominant) would lead to the desired $\mathbf{S}_x$, while a very compliant slave would behave near to its natural stiffness. Clearly by appropriate choice of the stiffness matrix $\mathbf{S}_x$ one may generate different compliance in different axes.

Let us recall that the last two concepts presented were based on artificial slave compliance without and with a **local** force sensor feedback loop, and no force reflection into the human arm. Basically even in case of active local compliance force reflection into the human arm seems feasible, but presumably it would be reasonable to supply force feedback to the operator only in those directions which are not locally force-controlled.

Note that glove-like input devices as pure positional / rotational controllers and (force-reflecting) exoskeletons fully fit into this framework, too.

Alternative concepts as developed at the German Aerospace Research Establishment (DLR) and widely applied in ROTEX are based on local sensory feedback and purely feed forward type of 6 dof handcontrollers, too, but the master input devices are designed in a way so that they work as **rate and force-torque command systems** simultaneously in contrast to the positional master arms discussed so far. Motions permitted are very small (typically 1--2 mm via springs, i.e. no joints) making the mechanical design fairly simple. DLR's **sensor** or **control ball** (meanwhile redesigned into the „SPACE MOUSE“ or MAGELLAN (chapter 3)) contains an optically measuring 6 component force-torque sensor (Fig. 9 and Fig. 10), the basic principle of which is also used in DLR's compliant wrist sensors.

The main features of the underlying more general telerobotic concept are shown in Fig. 6 and Fig. 7. Rudimentary commands $\Delta\mathbf{x}$ (the former deviations between master and slave arm) are derived either from the 6 dof device as forces (using a sort of artificial stiffness relation $\Delta\mathbf{x} = \mathbf{S}_x^{-1}\mathbf{f}$) or from a path generator connecting preprogrammed points ($\Delta\mathbf{x}$ being the difference between the path generator's, i.e. „master's“, position and the commanded ``slave" robot position $\mathbf{x}_{com}$). Due to the above-mentioned artificial stiffness relation these commands are interpreted in a dual way, i.e. in case of free robot motion they are interpreted as pure translational / rotational commands; however if the robot senses contact with the environment, they are projected into the mutually orthogonal „sensor-controlled“ (index **f**) and „position-controlled"“ (index p) subspaces, following the **constraint frame concept** of **Mason** [9]. These subspaces are generated by the robot autonomously using a priori information about the relevant phase of the task and actual sensory information: to discern the different task phases (e.g. in a peg-in-hole or assembly task) automatically. Of course the component $\Delta\mathbf{x}_p$ projected into the position controlled subspace is used to feed the position controller; the component $\mathbf{f}_f$ projected into the sensor-controlled subspace is either compared with the sensed $\mathbf{f}_{sens}$ to feed (via the robot's

Cartesian stiffness $\mathbf{S}_R$) the orthogonally complementary force control law, (which in fact is another position controller yielding a velocity $\dot{\mathbf{x}}_f$), or it is neglected and replaced by some nominal force vector $\mathbf{f}_{nom}$ to be kept constant e.g. in case of contour following. We prefer to talk about sensor-controlled instead of force-controlled subspaces, as non-tactile (e.g. distance) information may be interpreted as pseudo-force information easily, the more as we are using the robot's positional interface anyway. However we omit details as treated in [7] concerning transformations between the different Cartesian frames (e.g. hand system, inertial system etc.). The „resulting“ stiffness (e.g. when the path generator serves as master) in the sense of Fig. 4 and the corresponding derivations eqs.(3-5) are the same as for the scheme of Fig. 7, given by eq. (5).

It is worth to be pointed out again that in case of real human teleoperation although there is no force feedback into the operator's arm, the robot using its local feedback exerts only the forces (may be scaled) as given by the ``teacher", thus is fully under his control, or may behave autonomously in predefinable sensor or position-controlled subspaces. We are thus talking of **shared control** (see [8], [12], [4]), i.e. some degrees of freedom are directly controlled by a supervisor, while others are controlled autonomously and locally by the machine. The local loops in
Fig. 6 are (at least presently) characterized by modest intelligence but high bandwidth, while the loops involving the human operator`s visual system are of lower bandwidth but higher intelligence. Surely the long-term goal is to shift more and more autonomy to the robot's site and to move the operator's commands to an increasingly higher level. Shared local autonomy and feedback control as explained above using the different type of gripper sensors was a key issue for the success of ROTEX.

2.3 Predictive control

When teleoperating a robot in a spacecraft from ground or from another spacecraft so far away that a relay satellite is necessary for communication, the delay times are the crucial problem. Predictive computer graphics seems to be the only way to overcome the main problems. Indeed predictive 3D-stereographic simulation (Fig. 20) was a key issue for the success of the ROTEX experiment (chapter 4), with its typical round-trip delays of 5-7 seconds. Fig. 8 is to outline that the human operator at the remote workstation handles a 6 dof handcontroller (e.g. control ball) by looking at a ``predicted" graphics model of the robot. The control commands issued to this instantaneously reacting robot simulator were sent to the remote robot as well using the time-delayed transmission links. In addition to the simulated preview the groundstation computer and simulation system may contain a model of the uplink and downlink delay lines as well as a model of the actual states of the real robot and its environment (especially moving objects). The real down-link sensory and joint data are compared with the down-link data as expected from the simulator, the errors are used in an „observer“ or „estimator“ type of scheme to correct the model assumptions. Fig. 8 leaves open whether the dynamic model for the on-board situation includes local sensing feedback or not. If we consider tactile interaction with the environment, in our opinion local sensory feedback is absolutely necessary in case of a few seconds delay, presimulating forces may not be precise enough. However, if the robot has to visually servo and grasps a floating object, we have more or less perfect dynamic models and furthermore, estimation errors of, e.g., half a millimeter may be not crucial for a safe grasp. In ROTEX indeed we had both situations with and without local sensing feedback (Fig. 20).

Overlaying graphically generated sceneries [10] and real TV-images s particularly useful for calibration, e.g. for assuring that the computational viewpoint is identical to the real one, and that the real world is close to the virtual one. Other than that it may be sufficient in many cases to overlay the predicted graphics with e.g. a wire frame representation of the wrist as derived from the down-link joint data or even only an indication of the hand-frame. This was realized in ROTEX, where the real i.e. delayed robot gripper's position was characterized just by a cross-bar and by two patches displaying the gripper's actual opening width.

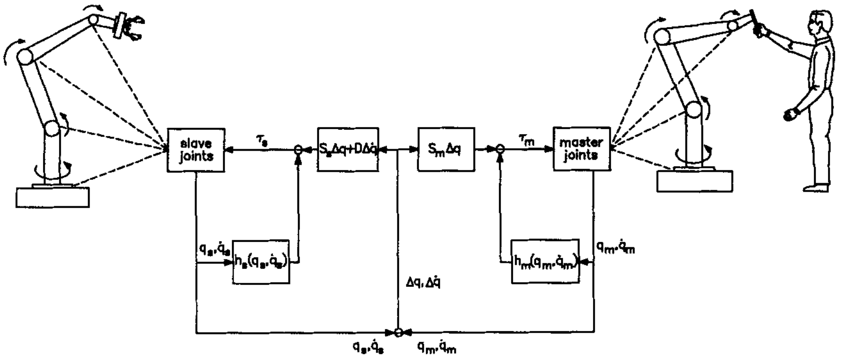


Fig. 1 An advanced bilateral position feedback master-slave teleoperator system

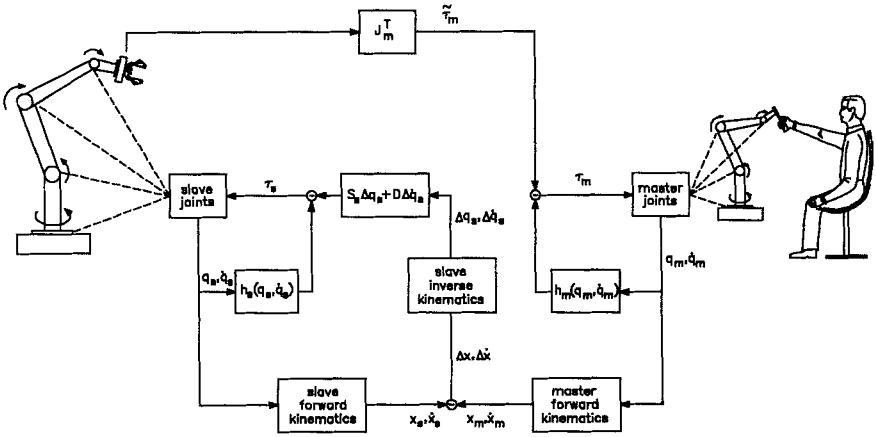


Fig. 2 A bilateral force feedback system, when master and slave are kinematically different

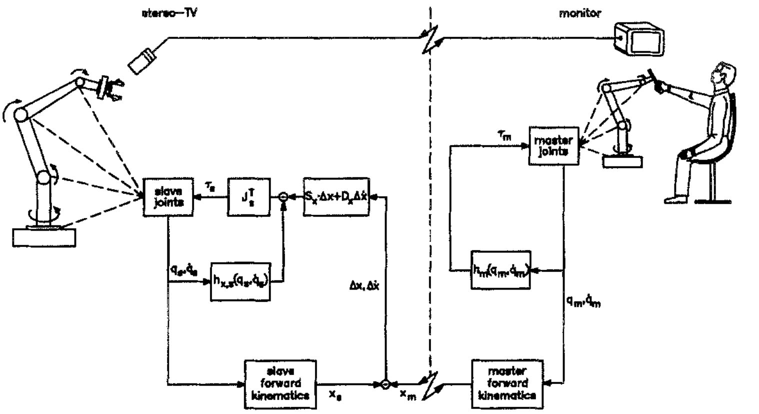


Fig. 3 Impedance control yields a compliant slave without using a force sensor

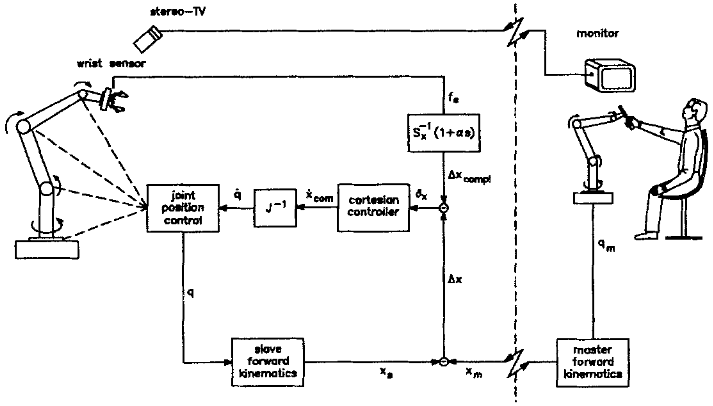


Fig. 4 Active compliance teleoperation via the positional slave interface (transmission delays indicated, master compensation not drawn)

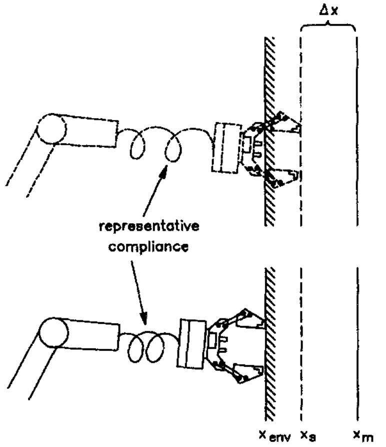


Fig. 5 Active compliance causes the slave robot to be commanded stationarily somewhat ''behind'' a rigid surface x_{env} in the environment, the final position x_s not being identical to the master position x_m

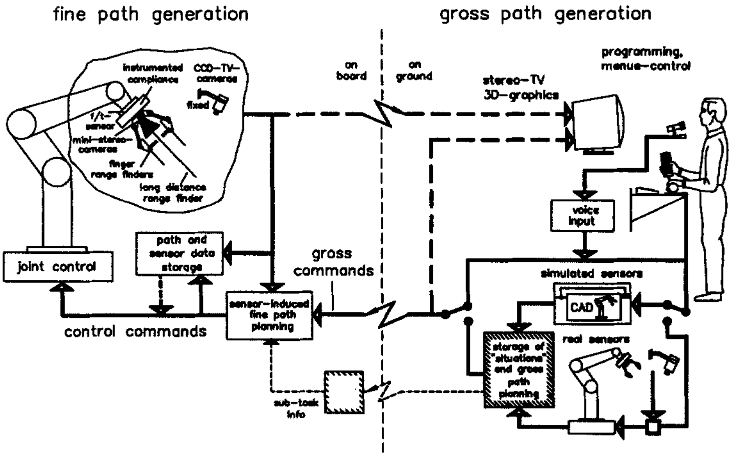


Fig. 6 Overall loop structures for a sensor-based telerobotic concept

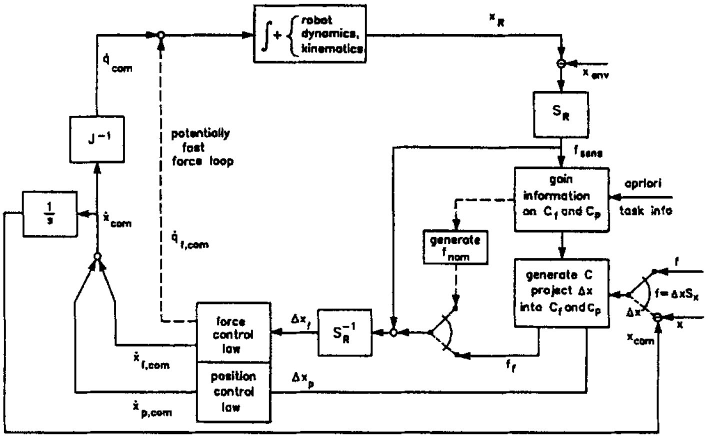


Fig. 7 A local closed loop concept with automatic generation of force and position controlled directions and artificial robot stiffness

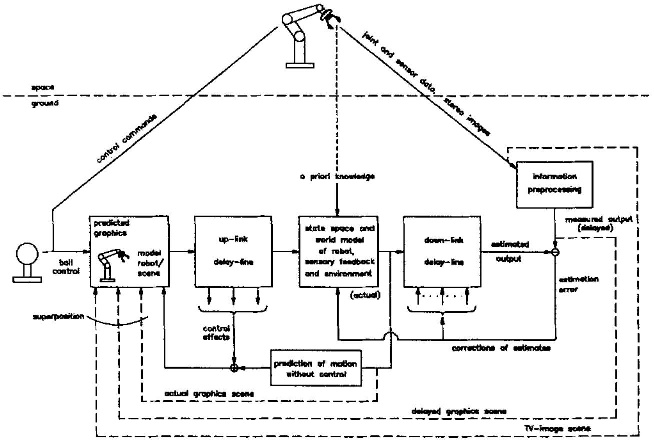


Fig. 8 Block structure of a predictive „dynamic“ estimation scheme

3 DLR`s 6 dof controllers

At the end of the seventies, we started research on devices for the 6 dof control of robot grippers in Cartesian space. After lengthy experiments it turned out around 1981 that integrating a six axis force torque sensor (3 force, 3 torque components) into a plastic hollow ball seemed to be the optimal solution. Such a ball registered the linear and rotational displacements as generated by the forces/torques of a human hand, which were then computationally transformed into translational/rotational motion speeds.
The first force torque sensor used was based upon strain gauge technoloy, integrated into a plastic hollow ball (Fig. 9). DLR had the basic concept ***center of a hollow ball handle approximately coinciding with the measuring center of an integrated 6 dof force/torque sensor*** patented in Europe and US [15]. The concepts developed and patented at that time did not only include the simple, remote 6 dof control (Fig. 14), but also the simultaneous programming of forces and torques via an additional wrist force-torque sensor and appropriate control loops (Fig. 16). And an alternative scheme (Fig. 15) proposed in these early patents referred to „leading the robot around" with a wrist mounted 6 dof input device (the robot trying to null the hand-controller's forces and torques by servoing) and (may be) an additional wrist sensor assuring that the robot would exert the operator's forces / torques onto its environment by appropriate control laws (Fig. 17).

From 1982-1985, the first prototype applications showed that DLR's control ball was not only excellently suited as a 6 dof control device for robots, but also for the first 3D-graphics system that came onto the market at that time. 1985 DLR's developer group succeeded in designing a much cheaper optical measuring system (Fig. 10). The new system used 6 one-dimensional position detectors and received a world-wide patent [16]. The basic principle is shown in Fig. 10. The measuring system consists of an inner and an outer part. The measuring arrangement in the inner ring is composed of the LED (Light emitting diode), a slit and perpendicular to the slit on the opposite side of the ring a linear position sensitive detector (PSD). The slit/LED combination is mobile against the remaining system. Six such systems (rotated by 60 degrees each) are mounted in a plane, whereby the slits alternatively are vertical and parallel to the plane. The ring with PSD's is fixed inside the outer part and connected via springs with the LED-slit-basis. The springs bring the inner part back to a neutral position when no forces/torque are exerted. There is a particularly simple and unique transformation from PSD-signals U_1....U_6 to the unknown displacements. The whole electronics including computational processing on a one-chip-processor was already integrable into the ball.

In the early nineties we redesigned the control ball idea with its unsurpassed, dirft- and wearless opto-electronic measuring system into the low-cost mass product SPACE MOUSE (Fig. 11). In the USA and in Asia it's called MAGELLAN by our license partner LOGITECH. Again all electronics and processing is integrated into the flat cap replacing the former ball shape. **Indeed, intensive ergonomic studies showed that manipulating such a control device with the fingertips instead of**

enclosing it with the palm (which easily happens in case of the ball form) yields highest sensitivity and dynamics.

With more than 12.000 installations, the device has become the most popular 3D-input device for 3D-graphics and 3D-CAD in Europe, and our US-license-company LOGITECH is going to make it **the** world-standard for 3D-graphics interfaces.

But interestingly enough, SPACE MOUSE has now returned back into its original fields of application [17]. Industrial robots of the newest generation (e.g. those of the German market leader KUKA and the French manufacturer STÄUBLI, Fig. 13) use it for programming and teaching, as well as surgical microscopes (e.g. those of the company ZEISS, Fig. 12) use it for spatial control. And - after many years -robot manufacturers, too, are going now to make use of the concept „leading the robot" manually as shown in Fig. 12 and Fig. 15. Thus SPACE MOUSE and its predecessors have massively inspired the thinking in 3D in graphics as well as in robotics.

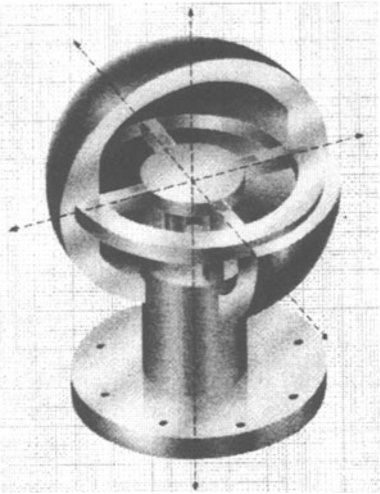


Fig. 9 The first DLR control ball in 1982 used strain gauge technology

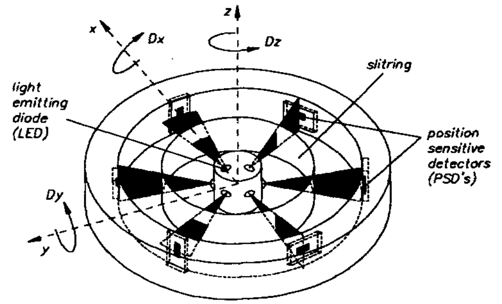

z
x
Dx
Dz
slitring
light emitting diode (LED)
position sensitive detectors (PSD's)
Dy
y

Fig. 10 The patented opto-electronic measuring system core element of the second generation of controll balls as well as of the SPACE MOUSE

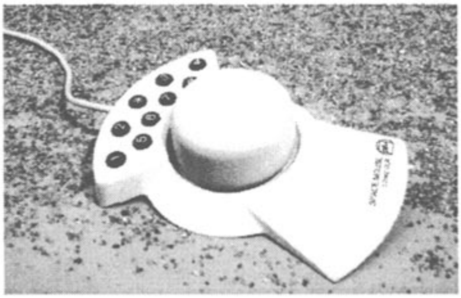


Fig. 11 SPACE MOUSE at the end of a long development chain.

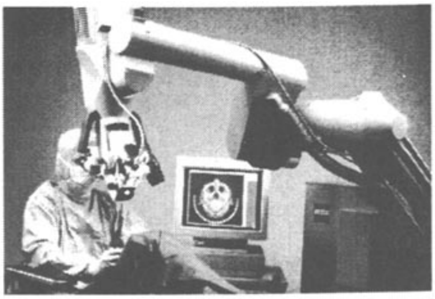


Fig. 12 Guiding surgical microscopes via SPACE MOUSE integrated in the handle (courtesy of Zeiss)

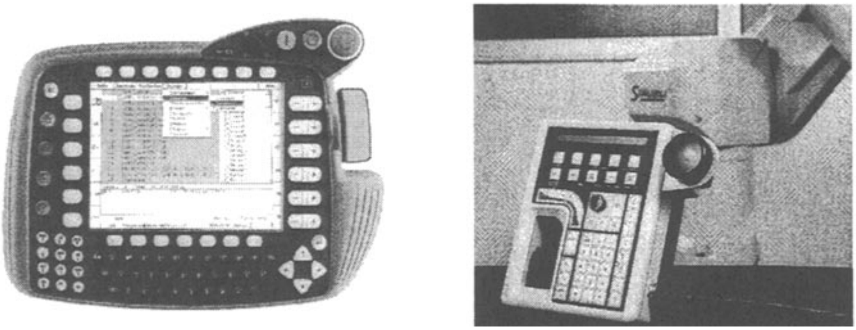


Fig. 13 SPACE MOUSE is used now by different leading robot manufacturers in their control panels (KUKA left, STÄUBLI right)

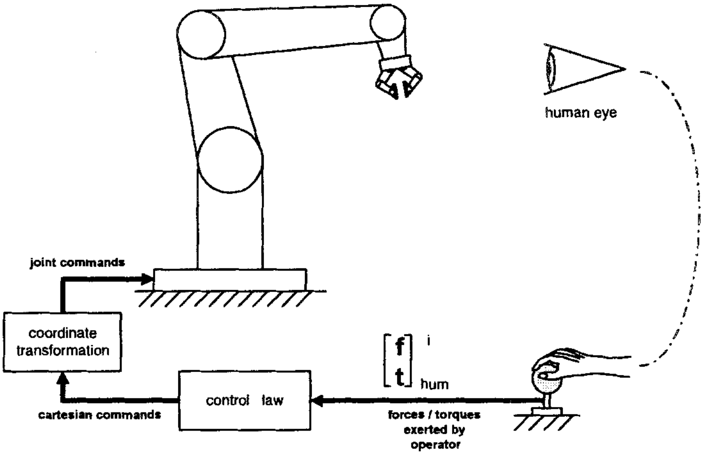

human eye
joint commands
coordinate transformation
control law
cartesian commands
f
t
i
hum
forces / torques exerted by operator

Fig. 14 Path-only-teach-in with fixed 6 dof controller

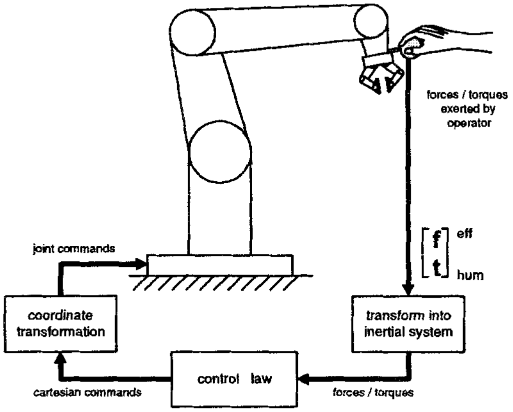

forces / torques exerted by operator
f
t
eff
hum
joint commands
coordinate transformation
transform into inertial system
control law
cartesian commands
forces / torques

Fig. 15 Path-only-teach-in with robot mounted 6 dof controller

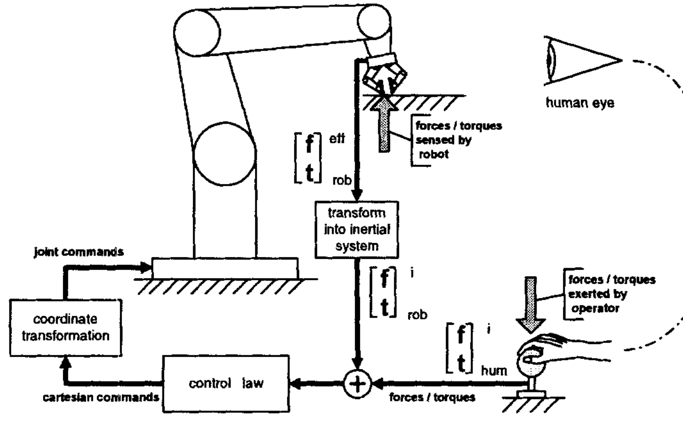

human eye
forces / torques sensed by robot
$\begin{bmatrix} f \\ t \end{bmatrix}^{eff}_{rob}$
transform into inertial system
joint commands
coordinate transformation
$\begin{bmatrix} f \\ t \end{bmatrix}^{i}_{rob}$
forces / torques exerted by operator
$\begin{bmatrix} f \\ t \end{bmatrix}^{i}_{hum}$
control law
cartesian commands
forces / torques

Fig. 16 Simultaneous path-force-torque-teach-in with fixed 6 dof controller

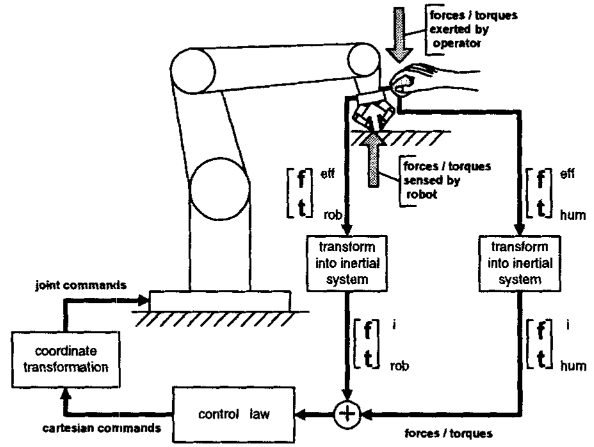

forces / torques exerted by operator
forces / torques sensed by robot
$\begin{bmatrix} f \\ t \end{bmatrix}^{eff}_{rob}$
$\begin{bmatrix} f \\ t \end{bmatrix}^{eff}_{hum}$
transform into inertial system
transform into inertial system
joint commands
coordinate transformation
$\begin{bmatrix} f \\ t \end{bmatrix}^{i}_{rob}$
$\begin{bmatrix} f \\ t \end{bmatrix}^{i}_{hum}$
control law
cartesian commands
forces / torques

Fig. 17 Simultaneous path-force-torque-teach-in with robot mounted 6 dof controller

4 ROTEX - a first European step into space robotics

Main features

ROTEX, flying with shuttle COLUMBIA (spacelab D2) in April 93, was the first remotely controlled space robot. Its main features were [18]:

- A small, six-axis robot (working space ~1 m) flew inside a space-lab rack (Fig. 22). Its gripper was equipped with a number of sensors, particular two 6-axis force-torque wrist sensors, tactile arrays, grasping force control, an array of 9 laser-range finders and a tiny pair of stereo cameras to provide a stereo image out of the gripper; in addition a fixed pair of cameras provided a stereo image of the robot's working area.
- Various prototype tasks, demonstrating servicing capabilities, were performed:
  a) assembling a mechanical truss structure from three identical cube-link parts

b) connecting/disconnecting an electrical plug (orbit replaceable unit ORU exchange using a ''bayonet closure'')
c) catching a free-floating object

A variety of telerobotic operational modes was verified, (Fig. 25, Fig. 19)

- teleoperation on board (astronauts using stereo-TV-monitor)
- teleoperation from ground (using predictive computer graphics) via human operators and machine intelligence as well.
- sensor-based off-line programming (in a virtual environment on ground including sensory perception, with sensor-based automatic execution later onboard).

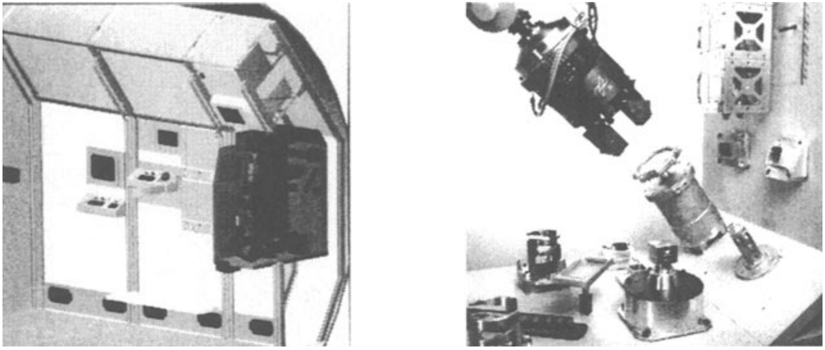


Fig. 18 The ROTEX-workcell (right) integrated into a rack of spacelab (left)

The telerobotic control system

An essential feature of the ROTEX telerobotic system was a *local autonomy, shared control concept* [26] based on sensory feedback at the robot's site, by which gross commands (from human operators or path planners) were refined autonomously.

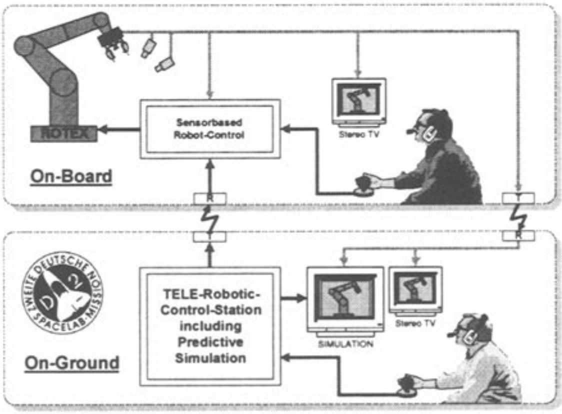

ROTEX
Sensorbased Robot-Control
Stereo TV
On-Board
R
T
T
R
ZWEITE DEUTSCHE SPACELAB-MISSION D2
TELE-Robotic-Control-Station including Predictive Simulation
SIMULATION
Stereo TV
On-Ground

Fig. 19 The ROTEX telerobotic control structures

The uniform man-machine-interface on board and on ground allowing to command „dual“ (i.e. rate and force-torque) information, was DLR`s control ball, a predecessor of the SPACE MOUSE (Fig. 11). Any operation like the dismounting of the bayonet closure was assumed to be composed of sensor based elementary moves, for which a certain constraint-frame- and sensor-type-configuration holds (to be selected by the operator). Thus, *active compliance* as well *as hybrid (pseudo-)force control* were realised locally.

Feedback to the operator - in case of on-line teleoperation - was provided via the visual system, i.e. for the astronaut via stereo TV images, for the ground operator mainly via predictive stereo graphics compensating signal delays of up to 7 seconds. For this type of predictive simulation to work the same control structures and path planners had to be realised on-board as well as on-ground; i.e. not only the robot's free motion but also its sensory perception and feedback behaviour had to be realised „virtually" in the telerobotic ground station (Fig. 20 and Fig. 21).

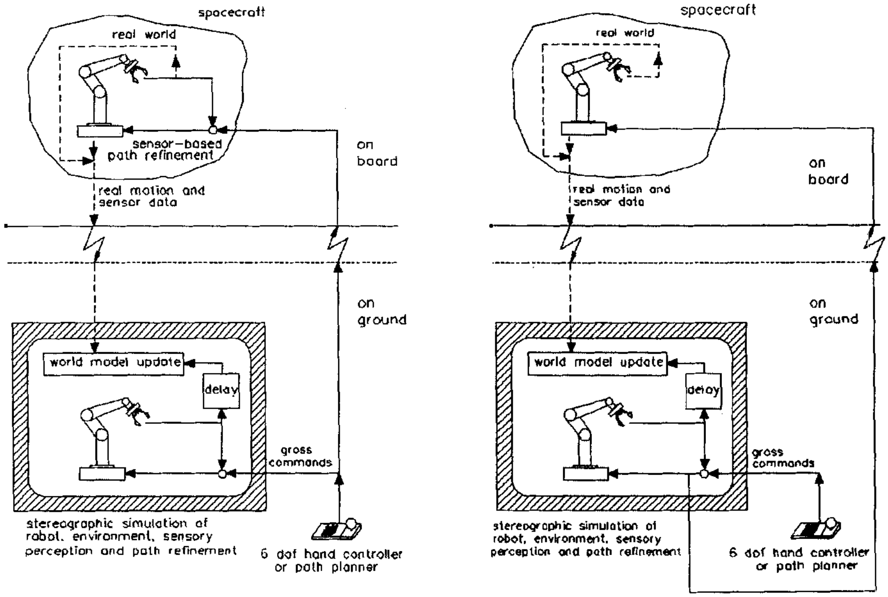

spacecraft
real world
sensor-based path refinement
real motion and sensor data
on board
on ground
world model update
delay
gross commands
stereographic simulation of robot, environment, sensory perception and path refinement
6 dof hand controller or path planner
spacecraft
real world
real motion and sensor data
on board
on ground
world model update
delay
gross commands
stereographic simulation of robot, environment, sensory perception and path refinement
6 dof hand controller or path planner

Fig. 20 Predictive simulation of sensory perception and path refinement
a) local on-board sensory feedback b) sensory feedback via ground station
(e.g. tactile contact) (grasping the free-flyer)

All sensory systems of ROTEX worked perfectly and the deviations between pre-simulated and real sensory data were minimal (Fig. 22), thus, allowing perfect remote control via the predicted virtual world.

There was only one exception from the local sensory feedback concept in ROTEX. It refers to (stereo-) image processing. In the definition phase of ROTEX no space qualified image processing hardware was available; nevertheless, we took this as a real challenge for the experiment "catching a free-floating object from ground". Hand-camera images were processed on ground in real-time to yield "measured" object poses and to compare them with estimates as issued by an extended Kalman filter that simulated the up- and down-link delays as well as robot and free-flyer models (Fig. 8). This Kalman filter predicted the situation that would occur in the spacecraft after the up-link delay had elapsed and, thus, allowed to close the "grasp loop'' either purely operator controlled, or purely autonomously (Fig. 23).

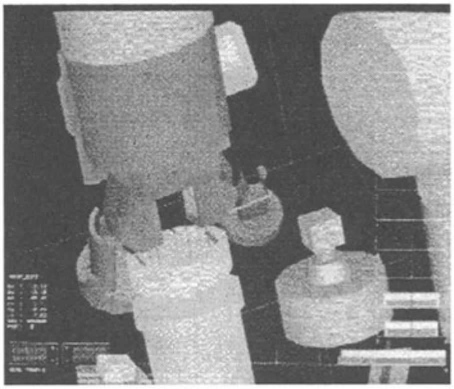


Fig. 21 Predictive Simulation of sensory perception in the telerobotic ground station

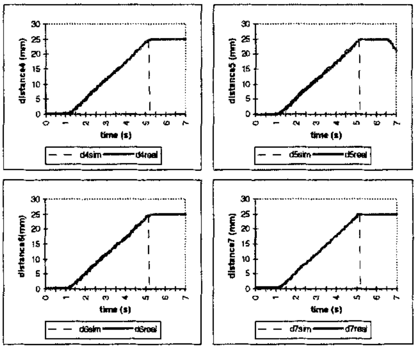


Fig. 22 Nearly perfect correlation between pre-simulated and real range finder data (removal from the bayonet closure)

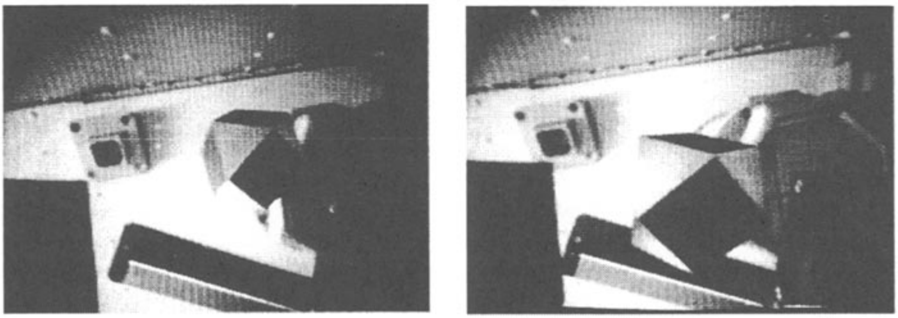


Fig. 23 Two subsequent TV-images out of one of the hand cameras shortly before successful, fully autonomous grasping of the free flyer from ground

Summarizing, the key technologies for the success of ROTEX have been

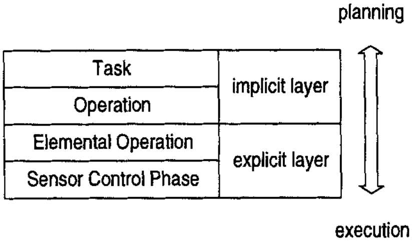
- the multisensory gripper technology, which worked perfectly during the mission.
- local, *shared autonomy,* sensory feedback, refining gross commands autonomously
- powerful delay-compensating 3D-stereo-graphic simulation (predictive simulation), which included the robot's sensory behaviour.

5 Advances in telerobotics: Task-directed sensor-based teleprogramming

After ROTEX we have focused our work in telerobotics on the design of a high-level task-directed robot programming system, which may be characterized as **learning by showing in a virtual environment** [19]; comparable concepts have e.g. been developed by R. Paul et. al [27]. The goal was to develop a unified concept for

- a flexible, highly interactive, **on-line programmable teleoperation station** as well as
- an **off-line programming tool**, which includes all the sensor-based control features as tested already in ROTEX, but in addition provides the possibility to program a robot system on an **implicit, task-oriented level.**

A non-specialist user - e.g. a payload expert - should be able to remotely control the robot system in case of internal servicing in a space station (i.e. in a well-defined environment). This requires a sophisticated man-machine-interface, which hides the robot control details and delivers an intuitive programming interface. However, for external servicing (e.g. the repair of a defect satellite) high interactivity between man and machine is requested. To fulfill the requirements of both application fields, we have developed a 2in2-layer-model, which represents the programming hierarchy from the executive to the planning level.



Task	implicit layer
Operation	
Elemental Operation	explicit layer
Sensor Control Phase	

Fig. 24 2in2-layer-model

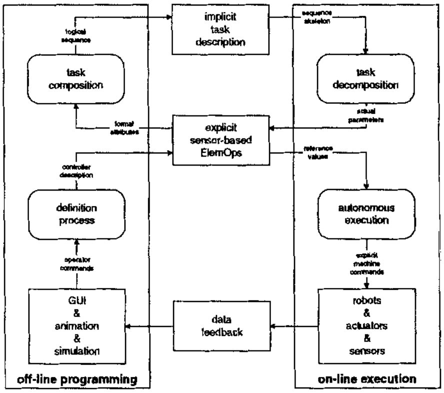


Fig. 25 Task-directed sensor-based programming (right)

On the implicit level the instruction set is reduced to **what** has to be done. No specific robot actions will be considered at this task-oriented level. On the other hand the robot system has to know **how** the task can be successfully executed, which is described in the explicit layers.

Sensor controlled phases

On the lowest programming and execution level our **tele-sensor-programming** (TSP) concept [19] consists of so-called *SensorPhases*, as partially verified in the local feedback loops of ROTEX. They guarantee the local autonomy at the remote machine's side. TSP involves ***teaching by showing*** the reference situation, i.e. by storing the nominal sensory patterns in a virtual environment and generating reactions on deviations. Each SensorPhase is described by

- **a controller function**, which maps the deviations in the sensor space into appropriate control commands. As the real world in the execution phase does not perfectly match with the virtual reference in the programming phase, we have developed two types **of mapping from non-nominal sensory patterns into motion commands** that „servo" the robot into the nominal situation: a **differential Jacobian**-based and a **neural net** approach,

- a **state recognition component**, which detects the end conditions and decides with respect to the success or failure of a SensorPhase execution
- the **constraint frame information**, which supports the controller function with necessary task frame information to interpret the sensor data correctly (realizing shared control)
- a **sensor fusion** algorithm, if sensor values of different types have to be combined and transformed into a common reference system (e.g. vision and distance sensors).

Elemental operations

The explicit programming layer is completed by the Elemental Operation (*ElemOp*) level. It *integrates the sensor control facilities with position and end-effector control.* According to the constraint frame concept, the non-sensor-controlled degrees of freedom (dof) of the cartesian space are position controlled

- in case of *teleoperation* directly with a telecommand device like the SpaceMouse
- in case of *off-line programming* by deriving the position commands from the selected task. Each object, which can be handled, includes a relative approach position, determined off-line by moving the end-effector in the simulation and storing the geometrical relationship between the object's reference frame and the tool center point.

The ElemOp layer aims at a *manipulator-independent* programming style: if the position and sensor control function are restricted to the cartesian level, kinematical restrictions of the used manipulator system may be neglected. This implies the general ***reusability*** of so-defined ElemOps in case of changing the robot type or modifying the workcell. A model-based ***on-line collision detection*** algorithm supervises all the robot activities, it is based on a discrete workspace representation and a distance map expansion [20]. For global transfer motions a ***path planning*** algorithm avoids collisions and singularities.

Operations

Whereas the SensorPhase and ElemOp levels require the robot expert, the implicit, task-directed level provides a powerful man-machine-interface for the non-specialist user. We divide the implicit layer into the Operation and the Task level.

An Operation is characterized by a sequence of ElemOps, which hides the robot-dependent actions. Only for the specification of an Operation the robot expert is necessary, because he is able to build the ElemOp sequence. For the user of an Operation the manipulator is ***fully transparent***, i.e. not visible.

We have categorized the Operation level into two classes:

- An ***Object-Operation*** is a sequence of ElemOps, which is related to a class of objects available within the workcell, e.g. GET <object>, OPEN <door>.
- A ***Place-Operation*** is related to an object, which has the function of a fixture for a handled object, e.g. INSERT <object> INTO <place>. <object> is the object, known from the predecessor Object-Operation, <place> the current fixture, to which the object is related.

Each object in the environment can be connected with an Object- and/or Place-Operation. Because an Operation is defined for a class of objects, the ***instantiation*** of formal parameters (e.g. the approach frame for the APPROACH-ElemOp) has been done during the connection of the Operation with the concrete object instance (Fig. 25). To apply the Operation level, the user only has to select the object/place, which he wants to handle, and to start the Object-/Place-Operation. For that reason the programming interface is based on a ***virtual reality*** (VR) environment, which shows the workcell without the robot system (Fig. 26). Via a 3D-interface (DataGlove or a 3D-cursor, driven by the Space Mouse) an Object/Place is selected and the corresponding Operation started. For supervision the system shows the state of the Operation execution, i.e. the ElemOp, which is currently active, as well as the pose of the currently moved object; and it hands over control to the operator in case of an error automatically. To comply with this autonomy requirement it is necessary to decide after each ElemOp how to go on depending on the currently sensed state.

- We admit sets of ***postconditions*** for each ElemOp. Each postcondition of such a set describes a different state. This can be error states or states that require different ways to continue the execution of an Operation.
- We extend the specification of an Operation as a linear sequence of ElemOps to a ***graph*** of ElemOps. If the postcondition of the current ElemOp becomes true, the Operation execution continues with the ElemOp in the graph belonging to the respective postcondition: POSTCOND(PRED) = PRECOND(SUCC)
- As the execution of each ElemOp requires a certain state before starting, we have introduced so-called ***preconditions***. For each ElemOp a set of preconditions is specified, describing the state under which the respective ElemOp can be started.

A formal language to specify the post- and preconditions has been defined, a parser and an interpreter for the specified conditions has been developed. Also the framework to supervise the Operation execution as well as a method to visualize the conditions has been implemented.

Tasks

Whereas the Operation level represents the subtask layer, specifying complete robot tasks must be possible in a task-directed programming system. A ***Task*** is described by a consistent sequence of Operations. To generate a Task, we use the VR-environment as described above. All the Operations, activated by selecting the desired objects or places, are recorded with the respective object or place de scription.

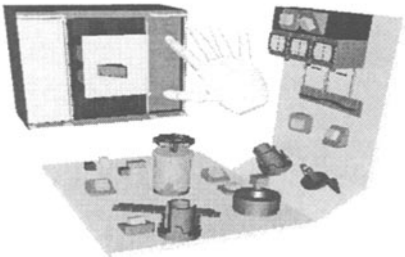


Fig. 26 VR-environment with the ROTEX-workcell and the Universal Handling Box, to handle drawers and doors and peg-in-hole-tasks

Fig. 27 DLR's new telerobotic station

Our task-directed programming system with its VR-environment provides a man-machine-interface at a very high level i.e. without any detailed system knowledge, especially w.r.t. the implicit layer. To edit all four levels as well as to apply the SensorPhase and ElemOp level for teleoperation, a sophisticated graphical user interface based on the OSF/Motif standard has been developed (Fig. 27 bottom, screen down on the left). This GUI makes it possible to switch between the different execution levels in an easy way. Fig. 27 shows different views of the simulated environment (far, near, camera view), the Motif-GUI, and the real video feedback, superimposed with a wireframe world model for vision-based world model update („augmented reality“, top screen up on the right).

The current sensor state, fed back to the control station, is used to reconcile the VR-model description with the real environment. This ***world model update*** task uses contactless sensors like stereo camera *images* or laser range finders with *scanner* functionality, as well as contact sensing. A method to perform *surface reconstruction,* merging the data from different sensors has been developed and successfully tested. This process is crucial for modelling of unknown objects as well as for a reliable recognition and location method.

6 Human interfaces for robot-progamming via Skill-Transfer

The consequent next step in our telerobotic concept which is based on learning by showing, is the integration of skill-transfer modules. A promising approach to solve this problem is the observation of an expert performing the manual task, collecting data of its sensorimotion and generating programs and controllers from it automatically that are then executed by the robot. In this approach the correspondence problem has to be solved, since the human executes an action $\boldsymbol{u}(t)$, due to a perception $\boldsymbol{S}(t - \mathrm{r})$. In [22] we show that the correspondence problem is solved by approximating the function

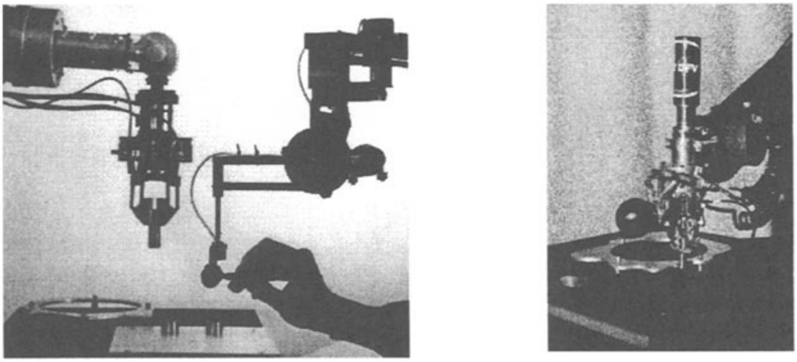
$$\boldsymbol{u}(\mathrm{t}) = \boldsymbol{f}\,[\boldsymbol{S}(t),\ \dot{\boldsymbol{x}}\,(t)]$$

with a neural network, taking the velocity $\dot{\boldsymbol{x}}\,(t)$ as an additional input to the network. Therefore, no reassembling of the data samples is necessary, taking into account which sensor pattern $\boldsymbol{S}$ perceived at time $(t - \mathrm{r})$ caused the operator to command an action $\boldsymbol{u}$ at a later time t. Nor is it necessary to determine the dead time r of the human feedback. Since humans describe tasks in a rule based way, using the concept of fuzziness, we have designed a neural network, prestructured to represent a fuzzy controller The control law, determining the compliant motion, is acquired by the learning structure.

With man-machine interfaces for human-robot interaction as developed at our institute a robot task can be „naturally“ demonstrated by an expert in three different ways:

1. **In cooperation with a powered robot system:**
   The assembly object is grasped by the robot and the robot is controlled by the expert in a direct or indirect type cooperating manner (Fig. 14 to Fig. 17).
2. **By interaction with a virtual environment:**
   (e.g. in the telerobotics station). The object is virtual and manipulated by the expert through some virtual display device.
3. **By using his hands directly:**
   The object is directly manipulated by muscle actuation of the expert and forces and motion are measured by a teach device.

Powered robot systems can be controlled by indirect cooperating type interface systems, using visual feedback of the TCP movement, e.g. by the desktop interface or a teach panel integrated Space Mouse as well as using haptic feedback of inertia and external forces on the robot through a force-reflecting hand controller interface (Fig. 28 (a)).



(a) (b)

Fig. 28 Teaching compliant motion in cooperation with a powered robot system: by indirect interaction, e.g., using a haptic interface (a), and by direct interaction, e.g., guiding the robot with a robot mounted sensor (b).

To demonstrate the task without using the robot system, haptic interfaces and virtual environments can be used. Fig. 29 (a) shows the PHANToM haptic interface[21] connected to our telerobotics environment [22]. The user can manipulate the object in the virtual environment. Such systems require the simulation of forces and moments of interaction between the object and virtual environment.

The most natural way of demonstrating the task is certainly to let the operator perform the task by his/her hands directly. Then the sensor pattern and motion of the operator have to be recorded. This can be done by observing the human with a camera [23] or by measuring manipulation forces of the operator. For the latter case we designed a teach device which acquires the motion and forces of the operator directly [24]. It is similar but much lighter than the device used by Delson and West [25]. We mounted a commercially available position tracking sensor (Polhemus Isotrack II), and a force/torque sensor module of the Space Mouse on a posture (Fig. 29 (b)). The workpiece to be assembled is plugged into the standard interface shaft of the device. The operator performs the task by grasping the force/torque sensor control device. Before recording the compliant motion the weight of the device is compensated.

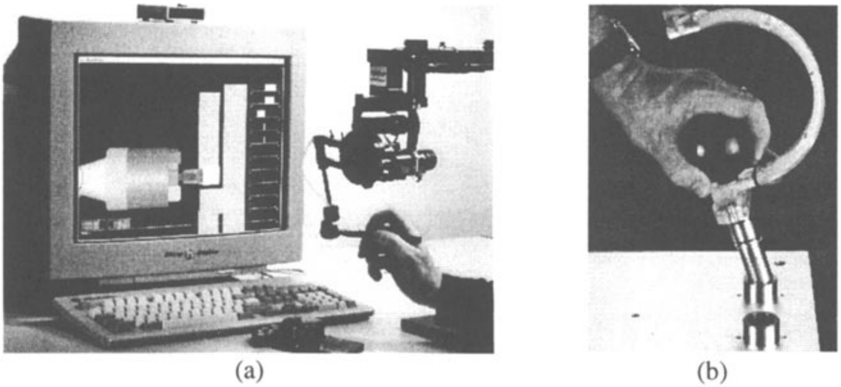


Fig. 29 Teaching compliant motion in a natural way: by interaction with a virtual environment using a haptic interface (a) and by collecting force and motion data from the operator directly using a teach device (b)

More recent work involves training stable grasps of our new 4 fingered hand, presumably the most complex robot hand that has been built so far with 12 position / force controlled actuators („artificial muscle") integrated into fingers and palm, 112 sensors, around 1000 mechanical and 1500 electronic components (Fig. 31). As input device for skill-transfer we use a data-glove here.

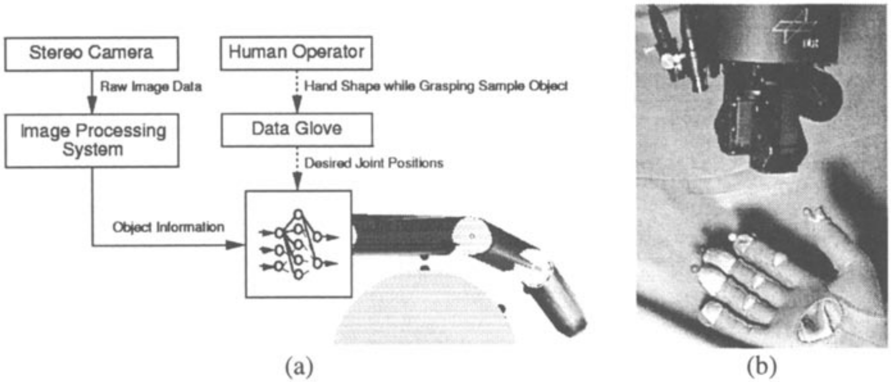


Fig. 30 The learning system for pre-shaping. (b) Data glove calibration by extracting the color marks mounted on the tip of the fingers of the glove

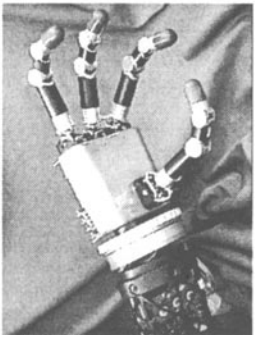


Fig. 31 DLR's new 4 fingered hand

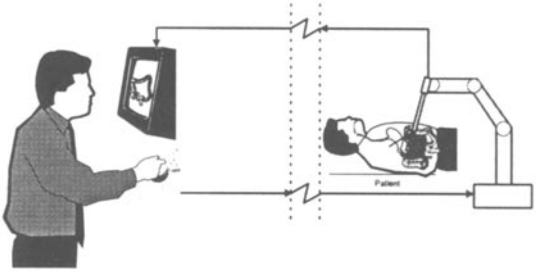


Fig. 32 Active telepresence in medicine

7 Robotics goes WWW - Application of Telerobotic Concepts to the Internet using VRML and JAVA

Teleoperation of robots over huge distances and long time delays is still a challenge to robotics research. But due to lack of widespread standards for virtual reality worlds most current teleoperation systems are proprietary implementations. Thus their usage is limited to the expert who has access to the specific hard- and software and who knows to operate the user interface.

New chances towards standardization arise with VRML 2.0, the newly defined Virtual Reality Modeling Language. In addition to the description of geometry and appearance it defines means of animations and interactions. Portable Java scripts allow programming of nearly any desired behavior of objects. Even network connections of objects to remote computers are possible. Currently we are implementing some prototype applications to study the suitability of the VRML technology to telerobotics.

We have implemented a VRML teleoperation which is based on the third layer of complex *operations* within our four layered telerobtic approach (section 5). This layer had been provided with a network transparent protocol named Marco-X. In this scenario the robot is operated through our telerobotic server, which implements layer 0 to 2 of the above hierarchy and the Marco-X protocol. The remote user again downloads the VRML scene, which shows the work cell as used during the ROTEX experiment in space, especially its movable objects and the robot gripper, but not the robot itself. The user may now pick objects and place them in the scene using the mouse. As interactions are done, Marco-X commands are generated and sent to the remote server which executes them. The current placement of objects is continuously piped back into the virtual scene where the objects now seem to be moved ghostlike. Using a task oriented protocol is the preferable method to remotely operate robots as it demands only extreme narrowband connections and doesn't bother about long time delays. It also provides a simple and intuitive method to interact with the virtual world as the user defines *what* he want's to be done, not *how* it has to be done. The main drawback is that possible manipulations are limited to the predefined ones.

8 Perspectives for the future

Telerobotic control and virtual reality will have many applications in areas which are not classical ones like space. An interesting example is laparoscopic surgery (minimal invasive surgery), where a robot arm may guide the endoscopic camera autonomously by servoing the surgeon's instruments. Such a system using realtime stereo colour segmentation, was successfully tested in the „Klinikum rechts der Isar“ Munich hospital, on animals as well as on humans [14]. It was found that the surgeon's concentration onto the surgery is massively supported by this technology. Now if the surgeon is not sure about the state of the organ he is investigating, he may call an expert somewhere in the world and ask him to teleoperate the camera-robot via ISDN-video transmission and active control with joystick or SPACE MOUSE (Fig. 32). We did these kind of experiments several times using e.g. two dof gastroscopes for real patients via arbitrary distances.

The consequent next step would be telesurgery using force reflection which is not discussed here in more detail. But we are sure that by using these telepresence and telerobotic techniques will massively influence medical care, as well as **teleservicing** in the field of mechanical engineering (maintenance, inspection and repair of machines over great distance) will have great impact in all export-dependent countries.

8.1 References

[1] T.B. Sheridan,
''Merging Mind and Machine'', Technology Review, 33-40, Oct. 1989.

[2] J.J. Craig,
Introduction to Robotics. Addison-Wesley Publishing Company, ISBN 0-201-10326-5, 1986.

[3] B. Hannaford,
"Stability and Performance Trade-offs in Bilateral Telemanipulation", *Proceedings IEEE Conference Robotics and Automation*, Scottsdale, 1989.

[4] W.S. Kim, B. Hannaford, A.K. Bejczy,
"Force Reflection and Shared Compliant Control in Operating Telemanipulators with Time Delay'' *IEEE Trans. on Robotics and Automation*, Vol. 8, No. 2, 1992.

[5] T. Yoshikawa,
Foundations of Robotics. MIT Press, ISBN 0-262-24028-9, 1990.

[6] D.E. Whitney,
"Force Feedback Control of Manipulator Fine Motions". *Journal of Dynamic Systems, Masurement and Control*, 91-97, 1977.

[7] G. Hirzinger, K. Landzettel,
"Sensory feedback structures for robots with supervised learning", *Proceedings IEEE Conference Robotics and Automation*, S. 627-635, St. Louis, Missiouri, 1985.

[8] S. Hayati, S.T. Venkataraman,
"Design and Implementation of a Robot Control System with Traded and Shared Control Capability", *Proceedings IEEE Conference Robotics and Automation*, Scottsdale, 1989.

[9] M.T. Mason,
"Compliance and force control for computer controlled manipulators". *Proceedings*

IEEE Trans. on Systems, Man and Cybernetics, Vol SMC-11, No 6, pp.418-432, 1981.

[10] A.K. Bejczy, W.S. Kim, St.C. Venema,
"The Phantom Robot: Predictive Displays for Teleoperation with Time Delay", *Proceedings IEEE Conference Robotics and Automation*, Cincinnati, 1990.

[11] L. Conway, R. Volz, M. Walker,
"Tele-Autonomous Systems: Methods and Architectures for Intermingling Autonomous and Telerobotic Technology", *Proceedings IEEE Conference Robotics and Automation*, Raleigh, 1987.

[12] P.G. Backes, K.S. Tso,
"UMI: An Interactive Supervisory and Shared Control System for Telerobotics", *Proceedings IEEE Conference Robotics and Automation*, Cincinnati, 1990.

[13] R. Lumia,
"Space Robotics: Automata in Unstructured Environment", *Proceedings IEEE Conference Robotics and Autmomation*, Scottsdale, 1989.

[14] G.Q. Wei, K. Arbter, and G. Hirzinger,
„Real-time visual servoing for laparoscopic surgery", *IEEE Engineering in Medicine and Biology*, vol 16, 1997.

[15] G. Hirzinger, J. Heindl
„Verfahren zum Programmieren von Bewegungen und erforderlichenfalls von Bearbeitungskräften bzw. -momenten eines Roboters oder Manipulators und Einrichtung zu dessen Durchführung'', Europ. Patent 83.110760.2302; „Device for programming movements of a robot", US-Patent 4,589,810

[16] J. Dietrich, G. Plank, H. Kraus
„In einer Kunststoffkugel untergebrachte optoelektronische Anordnung'', Deutsches Patent 3611 337.9, Europ. Patent 0 240 023; „Optoelectronic system housed in a plastic sphere", US-Patent 4,785,180

[17] G. Hirzinger and J. Heindl,
„Sensor programming, a new way for teaching a robot paths and forces/torques simultaneously", *International Conference on Robot Vision and Sensory Controls*, Cambridge, Massachusetts, USA, Nov. 1983.

[18] G. Hirzinger, B. Brunner, J. Dietrich, and J. Heindl,
„ROTEX - The first remotely controlled robot in space", *1994 IEEE International Conference on Robotics and Automation, San Diego, California*, 1994.

[19] B. Brunner, K. Landzettel, B.M. Steinmetz, and G. Hirzinger,
„Tele Sensor Programming - A task-directed programming approach for sensor-based space robots", Proc. *ICAR'95 7th International Conference on Advanced Robotics*, Sant Feliu de Guixols, Catalonia, Spain, 1995.

[20] E. Ralli and G. Hirzinger,
„A global and resolution complete path planner for up to 6DOF robot manipulators", *ICRA'96 IEEE Int. Conf. On Robotics and Automation*, Minneapolis, 1996.

[21] Thomas H. Massie and J. Kenneth Salisbury;
The phantom haptic interface: A device for probing virtual objects. *Proc. of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Chicago, November 1994.

[22] Ralf Koeppe and Gerd Hirzinger;
„Learning compliant motions by task-demonstration in virtual environments". *Fourth Int. Symp. on Experimental Robotics, ISER*, Stanford, June 30 -July 2 1995.

[23] Katsushi Ikeuchi, Jun Miura, Takashi Suehiro, and Santiago Conanto;
Designing skills with visual feedback for apo. In Georges Giralt and Gerd Hirzinger, editors, Robotics Research: *The Seventh International Symposium on Robotics Research,* pages 308--320. Springer-Verlag, 1995.

[24] Ralf Koeppe, Achim Breidenbach, and Gerd Hirzinger;
Skill representation and acquisition of compliant motions using a teach device. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS*, Osaka, November 1996.

[25] Nathan Delson and Harry West;
Robot programming by human demonstration: Subtask compliance controller identification. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, July 1993.

[26] S. Lee, G. Bekey, A.K. Bejczy,
"Computer control of space-borne teleoperators with sensory feedback", *Proceedings IEEE International Conference on Robotics and Automation*, S. 205-214, St. Louis, Missouri, 25-28 March 1985.

[27] J. Funda, R.P. Paul,
„Remote Control of a Robotic System by Teleprogramming", *Proceedings IEEE International Conference on Robotics and Automation,* Sacramento, April 1991

Cooperative Behavior between Autonomous Agents

Toshio Fukuda and Kosuke Sekiyama
Department of Micro System Engineering, School of Engineering
Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-01, Japan
Phone +81-52-789-4481, Fax. +81-52-789-3909
Email {fukuda, sekiyama}@.mein.nagoya-u.ac.jp

Abstract: Cooperative behavior is a central issue in distributed autonomous robot system (DARS). But, it has been discussed in case-dependent contexts. In this paper, we attempt to outlook cooperative behavior from viewpoint of the local and global coordination. Standing from the different point, basic methodologies are discussed.

1. Introduction

Toward realization of flexible systems, and from the interest regarding behavior science such as group behavior, the issues of cooperation have been attracting much research interest and becoming a central research topic in the field of Multi-Agent System (MAS). In the distributed AI (DAI) literature, the agent is often treated as a computational process. To save computational cost, the complicated problem is divided into several less complicated subproblems and they are executed by respective agent with cooperation. Contract net protocol developed by Smith is a representative cooperation technique for the distributed problem solver[1]. Several related problems and extended versions have been studied. In the meantime, the MAS has been rigorously discussed in the group robotics and complex system literature. These are dynamical system interacting with environment as well as between agents. Where, an autonomous agent is considered as an entity that possesses following features; 1) recognition, 2) decision making, 3) actuation. In this sense, biological individual is an agent in the ecosystem or society. Also, the *robot* is denoted as a programmable agent in the distributed autonomous robot system(DARS) or robot society. Hence, it should be underlined that the agent in the dynamical system is more physically based. Behavior of the agent is fully dependent on what it perceives as a sensing information and how it behaves under physical constraints from the environment and the other agents, such as resource, energy, and information. Therefore, the issue of cooperation involves physical contacts, time and space restrictions. Cellular robotic system which is composed of numerous functional unit and change its formation according to change of environment, has been proposed[2,3]. Also, several architecture for robot cooperation for static environment[4, 5] have been presented. However, arguments on cooperative system seem to be too case-dependent. In this paper, we attempt more comprehensive discussion on the issue of cooperative behaviors, introducing authors' recent work. Organization of the paper is as follows; section 2 described the classification of cooperation form, where intentional cooperation and self-organized cooperation are addressed. In section 3, the issue of intentional cooperation is focused. Distributed sensing are proposed for essential technique for multi-robot cooperation as an example. In section 4, self-organized coordination is discussed. In section 5, conclusions are drawn and future work is suggested.

2. Science of Cooperation

What is a cooperative behavior ? Though the question is seemingly simple, it is rather difficult to find a straightforward answer. There seems no definite objective criteria to estimate whether the behavior is cooperative. However, it can be said, at least, that the cooperation is a representation form of the intelligence produced by a group of agents. In general, we consider that the system is cooperative when its condition appears to be coordinated or balanced through the interaction between agents. In order for the system to evolve to the coordinated states or maintain harmonious condition, there seems to be different types of cooperation. Hence, we attempt to classify the "cooperation" as following types; *intentional cooperation* and *nonintentional cooperation.* In the following subsections, we describe that there are large difference between these cooperative systems in terms of design methodology.

2.1. Intentional Cooperation

Intentional cooperation is characterized by a metaphor, such as teamwork or collaboration, where agents work together for the *shared purpose* or a *common goal.* Therefore, the form of cooperation in this type is task-dependent. Tasks often considered are object handling, luggage transportation, resource sharing, cooperative navigation, environment exploration, and map building etc. Related issues are collision avoidance, deadlock avoidance and resolution, negotiation protocol, prediction, planning and scheduling, and distributed sensing etc. Learning abilities are also essential. These issues cover large part of domain in multiple robot system. However, despite a large number of papers deal with these issues, most of the work are too case-dependent and a general methodology has not been established. This indicates that cooperative behavior is pre-programmed in which task information is often explicitly incorporated to behavior strategy. In real world, cooperation form is situation-dependent as well as task-dependent. The same task may require a different cooperation form according to the change of environment. Therefore rigidly pre-programmed cooperation strategy is not effective for dynamical task environment. Evolutionary approaches are under explored in recent years. The scheme of intentional cooperation, which can deal with local and closed up problems in MAS, depends on the function and ability of individual agent in terms of hardware structure and intelligence level. Is it over statement that current intelligence level of the robot is much lower than expected for required task? In this sense, more rigorous research should be performed for realizing intelligent agent. Particularly, for more complicated tasks, task interpretation of other robots should be pursued for higher level cooperation. Also, cooperation between functionary heterogeneous agents should be more discussed. Where, different kind of sensory information is utilized. Distributed sensing which is discussed in section 3, becomes the fundamental technique for advanced cooperation strategy. In the next subsection, another type of cooperation for global coordination is discussed.

2.2. Emergence of Cooperative Behavior

Since the abilities of individual agent, in terms of perception and information capacity, are limited, it is difficult to consider that agents can share the common information in a large scale system. Without knowing the global information, is it possible to realize the global coordination ? Typical successful examples can be seen in the biological system, i.e., societies of ants and bees. It is hardly assumed that the ants or bees recognize their behavior purpose or any leaders exist to coordinate numerous autonomous agents. What is meant by nonintentional cooperation is thus the spontaneous emergence of coordinated or coherent states due to the self-organizing effects of a large group of interactions, which cannot be explained directly in the level of intentional behaviors. The paradigm of group intelligence or collective behavior has been attracting research interest from the view point of behavior science, as well as engineering applications. It is truly inter-disciplinary field of nonlinear physics, chemistry, biology, social science, robotics and the other engineering fields, etc. Self-organizing robot system (SORS) is a multi-ARS which exhibits emergence feature to coordinate its global behaviors. Research targets include realization of pattern formation in the group behavior which is explored by real multiple robots as well as simulation work, collective learning and evolution based on the biologically inspired approaches, and also philosophical consideration from the view point of system science. We believe that SORS has a great potential for realizing advanced flexible system. However, since theoretical foundation for design principle is still the stage of under explored, it will take time to reach the level of the use for engineering applications and to prove its powerfulness, except very limited cases. In research for the group robotics, we can have two types of possible view for robot. One is conventional view; a machine for executing a given task. In other words, that is application-oriented viewpoint which regards much on the aspect that how efficiency is improved by introducing the system. Another is principle-oriented viewpoint where the ARS is considered to be a testbed for the study of complex group behavior. For this research, the purpose is to understand and explore the mechanism of the cooperative behavior, where a hypothetical model is addressed and verified through construction using *programmable* agent; robot. Both of viewpoints are essential, but it should be careful that misunderstanding of these research aims can cause meaningless criticisms each other.

2.3. Role of Competition

One thing we have to add to consider the cooperative behavior is importance of *competition*. In conventional research, conflicts between agents are believed to exclude from the system. Numerous methods have been proposed to avoid and resolve competitions in various situations. However, it is interesting enough to observe that even the competition takes an essential role for global harmony in natural world. In large scale systems, we can raise many examples that some local conflicts and competitions often bring an important effect to global coordination. Relation between pray and predator is a fundamental for understanding ecosystem, where the local competition is indispensable for global harmony. One of the fundamental process in self-organization is a balance of positive-feedback and negative-feedback. The former implies to facilitate the growth of some factors, and the latter indicates reflection for the growth. Also, internally caused fluctuation becomes the trigger for the system to evolve. In this sense, the role of cooperative behavior is considered to be positive-feedback and competition is regarded as a negative-feedback or fluctuation. In particular, fluctuation is a cause of disorder, but it also give diversity to the system, that is, the chance of evolution. In designing self-organizing cooperative behavior, we must understand the relation of these processes in contrast with actual behavior and interaction rules. The role of competition is, thus, as important as that of cooperation for the global coordination of the MAS.

3. Cooperation with Distributed Sensing

3.1. Distributed Sensing in DARS[9]

Multiple robots are distributed in different space. What should be emphasized here is that individual robots must have the sensing ability to recognize world. Sensing actions are distributed in space. Through such a kind of sensing actions, bodies of sensing information from different view angles are obtained. To an individual robot in DARS, although several kinds of sensors may be equipped, its sensing ability in time, space and function is still limited. For this reason, besides requirements to the cooperation for actions of multiple robots, their sensing information should also be shared to enhance the reliability of whole system. Hence, when the sensing information belonging to different robots are shared through some links, a distributed sensing net as showed in Fig.1 can be formed. Distributed sensing centers on integration of sensing information of multiple robots. Although it is similar to multi-sensor system in some degree from viewpoint of integration of sensing information, differences between them are mainly embodied in integration structure and sensing information level describing environment states. Approach to integrate distributed sensing depends on the characteristics of sensing information of robots. When individual robots sense the environment, due to errors of sensor measurements and sensing system modeling, the information on environment inferred from sensor data is uncertain, and may be inconsistent. When robots locate in different locations, the sensing information may be incomplete. So, the primary characteristics of sensing information are uncertain, inconsistent and incomplete. On the other hand, in considering human beings reasoning, evidence supporting some propositions that comes from real observations is usually uncertain, incomplete and inconsistent. Robot sensing has also the characteristics similar to that of evidence supporting propositions, and can be refereed to be "evidential information". Evidential information leads that individual robots describe environment states in an uncertain way. In other words, when it infers based on its sensing, one robot usually can not explain its sensing information to one of environment states certainly, but to several possible environment states or subsets of the environment state set with uncertainty. Based on the analysis, we have every reason to formulate individual robots sensing about environment as followings.

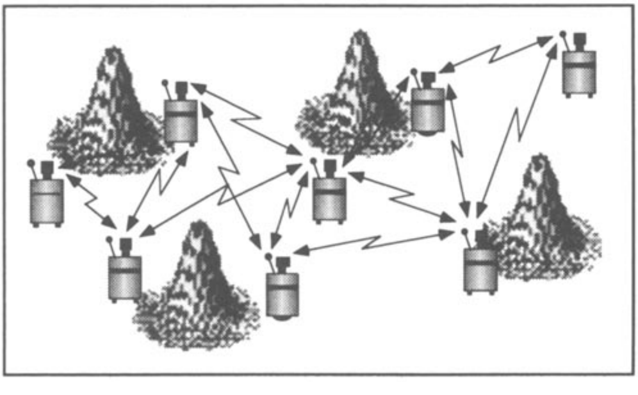


Fig.1 Distributed Sensing Net in DARS

Let Ω be the set of basic environment states relative to problem domain. Let O be the set of sensing of multiple robots.

$$\Omega = \{\omega_1, \omega_2, \ldots\ldots, \omega_N\} \qquad (3.1.1)$$

$$O = \{o_1, o_2, \ldots\ldots, o_M\} \qquad (3.1.2)$$

ω_i $(1 \leq i \leq N)$ is an environment state. $o_i (1 \leq i \leq M)$represents the sensing completed by robot i. So, to sensing o_i of robot i, possible explanations about environment states can be expressed as

$$A^i{}_k \subseteq \Omega \qquad 1 \leq k \leq K \qquad (3.1.3)$$

$A^i{}_k$ is a subset of Ω, an explanation about environment states based on o_i. Formula (3.1.3) means that there are K explanations about environment states that constitute an explanation structure concerned with sensing o_i of robot i. To different robot sensing, the explanation structures for the environment states will be different.

Formula (3.1.1)-(3.1.3) representing robot sensing and its explanations about environment states are consistent with representations of observations and explanations in evidential reasoning theory of Dempster-Shafer[10][11]. So, evidential reasoning is considered for processing of distributed sensing in this research. There are two important points in evidential approach for reasoning with uncertain knowledge different from other tools, such as probabilistic approach. One is assignments of belief with pieces of evidence to some propositions. This means that when one part of a belief is assigned to a proposition, it not necessary that the remaining belief must assign to the complement of the proposition, but to basic proposition space. This is particularly useful for distributed sensing. Because individual robot usually can not describe one of world states clearly due to its sensing uncertainty, it is natural for the sensing of individual robot to be considered related to several world states with different belief assignments. Another one is its powerful combination rule of evidence from different sources, which makes it possible to realize the integration of distributed sensing information effectively.

3.2 Basic Definition and Formulation

Explanations of sensing o_i, corresponding to set Ω are constructed in formula (3.1.3). The body of evidence relative to observation o_i assigns a measurement called evidence mass and expressed as $m_i(A^i{}_k)$ to explanation $A^i{}_k$. $m_i(A^i{}_k)$ represents the opinion that robot i thinks explanation $A^i{}_k$ is true, based on its sensing o_i. The result of assigning evidence mass to all explanations constitutes a belief structure expressed as $m_i()$relative to observation o_i over Ω. Belief structure $m_i()$ is characterized as

$$m_i(): 2^{\Omega} \Rightarrow [\,0,\ 1\,] \qquad (3.2.1a)$$

$$\sum_{A^i{}_k \subseteq \Omega} m_i(A^i{}_k) = 1 \qquad (3.2.1b)$$

$$m_i(\phi) = 0 \qquad (3.2.1c)$$

Generally, to different sensing, belief structures will be different. Based on evidence mass assignments over explanations, interval probability for proposition $A^i{}_k$ is calculated as followings.

$$Sp(A^i{}_k) = \sum_{A^i{}_l \subseteq A^i{}_k} m_i(A^i{}_l) \qquad (3.2.2a)$$

$$\mathrm{Pl}(A^i{}_k) = 1 - \sum_{A^i{}_l \subseteq \overline{A}^i{}_k} m_i(A^i{}_l) = 1 - \mathrm{Sp}(\overline{A}^i{}_k) \qquad (3.2.2b)$$

Sp() called Support is lower bound of interval probability indicating the lowest possible degree of evidence to support proposition $A^i{}_k$, while Pl() called plausibility is upper bound of interval probability indicating the highest possible degree of evidence to support proposition $A^i{}_k$. Decisions, such as analysis and judgment etc. related to the recognition of world states, are made based on interval probabilities. There exist a number of rules that can be selected for decision making through combining support and plausibility in different way. For example, there are: (1)Maximum Support Rule, (2)Maximum Plausibility Rule, (3)Maximum Support and Plausibility Rule.

3.3. Integration of Sensing Information

Based on Dempster's rule called orthogonal sum, bodies of evidence from multiple sources can be integrated. The integration is completed through the combination of different belief structures relative to different observations. Let $m_i()$ and $m_j()$ be two different belief structures relative to sensing o_i and o_j. the rule combines $m_i()$ and $m_j()$ to produce a new belief structure $m_{ij}()$ over Ω as following.

$$m_{ij}(A_n) = \frac{1}{1-T} \cdot \sum_{A^i{}_k \cap A^j{}_l = A_n} m_i(A^i{}_k) \cdot m_j(A^j{}_l) \qquad (3.3.1a)$$

$$(\text{ for all } A^i{}_k \cap A^j{}_l = A_n)$$

$$T = \sum_{A^i{}_k \cap A^j{}_l = \phi} m_i(A^i{}_k) \cdot m_j(A^j{}_l) \qquad (3.3.1b)$$

$$(\text{for all } A^i{}_k \cap A^j{}_l = \phi)$$

According to formula (3.3.1a) and (3.3.1b), the new evidence mass $m_{ij}(A_n)$ in the new belief structure $m_{ij}()$ corresponds to a new explanation $A_n(A_n \subseteq \Omega)$, which is the conjunction between a pair of explanations related to belief structure $m_i()$ and $m_j()$ respectively. Two essential conditions to above rule are: (1) bodies of evidence to be integrated must be independent; (2) these independent bodies of evidence must refer to the same Ω. In this research, condition (2) is usually contented when Ω is initially set. Condition (1) can also be contented in considering that sensing of one robot is not affected by other robot, and to the same robot, its sensing at one moment does not affect sensing at other moments. Because this rule is both commutative and associative, sensing information from multiple robots or in different time can be integrated by pairs in any order without the result being affected.

3.4. Application Example

According to above general framework, distributed sensing is examined in an application; environment map building and maintaining. For unknown or partially known environment, environment map building and maintaining is important through robot sensing. Environment map building and maintaining in CEBOT is typically concerned with distributed sensing.

3.4.1 Robot and Environment Settings

Basic settings for simulation are as followings.

(1) Environment is a 2-dimensional plane.
(2) Environment is modeled by GM (Grid Map).
(3) Sensors are equipped on robots.
(4) Communication devices are equipped on robots.
(5) Robots know their positions exactly.

The set Ω of basic problems relative robot sensing is set as $\Omega = \{\omega_1, \omega_2\} = \{\text{occ}, \overline{\text{occ}}\}$ in case environment is modeled by GM. occ and $\overline{\text{occ}}$ are states of cell in environment map. occ represents that cell is occupied and $\overline{\text{occ}}$ represents that cell is empty. Here, to the sensing of each individual robot, explanation structure corresponding to Ω is set to be same as followings.

Explanation 1: $A_1 = \{\text{ occ }\}$ (3.4.1a)

Explanation 2: $A_2 = \{\text{ }\overline{\text{occ}}\text{ }\}$ (3.4.1b)

Explanation 3: $A_3 = \Omega = \{\text{occ}, \overline{\text{occ}}\}$ (3.4.1c)

Figure 2 shows the flow chart of concrete realization of Integrating phase, concerned with World Model (environment map building and maintaining) and distributed sensing based on evidential reasoning.

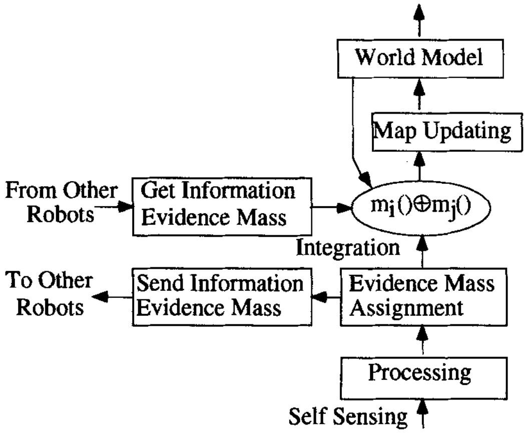


Fig.2 Flow Chart of Distributed Sensing Information Integration

3.5. Simulation Experiments

Simulation is carried out in an environment consisting of 100x100 cells as showed in fig.3. Evidence masses of explanation A_1, A_2 and A_3 for each cell in map are initially set as 0, 0, 1 respectively. So, Initial interval probability of explanation $A_1$for each cell in map is [0,1] which represents complete ignorance about environment. Three objects exist and three robots are used. Three robots start from initial positrons showed in Fig.4. Sensing is done while robots are moving. When 3 robots move to the positions showed in Fig.4, support(SP) and plausibility(PL) of explanation A_1 for each cell in map, based on sensing integration of 3 robots are shown in Fig.5 and Fig.6. In general, 3 situations for each cell in map are indicated with interval probability: (1) high SP and high PL value indicate that the possibility that cell is occupied is high; (2) low SP and low PL indicate that the possibility that cell is occupied is low; (3) low SP value and high PL (long interval between SP and

PL) show that the cell is ignorant in high degree. So, to the environment in this simulation, area where object 1 — object 3 exist, area where no object exists and area where it is still ignorant can be clearly understood according to fig.4 and fig.5.

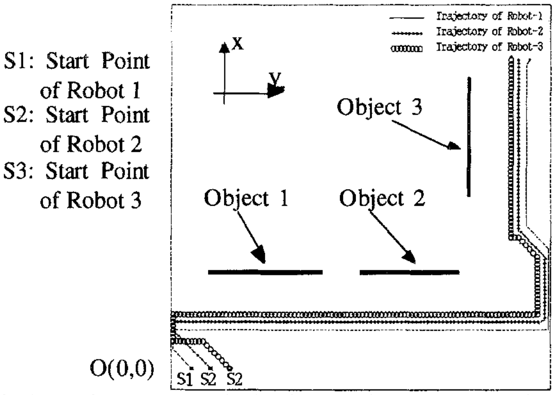


Fig.3 Environment and Moving Trajectories of Robots

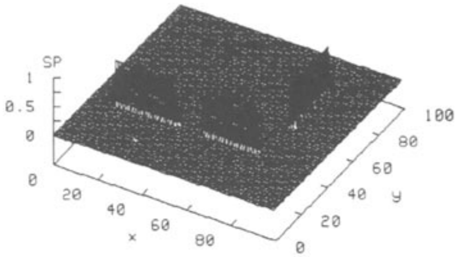


Fig.4 Lower bound (SP) of Interval Probability

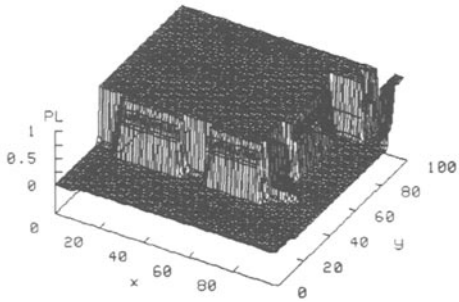


Fig.5 Upper bound (PL) of Interval Probability

4. Self-organizing Cooperative Behavior

Group behavior exhibited by self-organizing effects is not explicit intentional behavior of individual agents. The reason why we investigate this kind of behavior is following; robot society, which is composed of a large number of robotic agents with flexible structure and emergence capability, is the goal of research. Based on the local interactions which include information exchanges, intentional cooperation and some degree of competitions, not trivial socially intelligent behavior is expected for coordination of the large scale system and basic research has been performed[2,3,6,7,8].

In the living system of nature, such a example is abundant. This fact of the universality of complex behavior observed in interdisciplinary fields, gathers much interest and many results have been obtained in nonlinear science. In this section, we discussed what features of self-organization should be provided to design the MAS.

4.1. Approach to Self-Organizing System

4.1.1. Forward and Inverse Problem in Self-Organizing System

There are two basic stances for the research on collective behavior. From the viewpoint of scientific attitude, we are interested in why simple rules and mutual interactions can cause the coordinated group behavior. What kinds of relations and mechanisms should exist behind it. This can be called a forward problem for understanding processes of the self-organization. On the other hand, the inverse problem is fundamental from the viewpoint of engineering applications, where the self-organizing behavior must be designed to satisfy specific purpose expected to the system. What particular behavior and interaction rule should be imbedded to each agent for the purposive group behavior ? For this inverse problem, two approaches are considered. One is that the designer investigates the forward problem and attempt to utilize obtained principle to construct the system[8]. Another is to let the system solve it. This is what the living systems actually implement. For the moment, let us consider the former approach. In order to solve the inverse problem, the forward problem must be revealed. Unfortunately, the most of the research works on group behavior do not attempt to analyze and reveal the behind principle of emerging behavior, although some kinds of interesting behaviors are presented. They are often explained as a result of the learning or adaptation of each agent to the environment. But this does not give the fundamental answer, because what should be explained is a relation of processes for evolution. Hence, it should be noted that our knowledge is not yet enough to establish satisfactory group robot system for actual applications. Therefore, constructive approach is taken, where a conjecture model is constructed and tested repeatedly.

4.1.2. Outlook of Self-Organizing System

Designing self-organizing system starts from understanding its basic mechanism. But, so called *self-organization* is discussed in so wide ranging contexts and different meaning. To make clear our discussion, let us overview the theory and concept on it. There are several classes in self-organizing system, and what it denotes is somewhat different. The aim of discussion in this section is to give an outlook of the level and features of self-organizing system and consider which level of theory or concept should be applied to design the MAS. Hence, rough sketch of classification becomes as follows;

Level 1: Physical / Chemical System
Level 2: Biological System
Level 3: Social System

Common features among them are that 1) collective system and 2) spontaneous structure emergence. Let us discuss respective class in more detail.

1) Physical / Chemical System

This class is extensively investigated. Many mathematical theories and models are presented for complex pattern formation and chaotic behavior due to collective interactions. The component unit of this system is atom or molecular. The molecular is not deserved to be called as a *agent,* because it is not considered as autonomous. It dose not recognize environment, and just follows the physical and chemical laws of the nature. Physical and chemical system is not living system by itself. However, obtained results are often useful to understand a partial process of the activity of biological or social system when the system is analyzed from the macroscopic viewpoint, where heterogeneous characters of the agent are averaged and the dynamics of the system are often reduced to low dimensional for simplicity of the mathematical treatment. It is often pointed out that this treatment may erase fundamental properties to understand more complex behavior. Bifurcation and chaos, dissipative structure, synergetics, catastrophe theory etc. are well known theoretical framework.

2) Biological System

Biological system is undoubtedly the living system. The component of system is *cell* which can be considered as autonomous agent. Neural network, immune network are self-organizing system which is well investigated. Biological system is a network system composed of heterogeneous functional organization. Cell is also heterogeneous, but the interesting fact is that the characters of cell depends on its location in the system, in other words, it depends upon the characters of the other cells around itself. Biological system is thus, a *self-referential system.* It follows not only natural law, but also the genetic rules which are acquired in the evolutional processes. The largest difference between living system and nonliving system is ability of self-generation of the rules, to which the agents obey as well as the natural laws. Although these are observable fact, the mechanisms are still open problems in life science. Many of the theoretical basis are given from physical / chemical nonlinear science. But also, philosophical issues are often discussed. Autopoiesis has been presented for the definition of biological autonomy which concept is innovative, but it cannot cover all of the features of biological activity.

3) Social System

Obviously, the social organization is a living system and the most complex system in this classification of the self-organizing system. There are several advanced considerations for the social system, but mathematical formulation seems to be very difficult. Some attempts have been made to grasp a fragmented aspect of social phenomena by chaos and nonlinear dynamics. But there is much argument for the way of treatment which ignores diversity of characteristics of the agents. The social system has a self-referential property as well as the biological system. Compared with biological system, role of symbolic information of language, political balance, and self-organization of information network become more explicit.

Based on the consideration, it can be suggested that the common feature in self-organizing system as a living system is the ability of control of internal rules which define organization and disorganization of the system structures. Hence, the system structure indicates the rule structure as well as the physical structure. According to social science, the *social structure* denotes relation of the rules, such as social laws, custom, regulations etc. Based on the rule structure, the physical structure; spatial pattern of distribution or quantity of agents etc., evolves. Also, the rule structure is reorganized through negotiative process among the agents, which is dependent on the physical structure. This coupling self-organizing processes enable consecutive evolution in the context of the environment. The component unit of social system does not have to be a biological agent. There can be a social system composed of artificial autonomous agents, that is a robot society.

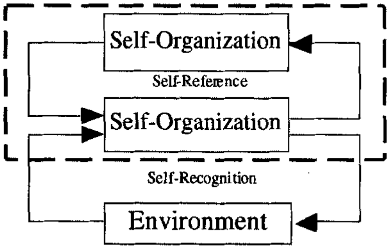
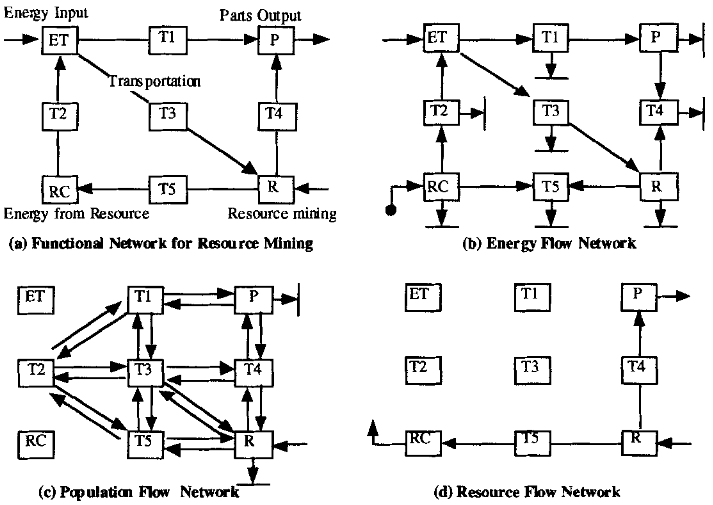


Fig.6 Coupling Structure of Self-Organizing Mechanism

4.2. Designing Social Group Behavior

4.2.1 Basic model description

Let us consider a minimum level organization network of resource mining task, as shown in fig.7-(a). The model represents the relation of functioning processes of robot society, that is, resource mining, resource and energy transportation, and parts assembling sites. The symbols used in fig.7 are defined in table 1. In this model, energy is supplied from environment to the site ET, and the energy is distributed to site P and R through path T1 and T3 respectively by the robots. At site R, robots execute resource mining task. The resource is transported to and consumed at site P or RC for the parts assembling and energy conversion. The converted energy at RC can be carried to ET for energy supplement. Environmental constraints to the system are twofold; constant energy supply from site ET and request of part production at site P. Parts are constantly removed from P. If too many robots work for parts production, the labor power to energy supply and resource mining decrease and this results in low performance of the system.



(a) Functional Network for Resource Mining

(b) Energy Flow Network

(c) Population Flow Network

(d) Resource Flow Network

Fig.7 Organization network structure for respective information currency

Thus, this network system requires task differentiation of a number of robots and self-coordination of internal population balance for respective task. Unlike conventional work of group robotics, robots in this model do not share common goal because desired condition at respective site is different from each other. The purpose of behavior for each robot is generated through internal observation of respective site.

4.2 Multi-layer Network Expression

Observation of the network structure is a projection based on particular criteria which are focused on. Hence, we are concerned with population, energy and resource flow in our model. Without saying, each robot behaves based on the local information of locating site and its neighbor condition, the spatial distribution of the population, which implies labor distribution in the network, evolves in the self-organizing manner. However, it should be noted that this self-organizing labor distribution depends on energy and resource distribution of the network. As well as quantitative amount, spatial distribution of energy and resource affects structure organization of population. On the other hand, since robots transport energy and resource from site to site, the distribution of energy and resource change based on population network. Thus, spatial distributions of energy, population and resource are closely coupled. Figure 8 represents the conception of this relation. In our model, fig.7-(b), (c), and (d) represent energy flow network (EFN), population flow network (PFN), and resource flow network(RFN), as a projection of the structure fig.4-(a).

Table 1 Role of the site

ET	Energy Tank site
P	Parts assembling site
R	Resource mining site
RC	Resource to energy conversion site
Tx	Transportation paths (x=1 to 5)

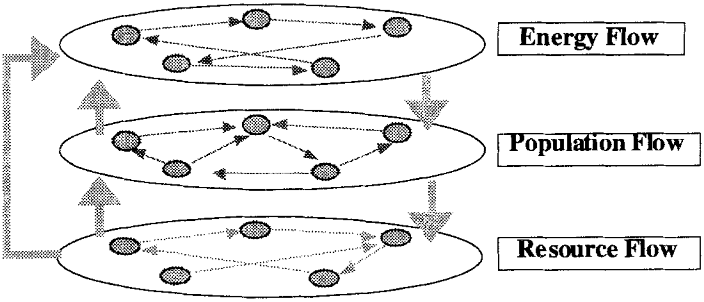


Fig.8 Interdependence of network structure

4.2.3 Nonequilibrium and Purpose generation

In general, the robot works so that a purpose is achieved. However, this model dose not prepare a priori defined purpose for the robots. The task and purpose for each robot are determined in a context of situation. There is an even the case that some robots do nothing. The only thing what robots do is to reduce the difference between the current state and desired state. Task selection of the robot depends on the site where the robot is currently located. For example, if the robot is located in the site P, and it recognizes shortage of the resource for parts assembling despite energy amount is satisfied, it heads to the site R via route T4 to supply resource to the site P. The recognition of the difference of state condition is performed based on eq.(4.3.1), and which is a driving force of purpose generation. Robots recognize amount of energy and resource of the neighboring sites, and decide next task type with eq.(4.3.2), which are given by,

$$\mu_i = \begin{cases} (x_i - x_c)^2 & \\ 0 & iff \quad x_i > x_c \end{cases} \qquad (4.3.1)$$

$$\Delta\mu_{ST} = \begin{cases} \mu_S - \mu_T & \\ 0 & iff \quad \mu_S > \mu_T \end{cases} \qquad (4.3.2)$$

where μ_i denotes shortage from critical value x_c of energy or resource at each site. Also, $\Delta\mu_{ST}$ represents gradient of the nonequilibrium potential of energy or resource amount between the sites. The decision of robot becomes different according to current location of the robot. Every condition and task selections is situation oriented. As is described in previous section, since energy and resource transportation are carried out by the robots, structure of EFN and RFN is organized based on the spatial pattern (structure) of PFN which denotes labor distribution on the network. Thus, the pattern of PFN is a constraint condition of EFN and RFN's organizing processes. On the other hand, since the labor distribution is reformed to facilitate solution of the nonequilibrium state, the structure of EFN and RFN are constraints conditions of reorganization of PFN structure as well. These networks are impossible to divide each other and have meaning only in the context of interrelation of the other network structures.

4.3 Simulation of Network Behavior

4.3.1 Structure configuration

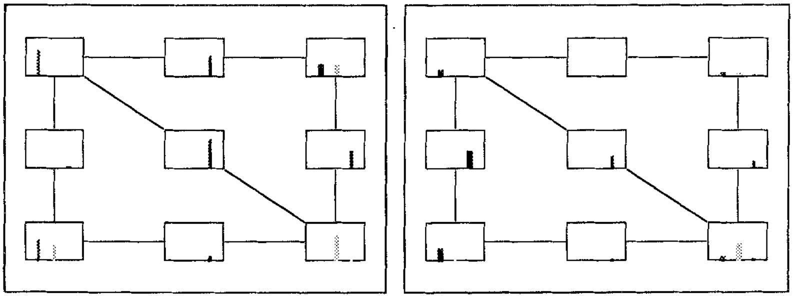
This section examines structure reorganization due to dynamical labor distribution. Time average labor distribution (PFN) characterizes behavior of the network system. If the energy supply is sufficient, labor power of the robot can concentrate to parts assembling task without considering energy amount and produce as many amounts of parts as required. PFN is organized such that parts production is maximized. On the other hand, if the energy supply is not enough, some ratio of robots should be devoted to the task related to energy conversion from the resource. This simulation is performed to examine that network robot system behaves as expected according to the energy supply. Population of the robot is 100. Other simulation setup values are shown in table 3. As shown in fig.9-(a), in case that sufficient amount of energy is supplied, most of the robots are afford to engage in the task related to the parts assembling work. However, in the case of lack of energy supply, the energy shortage at ET induces a nonequilibrium state at RC and a number of robots are distributed to configure self-complimentary structure of energy network as shown in fig.9-(b). More robots are engaged in energy transportation task rather than parts production task in this case. Thus, the robots decide how to use resource according to change of energy condition. It shows a characteristic of collective autonomy, where the group behavior work to maintain activity of the system. A case of that there are more choices as alternative conditions will be examined as a future work.

Table 2 Task type of robot ((*) denotes energy consumption for a behavior. cf. tab.3)

1	Resource mining (m)	5	Energy explore (e)
2	Resource supply (e)	6	Assemble parts (b)
3	Resource explore (e)	7	Doing nothing (0)
4	Energy supply (e)		

Table 3 Simulation setup value ([*] denotes unit of amount.)

1) total robot number N:	100
2) transportation step t [T]:	10.00
3) energy carrying/robot e [*l*]:	3.00
4) resource transportation r [*k*]:	1.00
5) energy consumption d [*l*]:	0.50
6) resource mining cost m [*l*]:	1.00
7) parts assembling cost b [*l*]:	2.00
8) resource exchange rate c [*l*/*k*]:	(8.0)
9) energy supply/time E [*l*/T]:	(3.00)
10) parts demand/time P [*p*]:	2.00


(a)

(b)

(a) In case of Sufficient Energy Supply (E=10) (b) In case of Insufficient Energy Supply (E=3) (From left to right, bar denotes amount of energy, resource, and population at each site.)

Fig. 9 Time average of Network Structure

4.3.2 Diversity, Redundancy, and Optimality

When we consider what is a good system, we will find that the answer is situation dependent. This section discusses the relation between diversity and optimality. To show this, let us consider a network structure where the site R is disconnected to RC as shown in fig.10 (Type II). Hence, let us call the network structure of fig.7(a) as type I and the one without RC as type II . With this network structure, fig.11 shows the comparison result of the total parts production from the site P during 1000 time steps by changing energy supply E. The other setup values are the same as the one as shown in table 3. In the fig.11, production-opt denotes total output of the network type II, while the others are results in case of type I. Both systems of type I and type II require energy support to ET. When energy supply is sufficient, type II network is more efficient in terms of productivity, because RC (energy conversion) is a redundant part in this case. However, the performance of parts production is sharply affected to the energy supply. If energy supply is small, its productivity is deteriorated. On the other hand, since type I network distributes some parts of the labor powers for energy conversion task at RC even when the energy is supplied sufficiently, type I cannot perform better than type II in case of enough energy supply. Structure of Type I contains a redundant part (RC) in this case. However in the case of energy shortage, RC site of the system takes an essential role to maintain work ratio as shown in fig.12. In case of energy shortage, large parts of robot are engaged in energy supply task{T2,T3,R,T5,RC} rather than parts production task. Thanks to this collective coordination, energy shortage is covered to some degrees, and energy is utilized for executing parts assembling task as well as maintaining the robot activity. Productivity in type I is better than that of type II in case of energy shortage. It can be said that redundant part is a vital component in this case. So, these results show that there is a trade off between flexibility and optimality. In general, optimization is to reduce redundancy form the system performance, but it deprive potential ability of flexibility to dynamical change of environment. Conventional systems are designed under assumption that necessary conditions are always satisfied and only considered optimization of performance. But this is not always guaranteed when we consider highly dynamical system, such as the robot society. It should be firstly considered whether the system can maintain its fundamental function when exposed dynamical change of environment.

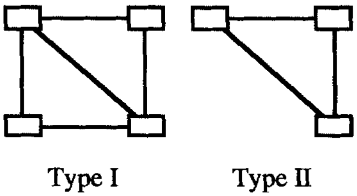


Fig. 10 Structure type of the Network

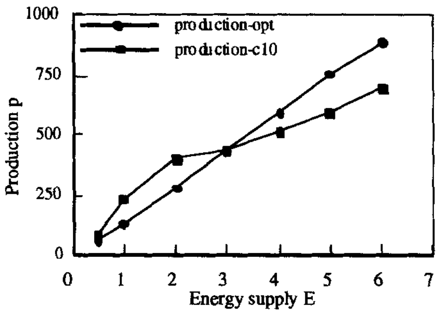


Fig.11 Total parts production during 1000 time

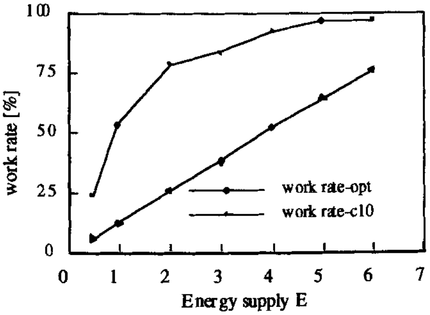


Fig.12 Average work rate of total population

5. Summary

In this paper, the issue of the cooperative behavior was discussed from local to global coordination. For the local coordination, intentional cooperation is central problem. As a basic technique for cooperation, this paper presented the information sharing between heterogeneous agent and distributed sensing. On the other hand, considering social robotics, we discussed how collective behavior should be designed. The self-organizing collective behavior is fundamental for the global coordination. There are several classes in self-organization. It should be more rigorously discussed that which class of self-organization is fundamental for our concerning system as a design principle, which may be different according to required task and environment. Also, relation between flexibility and optimality of the system performance are subtle problem, because these sometimes contradict with each other. Flexible system is robust for dynamical environment, but not necessarily optimal for specific task. How do we consider the balance of the system properties? In this paper, two classes of cooperative behaviors were treated independently. However, these are not separated problems each other. Local behavior cause a group behavior, but it is influenced by the group behavior. We must consider the coevolution between local interaction and group behavior. This is very interesting and essential issue for social agent system. The competition between agents may be also important to facilitate evolution. Our fundamental aim is to link these issues as the general framework.

References

[1] Smith,Reid G. and Davis, R.,Framework for cooperation in Distributed Problem Solving, IEEE Trans. Sys. Man. Cybern., SMC-11-1, 61-70, 1981.

[2] Fukuda, T., and Ueyama, T., Cellular Robotics and Micro Systems, World Scientific in Robotics and Automated Systems-Vol.10, World Scientific, 1994.

[3] J. Wang and G. Beni, "Cellular Robotic Systems: Self-Organizing Robots and Kinetic Pattern Generation", Proc. of IROS, pp.139-144, 1988.

[4] Noreils, Fabrice R. and de Nozay, Route, An Architecture for Cooperative and Autonomous Mobile Robots, Proc. of International Conference on Robotics and Automation, pp.2703-2710,1992.

[5] Parker, L., E., Multi-Robot Team Design for Real-World Applications, Distributed Autonomous Robot Systems (DARS) 2, pp.91-102, 1996.

[6] Mataric, M. J., "Issues and Approaches in the Designing of Collective Autonomous Agents," Robotics and Autonomous Systems, Vol. 16, 321-331, 1995.

[7] Marco Dorigo, Vittorio Maneiezzo, and Alberto Colorni, The Ant System: Optimization by a colony of cooperative agents, IEEE Transactions on Systems, Man, and Cybernetics-part B, Vol.26.' No.1, pp.1-13, 1996.

[8] Sekiyama, K. and Fukuda, T., Modeling and Controlling of Group Behavior on Self-Organizing Principle, Proc. of IEEE International Conference on Robotics and Automation,pp.1407-1412, 1996.

[9] A. Cai, T. Fukuda and F. Arai, "Integration of Distributed Sensing Information in DARS based on Evidential Reasoning", Proc. of 3rd International Symposium on Distributed Autonomous Robotic Systems, pp.268-279, 1996.

[10] A. P. Dempster, A generalization of Bayesian inference, *J. Roy. Stat. Soc.*, vol. 30, pp. 205 ~ 247, 1968.

[11] G. Shafer, *A Mathematical Theory of Evidence*, Princeton, NJ: Princeton Uni. Press, 1976.

Mobile Manipulator Systems*

Oussama Khatib
Stanford University
Stanford, CA, 94305, USA
khatib@cs.stanford.edu

Abstract: Mobile manipulation capabilities are key to many new applications of robotics in space, underwater, construction, and service environments. In these applications, consideration of vehicle/arm dynamics is essential for robot coordination and control. This article discusses the inertial properties of holonomic mobile manipulation systems and presents the basic strategies developed for their dynamic coordination and control. These strategies are based on extensions of the *operational space formulation*, which provides the mathematical models for the description, analysis, and control of robot dynamics with respect to the task behavior.

1. Introduction

A central issue in the development of mobile manipulation systems is vehicle/arm coordination [1,2]. This area of research is relatively new. There is, however, a large body of work that has been devoted to the study of motion coordination in the context of kinematic redundancy. In recent years, these two areas have begun to merge [3], and algorithms developed for redundant manipulators are being extended to mobile manipulation systems. Typical approaches to motion coordination of redundant systems rely on the use of pseudo- or generalized inverses to solve an under-constrained or degenerate system of linear equations, while optimizing some given criterion. These algorithms are essentially driven by kinematic considerations and the dynamic interaction between the end-effector and the manipulator's internal motions are ignored.

Our approach to controlling redundant systems is based on two models: an *end-effector dynamic model* obtained by projecting the mechanism dynamics into the operational space, and a *dynamically consistent force/torque relationship* that provides decoupled control of joint motions in the null space associated with the redundant mechanism. These two models are the basis for the dynamic coordination strategy we are implementing for the mobile platform.

Another important issue in mobile manipulation concerns cooperative operations between multiple vehicle/arm systems. Our study of the dynamics of parallel, multi robot structures reveals an important additive property. The effective mass and inertia of a multi-robot system at some operational point are shown to be given by the sum of the effective masses and inertias associated with the object and each robot. Using this property, the multi-robot system

*presented at RoManSy'96, the 11th CISM-IFToMM Symposium, Udine, Italy.

can be treated as a single *augmented object* [5] and controlled by the total operational forces applied by the robots. The control of internal forces is based on the *virtual linkage* [6] which characterizes internal forces.

2. Operational Space Dynamics

The joint space dynamics of a manipulator are described by

$$A(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \mathbf{\Gamma}; \tag{1}$$

where $\mathbf{q}$ is the n joint coordinates and $A(\mathbf{q})$ is the $n \times n$ kinetic energy matrix. $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})$ is the vector of centrifugal and Coriolis joint-forces and $\mathbf{g}(\mathbf{q})$ is the gravity joint-force vector. $\mathbf{\Gamma}$ is the vector of generalized joint-forces.

The operational space equations of motion of a manipulator are [4]

$$\Lambda(\mathbf{x})\ddot{\mathbf{x}} + \mu(\mathbf{x}, \dot{\mathbf{x}}) + \mathbf{p}(\mathbf{x}) = \mathbf{F}; \tag{2}$$

where $\mathbf{x}$, is the vector of the m operational coordinates describing the position and orientation of the effector, $\Lambda(\mathbf{x})$ is the $m \times m$ kinetic energy matrix associated with the operational space. $\mu(\mathbf{x}, \dot{\mathbf{x}})$, $\mathbf{p}(\mathbf{x})$, and $\mathbf{F}$ are respectively the centrifugal and Coriolis force vector, gravity force vector, and generalized force vector acting in operational space.

3. Redundancy

The operational space equations of motion describe the dynamic response of a manipulator to the application of an operational force $\mathbf{F}$ at the end effector. For non-redundant manipulators, the relationship between operational forces, $\mathbf{F}$, and joint forces, $\mathbf{\Gamma}$ is

$$\mathbf{\Gamma} = J^T(\mathbf{q})\mathbf{F}; \tag{3}$$

where $J(\mathbf{q})$ is the Jacobian matrix.

However, this relationship becomes incomplete for redundant systems. We have shown that the relationship between joint torques and operational forces is

$$\mathbf{\Gamma} = J^T(\mathbf{q})\mathbf{F} + \left[I - J^T(\mathbf{q})\overline{J}^T(\mathbf{q})\right]\mathbf{\Gamma}_0; \tag{4}$$

with

$$\overline{J}(\mathbf{q}) = A^{-1}(\mathbf{q})J^T(\mathbf{q})\Lambda(\mathbf{q}); \tag{5}$$

where $\overline{J}(\mathbf{q})$ is the *dynamically consistent generalized inverse* [5] This relationship provides a decomposition of joint forces into two dynamically decoupled control vectors: joint forces corresponding to forces acting at the end effector ($J^T\mathbf{F}$); and joint forces that only affect internal motions, $\left([I - J^T(\mathbf{q})\overline{J}^T(\mathbf{q})]\mathbf{\Gamma}_0\right)$.

Using this decomposition, the end effector can be controlled by operational forces, whereas internal motions can be independently controlled by joint forces that are guaranteed not to alter the end effector's dynamic behavior. This relationship is the basis for implementing the dynamic coordination strategy for a vehicle/arm system.

The end-effector equations of motion for a redundant manipulator are obtained by the projection of the joint-space equations of motion (1), by the *dynamically consistent* generalized inverse $\overline{J}^T(\mathbf{q})$,

$$\overline{J}^T(\mathbf{q})\,[A(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \Gamma] \quad\Longrightarrow\quad \Lambda(\mathbf{q})\ddot{\mathbf{x}} + \mu(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{p}(\mathbf{q}) = \mathbf{F}; \tag{6}$$

The above property also applies to non-redundant manipulators, where the matrix $\overline{J}^T(\mathbf{q})$ reduces to $J^{-T}(\mathbf{q})$.

4. Vehicle/Arm Coordination

In our approach, a mobile manipulator system is viewed as the mechanism resulting from the serial combination of two sub-systems: a "macro" mechanism with coarse, slow, dynamic responses (the mobile base), and a relatively fast and accurate "mini" device (the manipulator).

The mobile base referred to as the *macro structure* is assumed to be holonomic. Let Λ be the *pseudo kinetic energy matrix* associated with the combined macro/mini structures and Λ_{mini} the operational space *kinetic energy matrix* associated with the mini structure alone.

The magnitude of the inertial properties of macro/mini structure in a direction represented by a unit vector $\mathbf{w}$ in the m-dimensional space are described by the scalar [5]

$$\sigma_{\mathrm{w}}(\Lambda) = \frac{1}{(\mathbf{w}^T\Lambda^{-1}\mathbf{w})};$$

which represents the effective inertial properties in the direction $\mathbf{w}$.

Our study has shown [5] that, *in any direction* $\mathbf{w}$, *the inertial properties of a macro/mini-manipulator system* (see Figure 1) *are smaller than or equal to the inertial properties associated with the mini-manipulator in that direction:*

$$\sigma_{\mathrm{w}}(\Lambda) \leq \sigma_{\mathrm{w}}(\Lambda_{\text{mini}}). \tag{7}$$

A more general statement of this *reduced effective inertial* property is that the inertial properties of a redundant system are bounded above by the inertial properties of the structure formed by the smallest distal set of degrees of freedom that span the operational space.

The reduced effective inertial property shows that the dynamic performance of a combined macro/mini system can be made comparable to (and, in some cases, better than) that of the lightweight mini manipulator. The idea behind our approach for the coordination of macro and mini structures is to treat them as a single redundant system.

The *dynamic coordination* we propose is based on combining the operational space control with a minimization of deviation from the midrange joint positions of the mini-manipulator. This minimization is implemented with a joint torques selected from the *dynamically consistent* null space of equation (4) to eliminate any coupling effect on the end-effector.

This is

$$\Gamma_{\text{Null-Space}} = \left[I - J^T(\mathbf{q})\overline{J}^T(\mathbf{q})\right]\Gamma_{\text{Coordination}}; \tag{8}$$

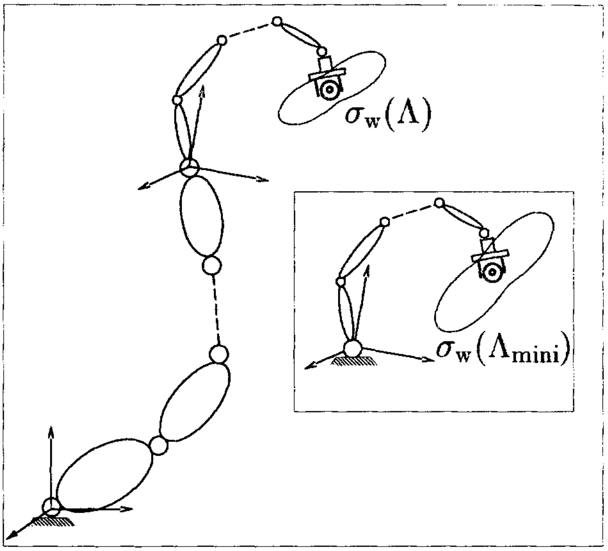

$\sigma_{\mathrm{w}}(\Lambda)$
$\sigma_{\mathrm{w}}(\Lambda_{\mathrm{mini}})$

Figure 1. Inertial properties of a macro/mini-manipulator

5. Cooperative Manipulation

Our research in cooperative manipulation has produced a number of results which provide the basis for the control strategies we are developing for mobile manipulation platforms. Our approach is based on the integration of two basic concepts: The *augmented object* [5] and the *virtual linkage* [6]. The *virtual linkage* characterizes internal forces, while the *augmented object* describes the system's closed-chain dynamics. These models have been successfully used in cooperative manipulation for various compliant motion tasks performed by two and three PUMA 560 manipulators [7].

5.1. Augmented Object

The *augmented object* model provides a description of the dynamics at the operational point for a multi-arm robot system. The simplicity of these equations is the result of an additive property that allows us to obtain the system equations of motion from the equations of motion of the individual mobile manipulators.

The *augmented object* model is

$$\Lambda_{\oplus}(\mathbf{x})\ddot{\mathbf{x}} + \mu_{\oplus}(\mathbf{x},\dot{\mathbf{x}}) + \mathbf{p}_{\oplus}(\mathbf{x}) = \mathbf{F}_{\oplus}; \tag{9}$$

with

$$\Lambda_{\oplus}(\mathbf{x}) = \Lambda_{\mathcal{L}}(\mathbf{x}) + \sum_{i=1}^{N} \Lambda_i(\mathbf{x}); \tag{10}$$

where $\Lambda_{\mathcal{L}}(\mathbf{x})$ and $\Lambda_i(\mathbf{x})$ are the kinetic energy matrices associated with the object and the i^{th} effector, respectively. The vectors, $\mu_{\oplus}(\mathbf{x},\dot{\mathbf{x}})$ and $\mathbf{p}_{\oplus}(\mathbf{x})$ also have the additive property.

The generalized operational forces $\mathbf{F}_\oplus$ are the resultant of the forces produced by each of the N effectors at the operational point.

$$\mathbf{F}_\oplus = \sum_{i=1}^{N} \mathbf{F}_i. \tag{11}$$

The dynamic decoupling and motion control of the augmented object in operational space is achieved by selecting a control structure similar to that of a single manipulator. The dynamic behavior of the augmented object of equation (9) is controlled by the net force $\mathbf{F}_\oplus$. Due to the actuator redundancy of multi-effector systems, there is an infinity of joint-torque vectors that correspond to this force.

5.2. Virtual Linkage

Object manipulation requires accurate control of internal forces. Recently, we have proposed the *virtual linkage* [7] as a model of internal forces associated with multi-grasp manipulation. In this model, grasp points are connected by a closed, non-intersecting set of virtual links, as illustrated in Figure 2 for a three-grasp task.

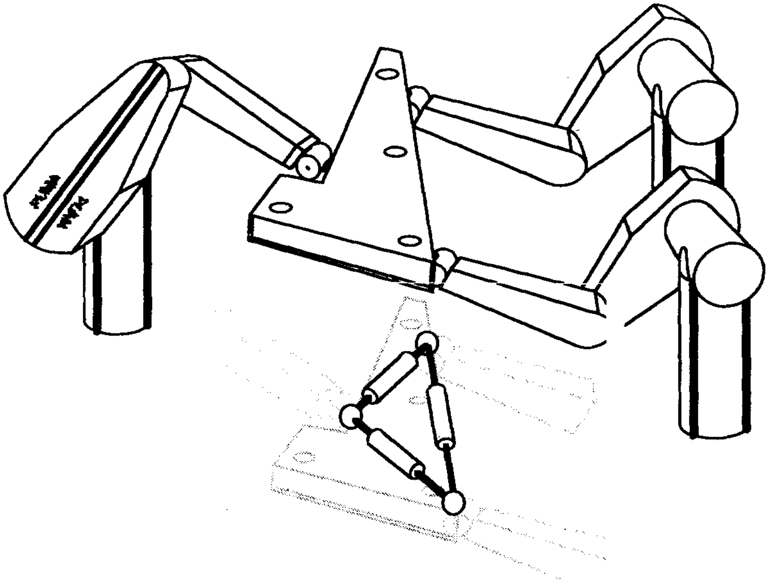


Figure 2. The Virtual Linkage

In the case of an N-grasp manipulation task, a *virtual linkage* model is a $6(N-1)$ degree of freedom mechanism that has $3(N-2)$ linearly actuated members and N spherically actuated joints. Forces and moments applied at the grasp points of this linkage will cause forces and torques at its joints. We can independently specify internal forces in the $3(N-2)$ members, along with

$3N$ internal moments at the spherical joints. Internal forces in the object are then characterized by these forces and torques in a physically meaningful way.

The relationship between applied forces, their resultant and internal forces is

$$\begin{bmatrix} \mathbf{F}_{res} \\ \mathbf{F}_{int} \end{bmatrix} = \mathbf{G} \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_N \end{bmatrix}; \tag{12}$$

where $\mathbf{F}_{res}$ represents the resultant forces at the operational point, $\mathbf{F}_{int}$ the internal forces and $\mathbf{f}_i$ the forces applied at the grasp point i. $\mathbf{G}$ is called the grasp description matrix, and relates forces applied at each grasp to the resultant and internal forces in the object.

5.3. Decentralized Cooperation

For fixed base manipulation, the *augmented object* and *virtual linkage* have been implemented in a multiprocessor system using a centralized control structure. This type of control is not suited for autonomous mobile manipulation platforms.

In a multiple mobile robot system, each robot has real-time access only to its own state information and can only infer information about the other robots' grasp forces through their combined action on the object. Recently, we have developed a new control structure for decentralized cooperative mobile manipulation [8]. In this structure, the object level specifications of the task are transformed into individual tasks for each of the cooperative robots. Local feedback control loops are then developed at each grasp point. The task transformation and the design of the local controllers are accomplished in consistency with the *augmented object* and *virtual linkage* models.

6. Experimental Mobile Platforms

In collaboration with Oak Ridge National Laboratories and Nomadic Technologies, we have completed the design and construction of two holonomic mobile platforms (see Figure 3). Each platform is equipped with a PUMA 560 arm, various sensors, a multi-processor computer system, a multi-axis controller, and sufficient battery power to allow for autonomous operation. The base consists of three "lateral" orthogonal universal-wheel assemblies which allow the base to translate and rotate holonomically in relatively flat office-like environments [9].

The control strategies discussed above have been implemented on these two platforms. Erasing a whiteboard, cooperating in carrying a basket, and sweeping a desk are examples of tasks demonstrated with the Stanford Mobile Platforms [10]. The dynamic coordination strategy has allowed full use of the relatively high bandwidth of the PUMA. Object motion and force control performance with the Stanford mobile platforms are comparable with the results obtained with fixed base PUMA manipulators.

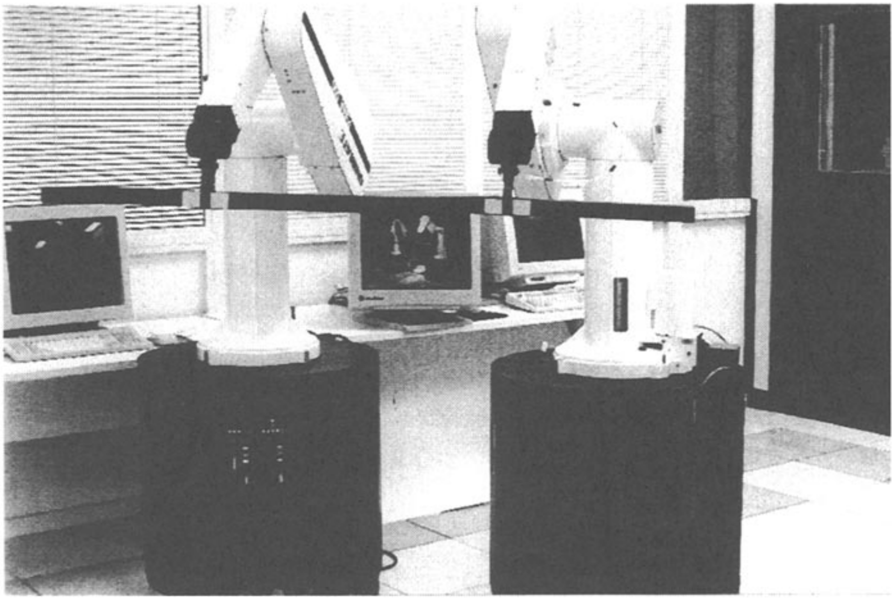


Figure 3. The Stanford Mobile Platforms

7. Conclusion

We have presented extensions of various operational space methodologies for fixed-base manipulators to mobile manipulation systems. A vehicle/arm platform is treated as a macro/mini structure. This redundant system is controlled using a dynamic coordination strategy, which allows the mini structure's high bandwidth to be fully utilized.

Cooperative operations between multiple platforms rely on the integration of the *augmented object*, which describes the system's closed-chain dynamics, and the *virtual linkage*, which characterizes internal forces. These models are the basis for the decentralized control structure presented in [8].

Vehicle/arm coordination and cooperative operations have been implemented on the two mobile manipulator platforms developed at Stanford University.

Acknowledgments

The financial support of Boeing, General Motors, Hitachi Construction Machinery, and NSF (grants IRI-9320017 and CAD-9320419) is gratefully acknowledged. Many thanks to Alain Bowling, Oliver Brock, Arancha Casal, Kyong-Sok Chang, Robert Holmberg, Francois Pin, Diego Ruspini, David Williams, James Slater, John Slater, Stef Sonck, and Kazuhito Yokoi for their valuable contributions to the design and construction of the Stanford Mobile Platforms.

8. References

1 Ullman, M., Cannon, R., Experiments in Global Navigation and Control of a Free-Flying Space Robot. Proc. Winter Annual Meeting, Vol. 15, 1989, pp. 37-43.

2 Umetani, Y., and Yoshida, K., Experimental Study on Two-Dimensional Free-Flying Robot Satellite Model. Proc. NASA Conf. Space Telerobotics, 1989.

3 Papadopoulos, E., Dubowsky, S., Coordinated Manipulator/Spacecraft Motion Control for Space Robotic Systems. Proc. IEEE Int. Conf. Robotics and Automation, 1991, pp. 1696-1701.

4 Khatib, O., A Unified Approach to Motion and Force Control of Robot Manipulators: The Operational Space Formulation. IEEE J. Robotics and Automation, vol. 3, no. 1, 1987, pp. 43-53.

5 Khatib, O., Inertial Properties in Robotics Manipulation: An Object-Level Framework, *Int. J. Robotics Research,* vol. 14, no. 1, February 1995. pp. 19-36.

6 Williams, D. and Khatib, O., The Virtual Linkage: A Model for Internal Forces in Multi-Grasp Manipulation. Proc. IEEE Int. Conf. Robotics and Automation, 1993, pp. 1025-1030.

7 Williams, D. and Khatib, O., Multi-Grasp Manipulation," IEEE Int. Conf. Robotics and Automation Video Proceedings 1995.

8 Khatib, O. Yokoi, K., Chang,K., Ruspini, D., Holmberg, R. Casal, A., Baader., A. "Force Strategies for Cooperative Tasks in Multiple Mobile Manipulation Systems," *Robotics Research 7, The Seventh International Symposium,* G. Giralt and G. Hirzinger, eds., Springer 1996, pp. 333-342.

9 Pin, F. G. and S. M. Killough, "A New Family of Omnidirectional and Holonomic Wheeled Platforms for Mobile Robots", IEEE Trans. on Robotics and Automation, Vol. 10, No. 4, August 1994, pp. 480-489.

10 Khatib, O., K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, A. Casal, and A. Baader. "The Robotic Assistant," *IEEE Int. Conf. Robotics and Automation Video Proceedings,* 1996.

Part Three

Applications

FORESTRY ROBOTICS
- why, what and when

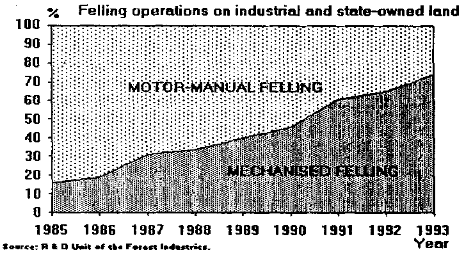
Aarne Halme
Automation Technology Laboratory*
Helsinki University of Technology
Espoo, Finland
aarne.halme@hut.fi

Mika Vainio
Automation Technology Laboratory
Helsinki University of Technology
Espoo, Finland
mika.vainio@hut.fi

Abstract: This paper overviews critically the state of the art of robotics in forestry applications and gives examples of potential development in the field. Benefits and restraints of forestry robotics are analysed and a scenario of future development is presented.

1. Introduction

Forestry is one of the oldest industries. Wood has been taken out from forests for houses, constructions, ship building and later for pulp and paper for several hundred of years. The technology has been, however, undeveloped relaying on manual technology and animal power until very recently. For example, in the Nordic countries Finland and Sweden, where the development has been probably the fastest in the world the intensive use of machines started only in the mid 80's. Since then, on the other hand, the development has been very fast. For example, the share of manually harvested industrial wood in Finland has decreased from about 80 % to about only 5 % during the last ten years as is shown in Fig. 1.



Source: R & D Unit of the Forest Industries.

Fig. 1: Mechanisation of felling in Finland (see http://www.metla.fi/forestfin/)

*See www.automation.hut.fi for detailed presentation of the laboratory

This means that the harvesting is done in practice by machines like those illustrated in Fig.2.

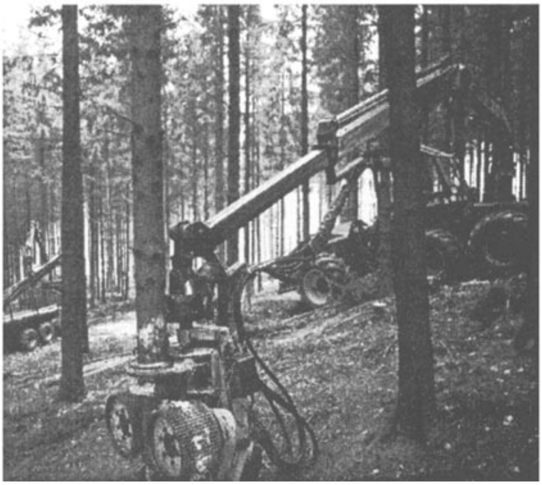


Fig.2: Difficult environment (e.g. steep slopes and tight spaces) creates real problems for machine and especially for robot applications

In other forestry intensive countries like Canada the trend is the same. The reason is partly economical, but also social facts play an important role. Due to the moving of people from the countryside to cities the amount of workers available for manual harvesting is decreasing rapidly and thus mechanisation and automation are in fact the only ways to replace this shortage.

The basis for the forestry robotics is the mentioned development. Although we are in half-way in this development at the moment it can be said positively that within a certain time frame we will see real robotic machines working in forests. In this paper some existing prototype machines are presented and furthermore some possible general guidelines for forestry robotics in future are given.

2. Development of forest machinery technology

The basic tasks in forestry are related in one hand to the flow of wood from forest to factory and on the other hand to the maintenance operations like thinning and silviculture by which the economic forest areas are taken care off. The basic tasks in harvesting are felling, delimbing, cutting and transportation to the roadside. Debarking which was done previously in forest is nowadays done in factories. Felling, delimbing and cutting are done by harvesting machines (see Fig. 3), which are operated by one person. One such machine has typically the capacity of 10-12 manual workers. The machines are partly automated and modern versions are digitally controlled by CAN-bus based mechatronics. Typically the machine measures automatically the volume of the wood processed by its processor head, saves the results and communicates with the remote production control unit. The control of machine operations is done by joy-sticks, but on a quite low level.

Operations like non-slipping motion control and coordinated control of the boom have only recently been developed. However, many plans aiming to gradually increase the level of automation exist, for example in boom operations. The technology basis is now ready for robotic operations in a large extent.

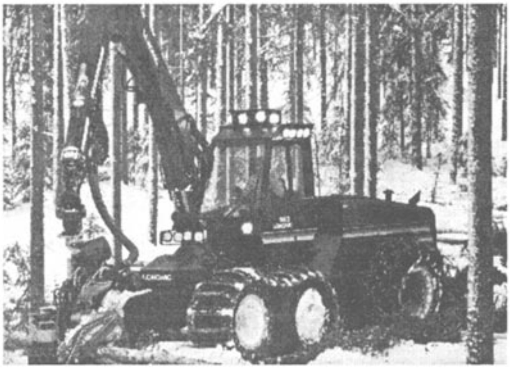


Fig. 3: A harvester in operation

Second typical machine is the forwarder, shown in Fig. 4a, which transports the wood from the felling side to the road side. It is a powerful tractor that takes a load of several tons. Modern machines are also mechatronic like harvesters. For example, an intelligent non-slipping motion controller is an important part of the control system. There have also been developments to achieve an autonomous AGV-type system for transportation, but this is still quite far in practice (Fig. 4b).

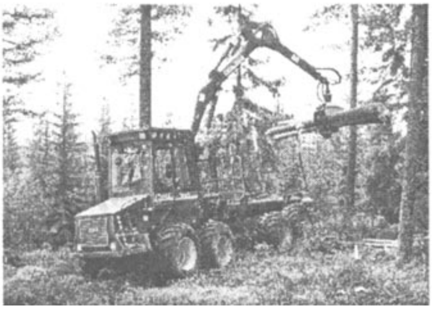


Fig. 4a: A normal forwarder

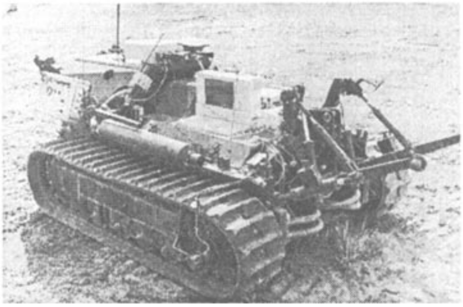


Fig. 4b: An autonomous prototype vehicle, which can be used e.g. for pulling trunks out of forest (Modulaire Ltd.)

The boom for loading and unloading wood is also a target for roboticing. The practical goal is to make these operations as short as possible and the operation of the boom as easy as possible. Recently digitally controlled booms have been developed for the markets. They can be controlled by the aid of joy-stick in Cartesian coordinates which makes the control easier. However, the great challenge is to automate the loading and unloading operations. Research on interactive robotics, which makes high level boom control possible has been done already more than ten years ago, see e.g. [1], but the results of practical product development are still missing.

In forestry maintenance operations are still done mainly manually. Need for automation is, however, as big as in harvesting. Planting is a task that has been tried to automate for a long time without any major success. The problem is that the ground quality varies in natural forest so much that finding proper places for plants is difficult even for a man and thus very demanding for a machine. The other area, where automation is needed is cleaning by which too tightly grown small trees are removed to obtain more room for growth. This is done still today mainly manually with motor saws because of the lack of proper machines.

3. Challenges for robotics

There are lots of challenges for robotics in forestry today. As stated before the general trend is toward more automated machines. To move man out of machines is, however, a big step and has not been taken this far except in some R&D projects (as to the authors' knowledge). There are two reasons for this. The first is the complicated environment. Forest is highly unstructured and uneven. Furthermore the ground quality varies considerably. Autonomous motion and above all autonomous working in such an environment need highly developed perception and adaptation capabilities from the robot. The second reason is the economics. Unmanned robotic vehicles also need maintenance and services by man. The simplest way to solve this problem is to let the vehicles have operators. However, a concept where a single operator can take care of several machines simultaneously is of interest because it can increase efficiency considerably. This means that the remote control technology will probably have the first priority in the process of removing the man out of the machine. It is also an interesting solution for safety problems that are encountered when working in mountain areas on steep slopes.

Before unmanned machines will be reality many steps must be taken to increase the usability of today's machines by adding robotic features to them. This means that some subtasks, like those of boom operations, machine locomotion (e.g., walking machine, see later on), thinning operations, planting, etc. can be automated and controlled by the operator. The use of only high level commands provides the operator time to concentrate on more demanding perception, assessment and planning tasks that are more difficult to automate. An example of such task division is a high level control of the harvester processor boom. At the moment the operator controls all motions of the boom. We could well think of a system in which the operator only chooses the next tree to be harvested by pointing it with a laser pointer. After that the rest of the task, i.e., controlling the boom to the tree, felling, delimbing and cutting, is done automatically. The laser pointer defines the location of the tree accurately enough in the machine based coordinate system. Nowadays it is relatively easy to make a robot control system that executes the mentioned subtasks, see e.g. [2].

Some of the tasks in forestry are of such nature that unmanned fully robotic type of technology is sound solution also from an economical point of view. In some countries forestry reminds of traditional farming. Extremely rapidly growing trees (e.g. eucalyptus trees in Portugal) are planted in regular lines with very little space between individual trees. After a short time (less than 10 years) all trees are cut down much like in field harvesting. This kind of operation would certainly be suitable for robots. Planting is an another suitable task as well as silviculture, where small trees are taken care of by cleaning their surrounding from other plants time to time. It is highly manpower intensive work and is coming economically impossible in many countries because chemical treatment is not allowed any more. Mechanical treatment can be done by robots that are working in colonies like herd of lambs. An example of such development is given in the next section. In generally there are at least two basic reasons why this kind of multi-robot concept is of potential interest as an engineering solution. The first reason is fault tolerance. A multi-robot system has a high redundancy, because the functions of a faulty individual can be always easily replaced by the other robots. The second reason is a robot to robot communication structure which makes it possible to increase or decrease the number of robots in a colony easily without any reconfiguration of the communication structure. Applications where a long term fully autonomous operation is needed and/or the work to be done can be executed in a parallel way by a group of individual robots are natural tasks for multi-robot systems. For more about the multi-robot systems see e.g. [3] and [4].

Transportation perception and navigation technology for unmanned AGV-type of vehicles are quite ready at the moment and it will be interesting to see if and when it will be realised in forestry. In some other areas, for instance in mining and construction, autonomous transportation with heavy trucks is very close to be taken into commercial use. Experiences this far indicate unambiguously that the most accurate and reliable vehicle navigation system can be made by fusing a dead reckoning gyro based navigation with a proper beacon based localisation system. Practical beacon systems are either optical or based on radio frequencies (for an excellent overview see [5]. By using a beacon system - even quite an inaccurate one - the drift error can be removed from the dead reckoning navigation system and its relative accuracy can be utilised in full power. The most potential beacon system for outdoor applications today is GPS (Global Position System) which is available almost all over the world. In vehicle navigation it is mostly used in the differential mode (DGPS) which gives around 5 m absolute localisation accuracy. With a moderate dead reckoning navigation system and the fusion method described shortly below the accuracy can be increased to the level of 0.5 m which is enough for most forestry or other work site applications.

A very useful tool for fusion of two or more different measurement sources is Kalman filtering. The idea is simple. We suppose here that the reader is familiar with the basic theory of Kalman filter or more precisely with its extended form. If not, one should become familiar with suitable text books, e.g. [6] and [7]. In the first phase of forming the algorithms needed a suitable model which describes the kinematic motion of the vehicle is derived. For wheel vehicles a simple "bicycle model" is usually sufficient, because these machines are moving slowly. The model is presented as follows,

$$X(k+1)=\begin{pmatrix}\mathrm{x(k+1)}\\ \mathrm{y(k+1)}\\ \theta\,\mathrm{(k+1)}\end{pmatrix}=\begin{pmatrix}\mathrm{x(k)}\\ \mathrm{y(k)}\\ \theta\,\mathrm{(k)}\end{pmatrix}+\begin{pmatrix}u_1T\cos(\theta\,\mathrm{(k)})\\ u_1T\sin(\theta\,\mathrm{(k)})\\ u_2T\end{pmatrix}, \tag{1}$$

where (x,y) is position of the vehicle body coordinate system origo (often set in the middle of the back axis) in some fixed world coordinate system, θ the heading angle of motion, u1 is the longitudinal velocity of the vehicle and u2 the angular velocity of the motion defined by the turning radius of the vehicle. The vector (x,y,θ) is estimated by using extended version of Kalman filter when the following measurement equation is used

$$z(k)=\begin{bmatrix}x_{DGPS}(k)\\ y_{DGPS}(k)\\ x_{DEAD.RECKONING}(k)\\ y_{DEAD.RECKONING}(k)\\ \theta(k)\end{bmatrix}+v(k)=H*\begin{bmatrix}x(k)\\ y(k)\\ \theta(k)\end{bmatrix}+v(k), \tag{2}$$

where v(k) is the measurement error vector (zero mean, white noise) and matrix H has the following form

$$H=\left[[10100]^T;[01010]^T;[00001]^T\right]. \tag{3}$$

The measurement data vector z contains position measurements both from DGPS and the dead reckoning system, and the heading measurement from the heading gyro. A practical problem related to the data reading is the different sampling intervals available. The dead reckoning instruments can be read with a high frequency typically from 1 up to 10 kHz. The DGPS instead can be read typically only with 1 to 2 Hz frequency. This problem can be solved by using a “little trick” and modifying the Kalman filter equations. The basic sample interval is chosen according to the high sampling frequency. The DGPS measurements are supposed to be valid within a time window around the reading time, the width being typically 5 to 10 sample intervals to both direction. The part of the covariance matrix related to the DGPS measurement is finite only in this time window and infinite elsewhere. The Kalman filter can be written in the form

$$\hat{X}(k|k)=\hat{X}(k|k-1)+K(z(k)-H*\hat{X}(k|k-1)), \tag{4}$$

$$\hat{X}(k|\kappa-1)=\begin{bmatrix}\hat{x}(k-1|k-1)\\ \hat{y}(k-1|k-1)\\ \hat{\theta}(k-1|k-1)\end{bmatrix}+\begin{bmatrix}u_1T\cos(\theta\,(k-1|k-1))\\ u_1T\sin(\theta\,(k-1|k-1))\\ u_2T\end{bmatrix}, \tag{5}$$

$$\underset{3\times5}{K}=[\mathrm{K}_1|\,\mathrm{K}_2], \tag{6}$$

where K_1 represents the part of the gain which comes from the dead reckoning part and K_2 the part coming from the DGPS part. K_2 is non-zero only in the time window mentioned above and thus contributes to the correction part of the filter.

For actual results of the method presented here see e.g. [8], where the positioning of an autonomous off-road vehicle has been done by fusing DGPS and inertial navigation. At the moment the price of the navigation and perception system needed for autonomous operation is nevertheless still too high for forestry machines that are not operated by industrial enterprises but small companies or individual farmers.

4. What exists?

Only few examples of forestry robots exist today. Most of them are still development projects, which means that the robots are not in everyday work but are used more or less for testing purposes. However, those described in the following are serious attempts to develop products that most probably will be seen in a form or another in real use in the future.

4.1 Walking harvester

Along with the tightening environment protection requirements forest harvesting has to be executed more carefully, not harming the ecosystem more than necessary. The typical harvesting machines, shown in Fig. 3, are relatively heavy and clumsy. They may easily harm the ground infrastructure and roots of the standing trees if the drivers are not very careful. This is mainly because of the locomotion system that uses tracks and large wheels. Legged locomotion system is an interesting alternative that has been already proven to be much more gentle to the ground infrastructure. A legged machine with extra degree of freedom is also more flexible in omnidirectional motion. Fig. 6 shows an experimental six legged machine, called MECANT, developed at the end of 80's at Helsinki University of Technology, see e.g. [9], to study walking locomotion in natural environment. MECANT is hydraulically driven stand alone 1 ton machine that is remotely controlled. The legged locomotion is fully automated by onboard computers and the operator can use all six d.o.f. of the body within the kinematic limits. This means extra flexibility when driving the machine on uneven terrain and in stabilising it in working position. Although the machine is not autonomous, but driven remotly by an operator using radio joy-stick, its locomotion system is robotic-like and needs a large software for its functions. The piloting system of MECANT is divided into different planners, as shown in Fig. 7 below. The main planners are the body motion planner, the gait planner and the foothold planner [10]. In this figure the "transfer feet" mean feet which are in motion in the air to find the next foothold, and the support feet mean feet presently supporting the body on the ground.

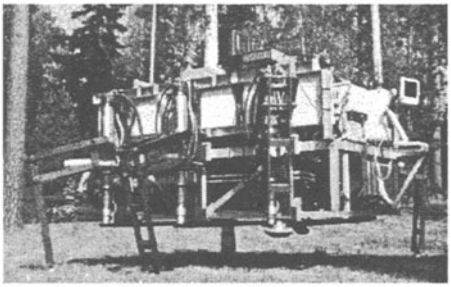


Fig. 6: MECANT

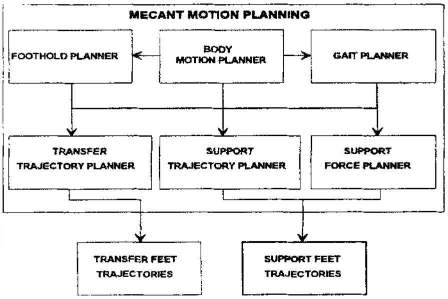

MECANT MOTION PLANNING
FOOTHOLD PLANNER
BODY MOTION PLANNER
GAIT PLANNER
TRANSFER TRAJECTORY PLANNER
SUPPORT TRAJECTORY PLANNER
SUPPORT FORCE PLANNER
TRANSFER FEET TRAJECTORIES
SUPPORT FEET TRAJECTORIES

Fig. 7: The motion planning of MECANT

The body motion planner determines the body motion trajectory according to the operator's velocity commands, the vehicle's sensor data and the local terrain model. *The gait planner* determines the "leg states" (support or transfer), and the phases and state periods (time lengths of the support or transfer states) according to the gait chosen. *The foothold planner* determines the optimal foothold position for a recovering (transfer) foot. *The support trajectory planner* calculates the velocities and positions of the supporting feet, according to the velocity vector of the vehicle's body. *The transfer trajectory planner* calculates the velocities and positions of the transfer feet. The functions of the planner can be further divided into different phases, such as lifting, moving and placing. *The support force planner* determines the supporting forces of the feet. If the legs are purely force controlled, the determination of the feet forces vectors can also be done in this planner. The body motion planner, the support trajectory planner and the support force planner are the same for every gait. The foothold planner, the transfer trajectory planner and, naturally, the gait planner depend on the gait chosen. In [11] a gait planner that is based on heuristic rules was presented. It took into account in every planning cycle the temporal direction and velocity of motion, the availability of the legs to start a new working cycle, the stability of the machine in the direction of motion, and the possible collisions between neighbouring legs. This considered *free gait* is especially useful in conditions where the machine is executing omnidirectional motion in rough terrain. Compared to wave gait or adaptive wave gait, a more flexible motion and an improved stability could be obtained. Currently research is continuing by implementing a *semidynamic walking* into MECANT. The motivation for this is to speed up the locomotion speed of the robot. The method is based on continuos optimization of the stability. It is done by adding an extra velocity vector component to the commanded velocity vector. This additional component drives the centre of gravity of the vehicle towards the most stable point of the leg support pattern. For more information see [12]. Another problem in the design of a walking machine is the velocity of the machine. One has to constantly trade between the stability and the speed. Fig. 8 illustrates the relationship between the leg transfer time and locomotion speed with different so called duty factors in periodic 6-legged wave gaits. Duty factor β is calculated as follows

$$\beta = \tau_S / \tau \,, \tag{7}$$

where τ_S is the leg support time and τ is the leg cycle time. Basically the duty factor value of n/6 indicates that n legs are supporting a six-legged vehicle during the locomotion cycle. Fig. 8 shows that if a maximum stability is needed (i.e. β = 5/6) and the locomotion speed should be 2 m/s, a leg transfer velocity of more than 15 m/s is needed. Needless to say that this kind of leg movement is very hard if not impossible to obtain. On the other hand if we are happy with minimum stability (i.e. β = 3/6) the locomotion speed can be reached with only 3m/s leg transfer velocity. For more extensive study on this matter see e.g. [13] and [14].

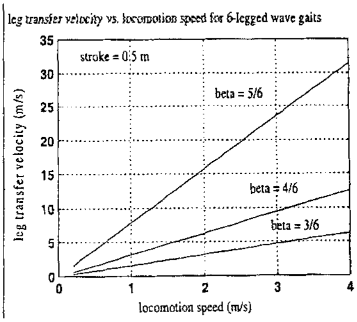


Fig.8: The average velocity of transfer foot as a function of locomotion speed in periodic 6-legged wave gaits [14].

Regardless of the presented limitations, the benefits of walking locomotion are so numerous, that the publishing of a prototype forest harvester in 1995 (by Plustech Ltd, Finland), shown in Fig. 9., was not at all unexpected.

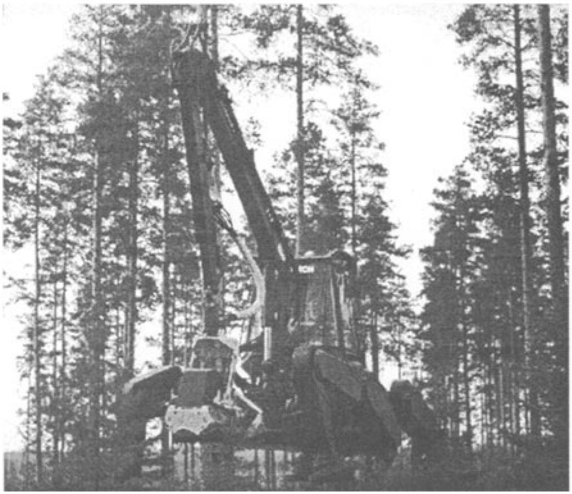


Fig. 7: Prototype walking harvester (Timberjack/Plustech Ltd)

The machine is based on walking locomotion system and standard harvesting tool (see http://www.plustech.fi). It has been used in test work during a full year in Nordic climate and experiences this far are positive. Especially the environment friendly properties have turned out to be better than expected. Within next few years it is probable that this technology becomes commercial in certain areas of the world where forest harvesting is under strong environmental regulations, like in Scandinavia and central Europe.

4.2 Slope climbing machine for silviculture applications in mountain areas

Another type of semi-walking machine for forestry use has been developed in Japan in co-operation with Tokyo University Forestry Department and a forestry company. This machine is intended for steep slope operations like thinning, brushing and planting. In Japan the forests are growing mainly on hillside and forestry operations are sometimes extremely difficult. The prototype machine is shown in Fig. 8.

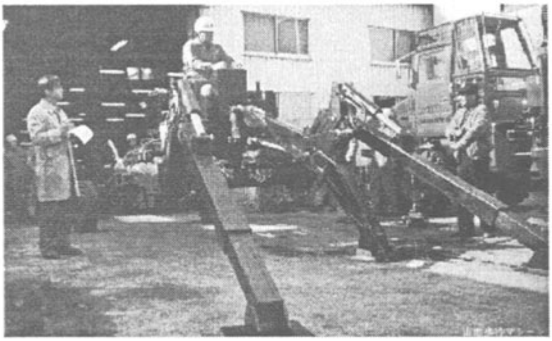
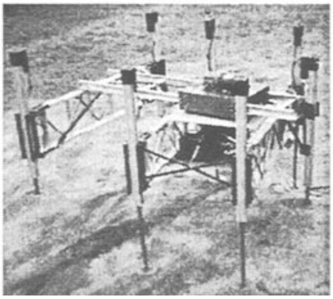


Fig.8: Slope climbing machine

It has three front legs that can perform gaiting and two rear wheels that are not powered but have brakes. The centre leg can be used also as a manipulator. When writing this the authors did not know whether or not the prototype has been already made commercial.

4.3 Multi-robot system for brushing and thinning of young trees

An interesting multi-robot concept for brushing and thinning of young trees has been developed in Canada by Petawawa National Forestry Institute, see e.g. [15]. This kind of operation is important especially during the first years after planting to provide the best possible growth condition for young trees. Traditionally the work has been done manually by using chemicals or mechanical tools. Today only mechanical treatment is possible and there is typically a lack of manpower to do the job. The idea in the project is to develop a multi-robot "heard" of slowly walking light weight machines that can find the plant and brush its environment. The most difficult problem is perceptive, i.e., to find the young tree among the grass and other vegetation. This is now under solving by using colour camera and image processing. The vehicle itself, shown in Fig. 9, is still a research prototype, but the development has continued in a company called Autonomous Walking Machines Inc.



See *www.pfc.forestry.ca/www_users/fgougeon/rem_sens/FG_RV.html* for detailed description

Fig. 9: The prototype robot, Jacob, for brushing and thinning

The basic idea in this project is very good and it is quite sure that it can be commercialised provided a good solution for the difficult perception problem can be found.

5. Discussion and conclusions

Many scenarios about the future development of forestry robotic technology can be made. One is illustrated in Fig. 10. We will see real applications of robotic technology, remote handled machines, multi-machine mobile operation stations and even applications with high level autonomy. There are three basic factors that affects this development: *environment consciousness of people, availability of manpower and cost of technology*. All these factors indicate at the moment that there will be a radical change in technology during the next 10-15 years.

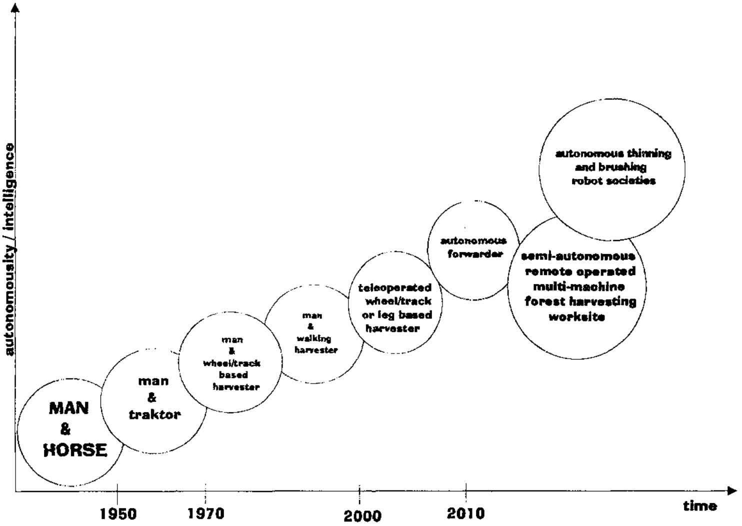

autonomousity / intelligence
MAN & HORSE
man & traktor
man & wheel/track based harvester
man & walking harvester
teleoperated wheel/track or leg based harvester
autonomous forwarder
semi-autonomous remote operated multi-machine forest harvesting worksite
autonomous thinning and brushing robot societies
1950
1970
2000
2010
time

Fig. 10: A vision for future development in forestry

For most of the new applications the technology basis already exists, see an overview in [16]. It is more question of when and in what form the applications will mature for commercial markets. One fact that sets a high threshold for acceptance of the new technology in this field is the mode of action in forestry business. In most cases the operations are taken care off by small contractor companies or even individual farmers whose possibilities to invest in technology are limited. Reliability and easy maintainability of the technology are also of crucial importance. The situation is much different as for instance in mining. This makes the machine makers in the field also conservative. However, the factors mentioned above are also strong and, as we have seen during the last ten years, they have influenced strongly - in fact more strongly than what was predicted in the mid 80's.

References

[1] Halme A, Heikkilä T, Torvikoski T 1987 An Interactive Robot Control System. *International Journal of Robotics and Automation*, Vol. 2. No 3.

[2] Manninen M, Halme A, Myllylä R 1984 An Aimable Laser Time-of-Flight Range Finder for Rapid Interactive Scene Description. In: *Proceedings of the 7th Annual Conference of British Robot Association*, Cambridge

[3] Halme A, Jakubik P, Schönberg T, Vainio M 1993 The Concept of Robot Society and Its Utilization. In: *Proceedings of the IEEE/Tsukuba International Workshop on Advanced Robotics*, pp 29 - 35

[4] *Distributed Autonomous Robotic Systems 2* 1996. Asama H, Fukuda T, Arai T, Endo I (eds), Springer-Verlag, Tokyo

[5] Everett H R 1995 *Sensors for mobile robots: theory and applications*, A K Peters, Wellesley, MA

[6] Maybeck P S 1979 *Stochastic models, estimation and control, Vol.1*, Academic Press, New York

[7] Maybeck P S 1982 *Stochastic models, estimation and contro*l, *Vol.2*, Academic Press, New York

[8] Schönberg T, Ojala M, Suomela J, Torpo A, Halme A 1996 Positioning an autonomous off-road vehicle by using fused DGPS and inertial navigation. *International Journal of Systems Science*, Vol. 27, 8: 745-752

[9] Hartikainen K, Halme A, Lehtinen H, Koskinen K 1992 Control and Software Structures of a Hydraulic Six-Legged Machine Design for Locomotion in Natural Environment. In: *Proceedings of IROS'92*, pp 590-596

[10]Halme A, Hartikainen K, Kärkkäinen K 1994 Terrain Adaptive Motion and free Gait of a six-legged Walking Machine. *Control Engineering Practice*. Vol. 2, 2: 273-279

[11]Salmi S, Halme A 1996 Implementing and testing a reasoning-based free gait algorithm in the six-legged walking machine "MECANT". *Control Engineering Practice*. Vol. 4, 4: 487-492

[12]Halme A, Salmi S, Leppänen I 1997 Control and stabilisation of the semi-dynamical motion of a heavy six-legged walking machine. In: *Workshop of Walking Machines, (ICAR'97)*, July 6 1997, Monterey

[13]Halme A, Hartikainen K 1996 Designing the Control System of an Advanced Six-Legged Machine. In: Gray J O, Caldwell D G(eds) 1996 *Advanced Robotics & Intelligent Machines*, The Institution of Electrical Engineers, London, pp 177-190

[14]Hartikainen K 1996 Motion planning of a walking platform designed to locomote on natural terrain. *Ph.D. Thesis*, Helsinki University of Technology.

[15]Gougeon F A, Kourtz P, Strome M 1994 Preliminary research on robotic vision in a regenerating forest environment. In: Borkowski A, Crowley J L(eds) *Int. Symp. Intelligent Robotic Systems '94*, pp 255-262

[16]Halme A 1995 Mobile Robotics in Unstructured Environments - Some Advanced Applications, In: *Proceedings of the 1995 National Conference of the Australian Robot Association*, Australian Robot Association (invited paper), Sydney

Robotics for the Mining Industry

Peter I. Corke and Jonathan M. Roberts and Graeme J. Winstanley
CSIRO Division of Manufacturing Science and Technology
CRC for Mining Technology and Equipment
P O Box 883, Kenmore, Queensland 4069, Australia
http://www.cat.csiro.au/automation

Abstract: The mining industry is highly suitable for the application of robotics and automation technology since the work is both arduous and dangerous. However, while the industry makes extensive use of mechanisation it has shown a slow uptake of automation. A major cause of this is the complexity of the task, and the limitations of existing automation technology which is predicated on a structured and time invariant working environment. Here we discuss the topic of mining automation from a robotics and computer vision perspective — as a problem in sensor based robot control, an issue which the robotics community has been studying for nearly two decades. We then describe two of our current mining automation projects to demonstrate what is possible for both open-pit and underground mining operations.

1. Introduction

Automation in mining is desirable because it offers the advantages of:

- higher productivity;
- increased safety, by reducing human exposure to hazards;
- reduced operating stress on equipment.

Most of the productivity increase in mining this century has been due to mechanisation. Mines have moved from using predominantly human and animal power to large electric and diesel powered machines. The trend in the past thirty years has been towards larger and more powerful machines but practical limits have now been reached. Underground machines are further constrained by the dimensions of the tunnels in which they operate. As growth in machine size tails off, other approaches must be pursued in order to achieve productivity growth. Automation is a strong contender due to the phenomenal growth in the capability of sensing, control and computing technologies over the last decade and the reduction in cost. The focus of the two projects described in this chapter is on increasing productivity of existing capital assets by retrofitting with enhanced sensing and control technology.

Mining is a dangerous occupation and despite almost continuous improvement fatalities still occur. In addition there are a great many other serious injuries and subtle long-term health risks due to noise, dust and fume inhalation, and vibration. Safety can be best improved by removing people from the dangerous parts of the mine and other health damaging occupations.

Maintenance is also a significant cost in mining, estimated to be as high as 30% of the total costs. Automated systems provide a non-obvious benefit through controlled demands on the machine, operating it within its design envelope. This will lead to reduced wear and tear and thus significant cost savings.

However despite all these apparent advantages mining automation has progressed far more slowly than factory automation. The principal reason is that automation is feasible only when the machine "knows" its environment. In industrial applications, considerable engineering effort is expended in providing a suitable work environment for machines. This entails the design and manufacture of specialised part feeders, jigs to hold the work in progress, and special purpose end-effectors. High non-recurrent engineering costs are incurred, but automation can proceed since the environment is known and can be described.

A great many tasks routinely performed by humans in unstructured environments (for example machine control and driving) are based on visually perceived information. In order for robots to perform such tasks, without extensive instrumentation or re-engineering of the environment, they must also have the ability to perceive and act upon visual information. Computer vision is therefore an important sensor for robotic systems since it mimics the human sense of vision and allows for non-contact measurement of the environment. Visual control, or visual servo, systems[1, 2] are those in which a machine vision sensor is used to provide position feedback for a machine. Much of the research in laboratories is based on small electric drive robots but the techniques are applicable to machines of the scale used in mining. This is an area of great current research interest and many robotic systems have been demonstrated in indoor laboratory environments. Some, such as the HelpMate robot, are in commercial operation. The mining environment is arguable much harsher, is continuously changing (it could be argued that this is the purpose of mining), and the machines involved are extremely large and powerful and hence dangerous if not controlled correctly.

The next two sections discuss in some detail two of the applications that currently being investigated at the Centre for Mining Technology and Equipment.

2. Dragline automation

Draglines are used to remove overburden[1] and uncover coal. As remaining coal seams become deeper, more overburden must be removed to uncover the same amount of coal, and this has become a bottleneck in the production process. At a cost of $50M to $100M, buying another dragline is a major investment for any coal mine, so improvement in productivity through automation is of considerable interest. Quite modest improvements in productivity, for example

[1] The economically valueless material which covers a coal seam.

4%, would produce a benefit of $3M/year per dragline.

A dragline, see Figure 1, comprises a rotating assembly that includes the 'house' (drive motors, controls, and operator cabin), tri-structure or mast, and boom. The house rotates on a bearing surface on top of the 'tub' which sits on the ground. A large diameter ring gear is fixed to the tub and the house is rotated by a number of pinions driven by motors in the house. A walking dragline is able to drag the tub along the ground by means of large eccentrically driven 'walking shoes' at the side of the machine.

The dragline has three driven mechanical degrees of freedom:

- the house and boom can *slew* with respect to the tub;
- the bucket can be *hoist*ed by a cable passing over sheaves at the tip of the boom;
- the bucket can be *drag*ged toward the house by a cable passing over sheaves at the base of the boom.

During digging the bucket motion is controlled using only the drag and hoist ropes. When the bucket is filled it is hoisted clear of the ground and swung to the dump position by rotating the house and boom. The drag and hoist drives now control the position of the bucket within a vertical plane that contains the centerline of the boom however the bucket is free to swing normal to that plane.

As can be seen from the Figure 1 draglines are very large machines. Typical draglines have boom lengths of over 100 m, weigh up to 5000 tonnes, and can employ up to sixteen 500 hp motors on the slew drive and bucket capacity can range from 100 to 250 tonnes.

A good deal of operator skill is required to control, and exploit, the bucket's natural tendency to swing. An early attempt to automate bucket swing by replaying a taught sequence of operator setpoints was unsuccessful, and we believe this was due to the lack of knowledge about the instantaneous state of the swinging load.

A complete cycle takes around 1 minute to complete, of which 80% is swinging the bucket through free space — it is this portion of the cycle that we propose to automate. Our work is based on the observation that control of the bucket moving through free-space is really a robotics problem. The electric-drive dragline can be considered to be a 3DOF robot with a flexible final link. The proposed swing automation system will be activated by the operator once the bucket is filled. It will automatically hoist, swing and dump the bucket (at an operator set reference point) before returning the bucket to a predefined digging point. Our prototype is a 3500 tonne Bucyrus-Erie 1370 dragline which is in production 24 hours/day, and this greatly constrains the physical access we have to the machine for fitting automation components and testing. Earlier experimental work on a one-tenth scale dragline [3] was used to demonstrate the automation concept to the industry. Modelling of the mechanical and electrical components of a production dragline and their control are discussed further in [4].

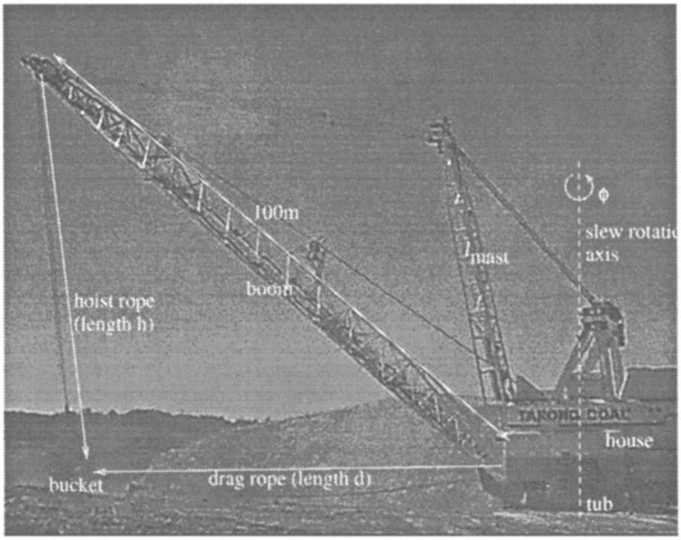


Figure 1. Annotated picture of the Bucyrus-Erie model 1370 dragline.

2.1. Bucket position sensing

As stated above a good deal of operator skill is required to control the bucket swing. The motion of a dragline bucket with respect to the boom can be considered as a pendulum with the ropes and aerodynamic forces providing some damping. Control of such a system requires a knowledge of the bucket position. Existing encoders on the winch drums provide hoist and drag rope lengths which define a locus along which the bucket moves, but the swing angle with respect to the boom is not known.

The bucket's harsh interaction with the environment precludes the use of instrumentation on the bucket itself. Various remote sensing techniques were investigated, but the specifications are stringent. The sensor must be capable of operating completely reliably 24 hours a day in all weather conditions, at a rate of at least 3 Hz (control constraint), and irrespective of changes in bucket type, contents, distance and orientation. An extensive experimental program [5] established that it was not practical to identify by the bucket against the mine background by conventional vision techniques. The deficiencies of the machine vision approach are best illustrated by the raw and edge bucket images, from a boom-tip mounted camera on the BE1370, shown in Figure 2. Real-world effects such as uncontrolled lighting, shadows, background texture and motion, and lack of contrast combine to thwart all approaches that we evaluated.

The approach that we selected instead is based on a scanning infra-red laser range finder (SICK Optic PLS), mounted near the boom tip to find the location of the hoist ropes. Figure 3 shows the PDF of target bearing and range occurrence for over 1000 trials in normal daylight conditions and in heavy rain. The heavy rain introduces small spikes but the hoist ropes are still clearly discernible and can be reliably located with a tracking filter. Preliminary results,

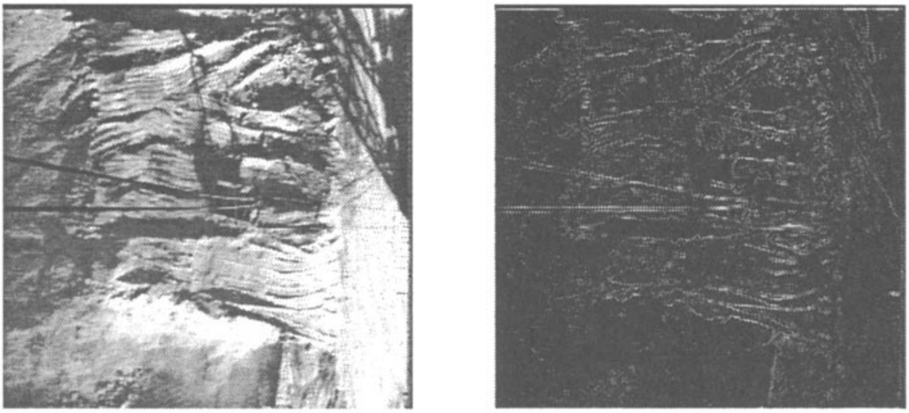


Figure 2. BE1370 bucket images (a) raw bucket image. (b) edge image.

Figure 4, show the range and bearing of the midpoint of the two hoist ropes during machine operation.

The sensing unit is mounted on a special platform just below the boom point sheaves (see Figure 5). The unit's associated electronics are mounted in a weatherproof cabinet on the boom. Raw data is sent back to the control computer via an optically isolated high speed RS422 serial link. The bucket sensing system combines the angle information from the rangefinder with the length of the hoist and drag ropes to determine the position of the dragline bucket in 3D space.

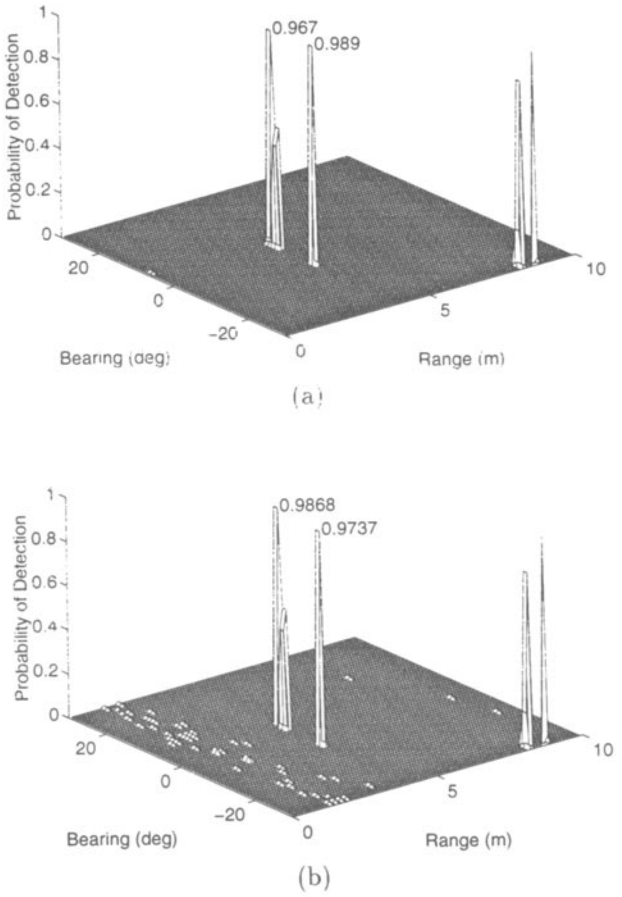
2.2. Dragline computer system

The control computer is a VMEbus PowerPC with dragline control and status signals connected to the control computer via Industry Pack (IP) interface modules. The quantities monitored include motor armature voltage, motor armature current, regular input voltage from the operator's control, and winch drum and house rotation.

The boom mounted PLSs connect to an industrial PC that processes the raw PLS data and sends hoist rope angle details to the control computer processor via a local 10BaseT network. Both computers run the real-time multitasking operating system LynxOS.

The computers are powered by a mains filtered UPS. All signals between the dragline house and the boom and mast units are connected via arrestors to protect against lightning strikes. Variations in ground potentials between different parts of the dragline are overcome by optically coupling signals into and out of the control computer interface.

A radio LAN, mounted on the dragline mast, allows remote code development and monitoring of dragline operation. The latter is necessary to establish parameters for our dynamic models of the dragline [4]. The other end of the radio LAN, up to 1 km, is a caravan that serves as a mobile laboratory for code development and data analysis without requiring a physical presence on the dragline.



(a)

(b)

Figure 3. PDF of target bearing and range occurrence over 1000 trials (a) hoist ropes in daylight. (b) hoist ropes in heavy rain. The two large spikes in the centre of the figures represent the dragline hoist ropes; the large spikes on the right are artifacts of the test range.

2.3. Operator interface

An operator controls the dragline by means of a joystick for each hand (drag and hoist rope rate control) and a set of pedals to rotate the house left and right. Our automation system 'drives' the dragline by physically moving the control joysticks and pedals; somewhat like the cruise control in a car or the auto-pilot of a fly-by-wire aircraft. To achieve this we have fitted servo motors to each of the control devices.

Servoing these controls facilitates the smooth transfer of system set points between the automatic system and the operator. The servoed operator controls also restrict the control computer to the same safety interlocks and limits as

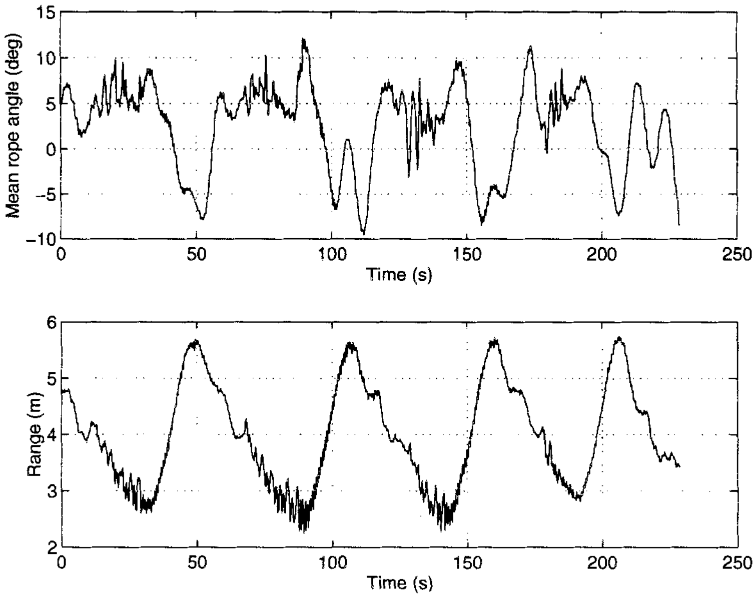


Figure 4. Hoist rope motion during dragline operation.

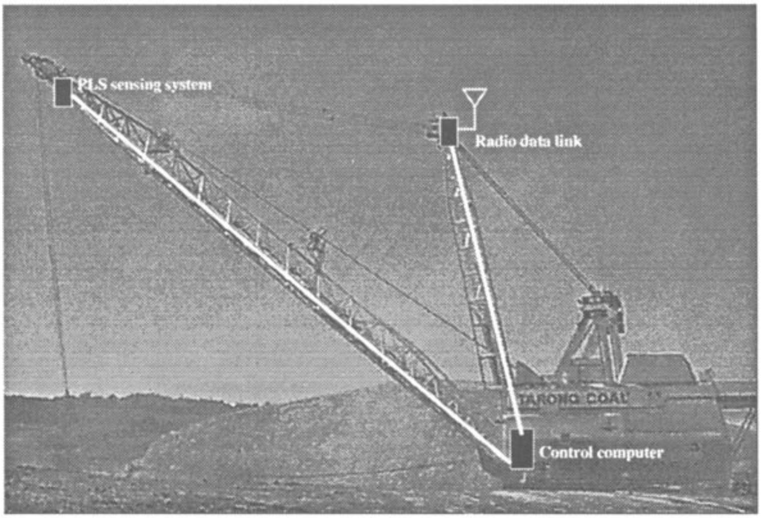


Figure 5. Location of major automation system components.

applies to input from an operator. Error sensing on the hoist and drag joystick servo motors allows the automatic system to sense if the operator is opposing the motion of the joysticks, in which case it smoothly transfers control of the dragline back to the operator.

2.4. System safety

The control system is designed to be fail safe; that is, it reverts to operator control should a system failure be detected. Watchdog timers ensure the consistent operation of the control computer and servoed operator controls. The system software incorporates both parameter range and change checks. A control pendant is used for testing the automation system. It ensures that a conscious physical action must be taken before the dragline can be placed in automatic mode, and a 'kill' button can immediately disable the automatic mode.

2.5. Summary

The automatic system is being deployed in two stages. Stage 1 is a logging phase in which the automatic system will be installed without any servoed operator controls. In the logging configuration the automatic computer system will be debugged and dragline characteristics required for the programming of the automation system will be measured. Stage 1 is scheduled to be installed in the second quarter of 1997.

Stage 2 of the deployment process involves the installation of the operator accepted servoed controls and the testing of the system in an automatic configuration. The servoed operator controls are scheduled to be installed in the third quarter of 1997 with testing of the automatic system commencing in the fourth quarter of 1997.

We see the current automation project as the first in a series that will eventually lead to fully automatic dragline operation. With the automation system operating we can use it as a platform to investigate issues such as automated bucket filling, kinaesthetic feedback of motor torque to the operator via the servoed control, dumping directly into a hopper or truck, and online planning based on local terrain sensing. These are all stages along the path to a fully autonomous operation.

3. Underground mining robotics

Many tasks in underground hard-rock (not coal) mining are performed by manually controlled hydraulic arms (known as 'booms') carrying tools such as drills, rock-breaking hammers, explosive charging hoses, shotcrete applicators, or cassettes of ground support bolts. A typical rock-breaking boom and tool is shown in Figure 6 and the similarity to a robot is clear, but all these underground machines are 'driven' by operators. A common characteristic of these booms is that are quite compliant, have considerable mechanical 'slop' in the joints, and the payload to self-mass ratio is much higher than for a factory robot.

The tasks performed by the operator typically involve positioning the tool with respect to some feature in the local environment, for example, positioning a rock breaker over a large rock to be broken, or a support bolt or an explosive charging hose into a pre-drilled hole. The operator is generally at a considerable disadvantage since the boom's compliance causes the end-point to bounce, the controls actuate joints directly rather than commanding motion in task space coordinates, and the operator sits a considerable distance away from the tool (up to 5 m) and the lighting is often poor.

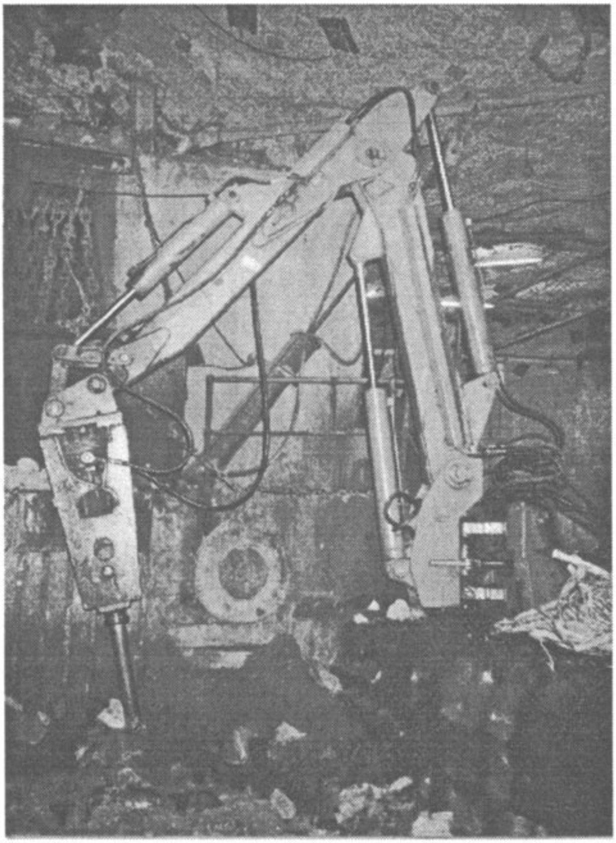


Figure 6. A rock breaker.

Most underground mining machines utilise hydraulic actuation with electrically driven valves, thus knowledge of electro-hydraulic systems is critical for automation purposes. Hydraulically actuated robots are not commonly used by the robotics research community, and those that do exist often make use of servo or precision components. In order to learn more about this technology a laboratory rig, shown in Figure 7, has been built to test our control concepts. The prototype was designed to simulate the mechanical and dynamic characteristics of typical underground hydraulic machinery, and with regard for issues such as robustness in underground use. We use standard proportional valves rather than the less rugged servo valves commonly used in much research work, and the hydraulic actuators have rugged inbuilt ultrasonic length transducers. Our approach to modelling the characteristics of the laboratory manipulator is described in [6].

The remainder of this section presents some preliminary results in 3D sensing and discusses the conceptual design of a semi-automated rock breaking system that demonstrates the need for sensing, control and human supervision.

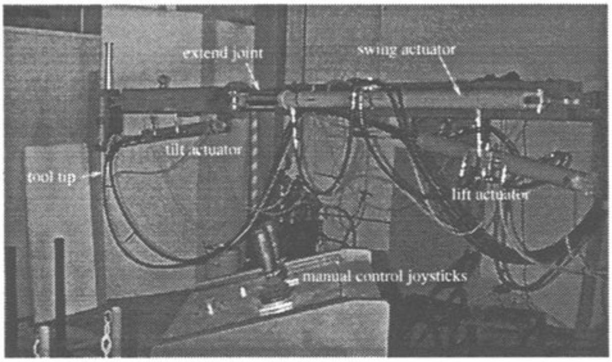


Figure 7. The laboratory electro-hydraulic boom.

3.1. 3D sensing

We are investigating a number of techniques for 3D sensing[7] such as stereopsis, scanning laser rangefinders, and structured lighting. Some early experimental results for the first two of these techniques are given in this section.

The early approaches to computer vision, were based on what would today be called passive computer vision. That is the computer analyses one or more digital images of a scene and attempts to interpret the scene, its components, and their interrelationship. Although seemingly simple, we do this all the time in everyday life, it remains an extremely difficult task for a computer. The human eye relies on many subtle visual cues in order to distinguish objects in a scene and their 3-dimensional structure[7].

Over the last decade an alternative approach, active ranging, has become technologically feasible. Here the scene is subject to controlled illumination, often in the form of a scanned laser, which enables direct measurement of scene 3-dimensional structure. Typically laser time of flight or triangulation between the light source and the camera is used. Much of the inherent ambiguity in attempting to analyse an image of a scene is removed by this direct approach. Such direct 3-dimensional sensing systems have been applied to vehicle navigation[8], and with ranges up to 100 m would seem to have considerable application in mining.

3.1.1. Stereo vision

Passive 3D vision has been extensively investigated for over 20 years by computer vision researchers[7] and for over 150 years by photogrammetrists[9]. Photogrammetry is a mature technology that is already widely used in the mining industry, particularly for open-pit mapping. The automation applications fit into the category of *close-range* or *terrestrial* photogrammetry.

Stereopsis[7] computes range from *disparity*, the phenomenon by which the image of a 3D object point shifts as the viewpoint is moved laterally to the depth axis. A stereo pair is typically taken by two cameras horizontally separated by a distance known as the *baseline*. The fundamental issue is to establish *correspondence* or *matching* of points between the two images in order to derive

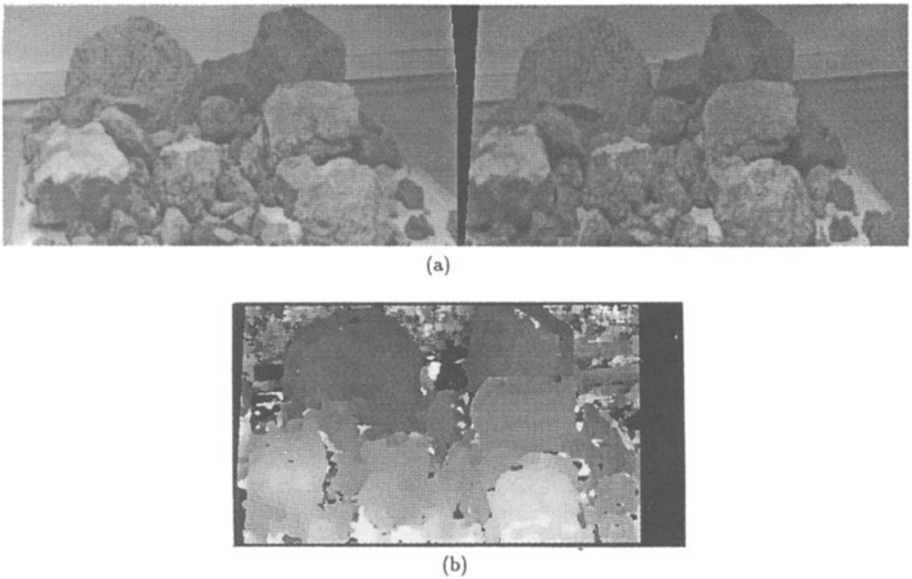


Figure 8. (a) Left and right camera images from the JISCT image database, (b) Dense disparity (inverse depth) image obtained using normalised cross-correlation matching (NCC). Light pixels are closest to the camera (Figures by Jasmine Banks).

disparity, and thence depth. Dense stereo matching involves performing this matching for every pixel in the scene and is computationally intensive.

If the correspondence is to be determined visually there must be sufficient visual 'information' at the matching points to establish a unique pairing relationship. The most common difficulties encountered in practice are where: the images have uniform intensity, regular repetitive texture will result in ambiguous matching, or some part of the scene appears in only one of the views because of occlusion effects (the *missing parts problem*).

The approaches that we have investigated include standard techniques such as sum-of-absolute difference, sum-of-squared-differences and normalised cross-correlation[10], see Figure 8. The latter gives the highest quality disparity images and is the most robust to variations in intensity between the cameras. In practice stereopsis has been found to work well with mining images due to the rich natural texture available. However lighting levels are critical to achieve low noise images which are required for reliable matching. Execution time is of the order of around 20 s for a 512 × 512 pixel image (on a 200 MHz Pentium Pro), but this will reduce in the future as a consequence of Moore's Law.

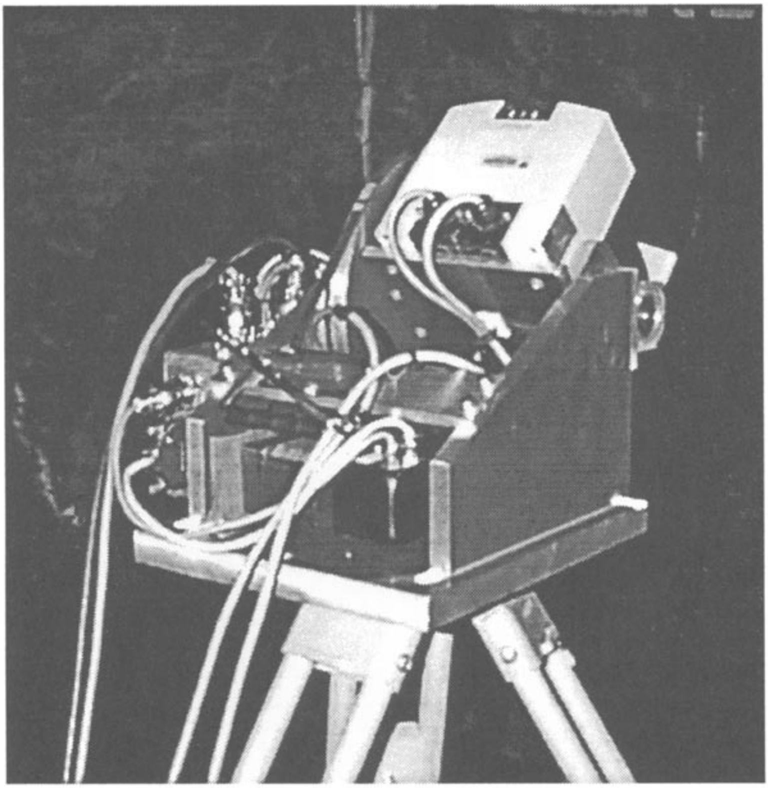


Figure 9. The CMTE Mark I imaging laser range finder, comprises a PLS laser scanner mounted on a custom 'nodding head'.

More recently, non-parametric measures have been proposed[11] which are based on rank or order statistics. Such measures are amenable to custom hardware implementation[12, 13] and can achieve frame rate performance for limited resolution images of 256 × 256 pixels.

3.1.2. Scanning laser rangefinder

The SICK Optic PLS (see also Section 2.1) is a time of flight rangefinder that measures range and bearing to points in a plane. In order to build up an area range image it is necessary to scan or move the device over the target scene. We constructed a "nodding head", see Figure 9, that uses a servo motor to rotate the entire scanner about an axis close to the axis of the sensors internal rotating mirror. The scanning plane can be servoed through 180 degrees using an Industry Pack servo module.

Figures 10 to 12 show some results from the PLS/nodding-head setup. These data were collected during a field trip. Figure 10 shows the results of a number of scans of a working grizzly (see Section 3.2). The diagram at the top left of the figure shows the position of the scanner which was far from optimal, resulting in the data containing shadowed regions behind large rocks. The

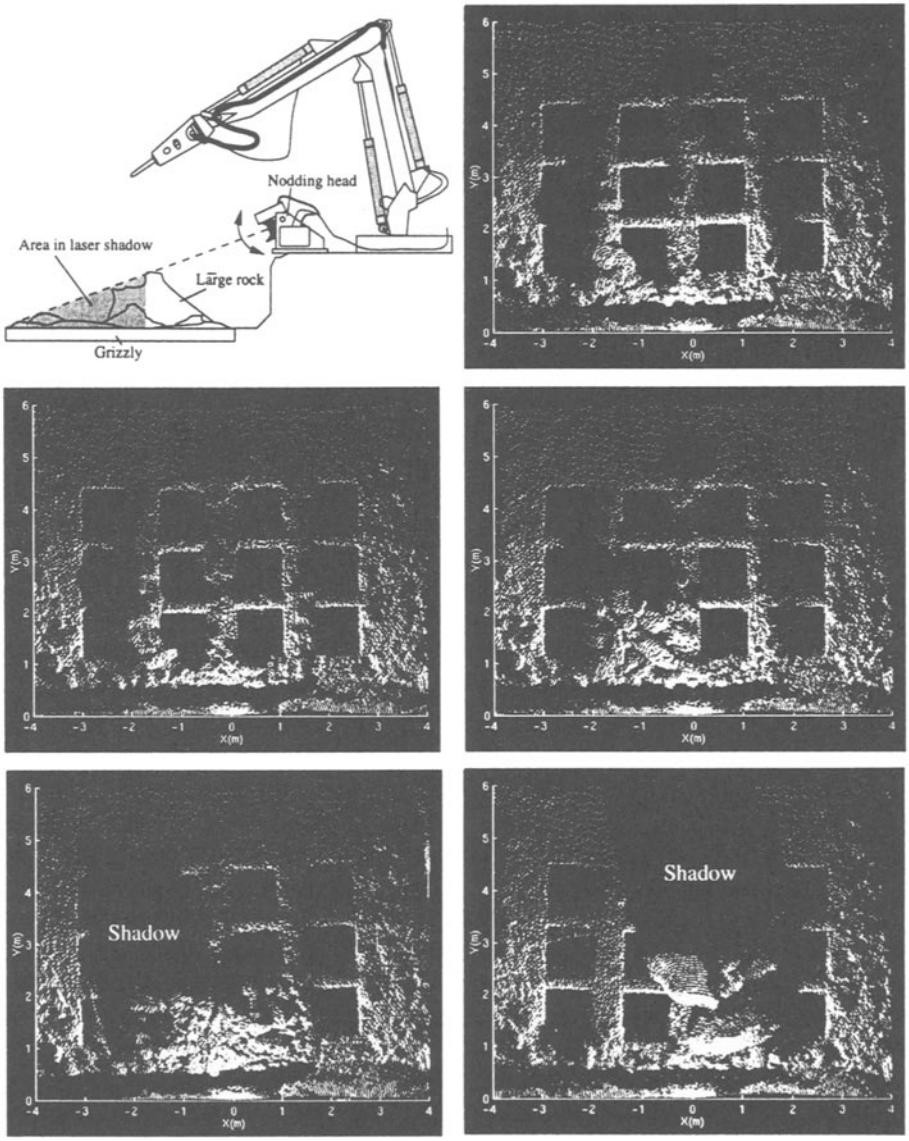


Figure 10. Results from field trip. The diagram in the top left hand corner shows the experimental setup. Note that this setup caused range shadows behind large rocks.

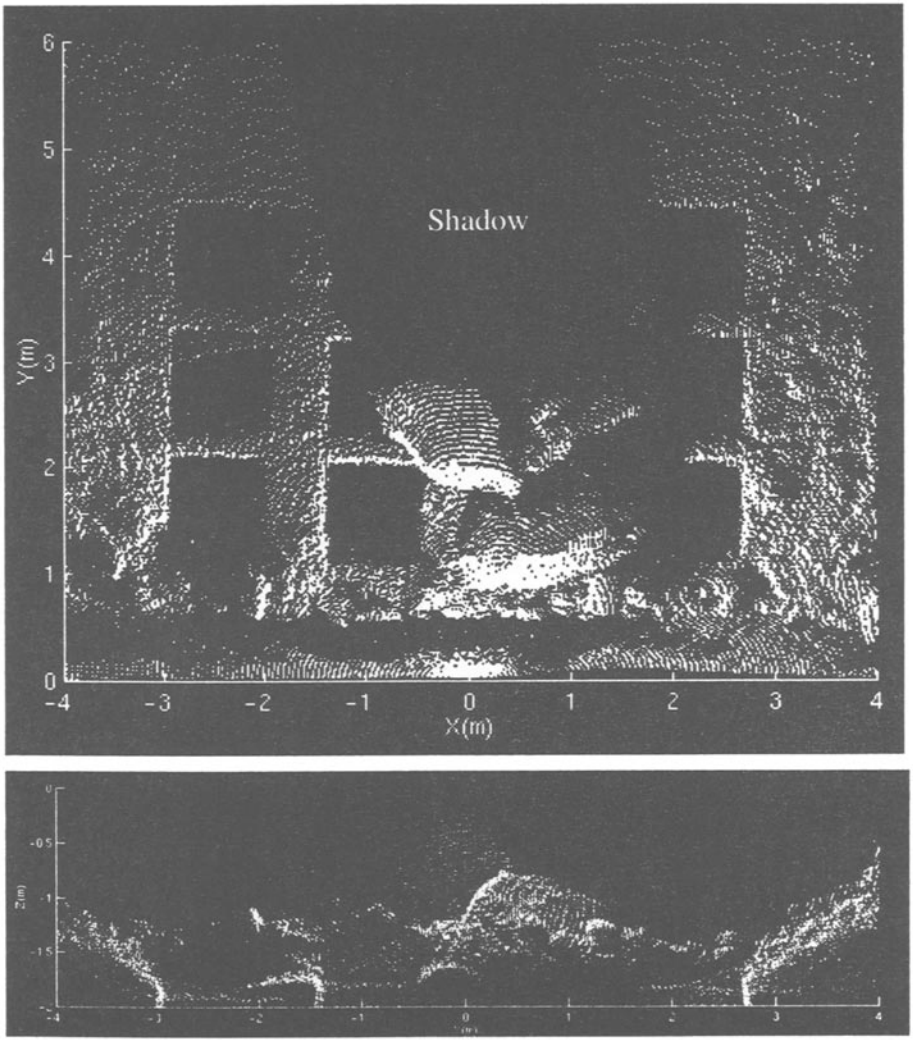


Figure 11. The view from above (top), and from the front (bottom) of the grizzly covered in large rocks.

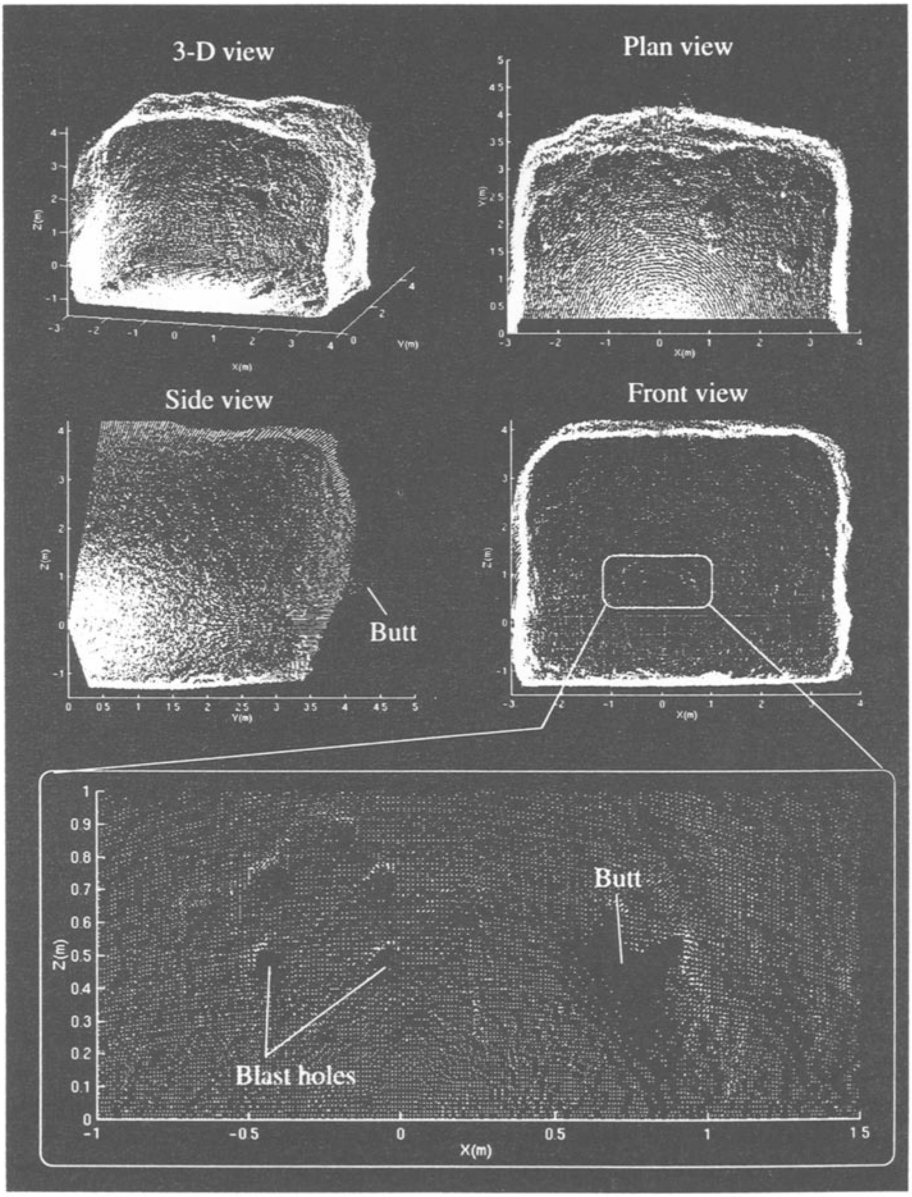


Figure 12. Results from the laser scanner looking at a charging face. Note the enlargement of the front view clearly shows a ring of drilled holes and a butt.

ideal location of the laser scanner would have been directly above the grizzly, eliminating most shadows, but this was not possible given the limited amount of time on site. Figure 11 shows in more detail the data collected when some large rocks were dumped on the grizzly. The lower portion of the figure shows the scene from the front.

Figure 12 shows some different views of the end of a mine tunnel, all derived from one scan. The figure clearly shows details such as the drilled blast holes and a blasting 'butt' (basically a hole in the rock face caused from the previous blasting cycle). The floor appears extremely smooth because it was a pool of water.

3.2. A semi-automated rock breaker

Dealing with oversize rocks is a common problem in surface and underground mining. Large rocks may jam a crushing plant or chute, or be too large to travel along a conveyor system. In underground mines grizzly screens are used to filter out oversize rocks. A grizzly screen is typically a very solid steel structure over an ore pass or crusher that provides a mesh size in the range 0.5 to 1m. Material may reach the grizzly from an ore pass via a chain feeder or be dumped directly by truck. The grizzly screens must be kept clear from build up of loose material or oversize rocks. A rock-breaker, see Figure 6, is a manually controlled hydraulic arm that carries a hydraulic impact hammer. In structure it is akin to a back-hoe excavator, and kinematically it is similar to a standard industrial robot.

In order to automate this process we would need

1. a computer controllable rock breaking boom,
2. a 3-D sensing system,
3. the automation system, and
4. a teleoperation system.

The proposed system would use 3-dimensional sensors to monitor the grizzly, and when necessary control the breaking boom so as to clear the grizzly. The imaging aspects of rock-breaker automation have been previously studied[14, 15] and trialled in the laboratory using a small scale model [16].

Due to the difficulty, in what is a complex and only partially structured environment, of foreseeing all eventualities[2] we do not believe that at this stage it is feasible to fully automate the process. Complications involved in carrying out this task include dealing with foreign objects such as timber props and ground support bolts. In the event of the system being unable to autonomously clear the grizzly, it would signal a remote operator who would use teleoperation of the breaker to complete the task. Such limited human intervention would be the most cost effective solution for dealing with these situations, and would make it possible for a single operator to supervise several rock breakers located at different sites around the mine.

[2] Rock shapes and type, and foreign objects.

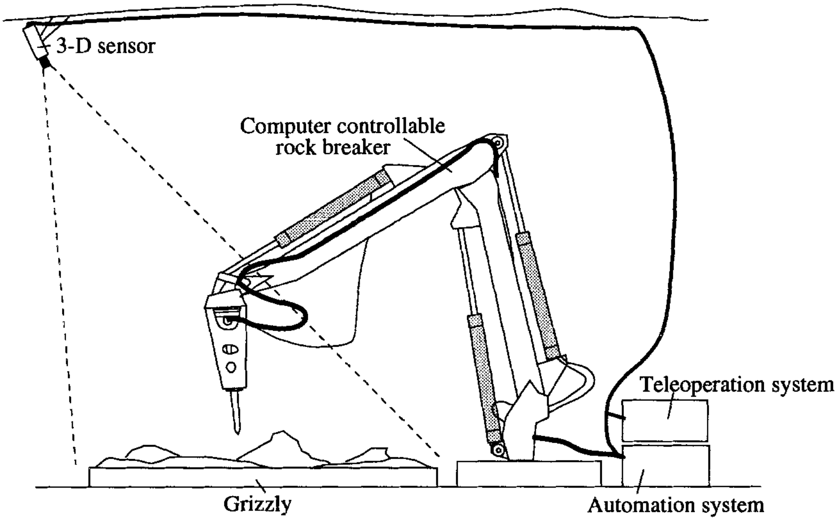


Figure 13. The proposed semi-automated rock breaker.

4. Conclusion

The ultimate aim of mine automation will be to remove miners from the hazardous areas of the mining environment where the work will instead be performed by autonomous and sensate mining machines. Such a vision is many decades from reality and in the interim we can hope to make small steps toward this goal. One step is to increase the productivity of existing mining equipment by assisting the operator so that one operator can supervise several machines. The dragline and the rock breaker automation projects described here are example of this.

Considerable work within the robotics and computer vision is highly applicable to real-world automation needs such as exist within the mining industry. The research community has demonstrated the feasibility of using machine vision to 'close the loop' on the position of robot manipulators. Such technology could be usefully applied to many applications in mining. Superficially these may seem very different problems, but this is largely a matter of scale. Larger machines in fact require reaction times that are considerably longer than those being demonstrated now in robotics laboratories. The biggest, but not insurmountable, challenge is the complexity of scene analysis in a complex mining environment.

Acknowledgements

The authors gratefully acknowledge the help of their colleagues Stuart Wolfe, David Hainsworth, Stephen Nothdurft, Zheng-De Li, Jasmine Banks, Hal Gurgenci, Don Flynn, Peter Nicolay, Allan Boughen and Daniel Sweatman. The dragline project is funded by a consortium consisting of the Australian Coal Association Research Programme (as project C5003), Pacific Coal Pty Ltd, BHP Australia Coal Pty Ltd and the Cooperative Research Centre for Mining Technology and Equipment (CMTE), a joint venture between AMIRA, CSIRO and the University of Queensland. Bucyrus Erie Australia and Tritronics Pty. Ltd. have provided valuable in-kind support, and Tarong Coal have generously allowed the automated swing system to be installed on their BE1370. The underground mining robotics work has been supported by the CMTE and AMIRA project P440 which was sponsored by Mount Isa Mines, Normandy Poseidon and Western Mining.

References

[1] P. I. Corke, *Visual Control of Robots: High-Performance visual servoing.* Mechatronics, Research Studies Press (John Wiley), 1996.

[2] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, pp. 651–670, Oct. 1996.

[3] D. Hainsworth, G. Winstanley, Y. Li, P. Corke, and H. Gurgenci, "Automatic control of dragline operation using machine vision control of bucket position," in *Proc. First CMTE Annual Conference*, (Brisbane), pp. 111–114, July 1994.

[4] P. I. Corke, G. Winstanley, and J. Roberts, "Dragline modelling and control," in *Proc. IEEE Int. Conf. Robotics and Automation*, (Albuquerque, NM), pp. 1657–1662, 1997.

[5] G. Winstanley, P. Corke, and J. Roberts, "Dragline swing automation," in *Proc. IEEE Int. Conf. Robotics and Automation*, (Albuquerque, NM), pp. 1827–1832, 1997.

[6] Z. Li, P. Corke, and H. Gurgenci, "Modelling and simulation of an electro-hydraulic mining manipulator," in *Proc. IEEE Int. Conf. Robotics and Automation*, (Albuquerque, NM), pp. 1663–1668, 1997.

[7] R. A. Jarvis, "A perspective on range finding techniques for computer vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-5, pp. 122–139, Mar. 1983.

[8] R. Chatila, S. Fleury, and M. Herrb, "Autonomous navigation in natural environments," in *Experimental Robotics III* (T. Yoshikawa and F. Miyazaki, eds.), pp. 425–443, Springer-Verlag, 1994.

[9] M. Thompson, ed., *Manual of Photogrammetry.* Falls Church, VA: American Society of Photogrammetry, 3 ed., 1966.

[10] P. Fua, "A parallel stereo algorithm that produces dense depth maps and preserves image features," *Machine Vision and Applications*, vol. 6, pp. 35–49, 1993.

[11] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proc. 3rd European Conf. Computer Vision*, (Stockholm), May 1994.

[12] P. Dunn and P. Corke, "Real-time stereopsis using FPGAs," in *Proc. Intl. Workshop on Field Programmable Logic,*, (Imperial College, London), Sept. 1997.

[13] J. Woodfill and B. V. Herzen, "Real-time stereo vision on the parts reconfigurable computer," in *IEEE Workshop on FPGAs for Custom Computing Machines*, pp. 242–250, Apr. 1997.

[14] S. Elgazzar, J. Domey, P. Boulanger, and G. Roth, "Three-dimensional imaging for mining automation," in *Proc. 5th Canadian Symp. on Mining Automation*, pp. 334–340, 1992.

[15] J. Domey and M. Rioux, "3-d vision sensors and their potential — applications in mining automation," in *3rd Canadian Symp. Mining Automation*, (Montreal),

pp. 187–193, Sept. 1988.

[16] C. Cheung, W., Ferrie, R. Dimitrakopoulos, and G. Carayanis, "Towards computer vision driven rock modelling," in *Proc. 2nd Canadian Conf. on Computer Applications in the Mineral Industry*, (Vancover, B.C.), Sept. 1991.

HelpMate®, The Trackless Robotic Courier: A Perspective on the Development of a Commercial Autonomous Mobile Robot

John M. Evans
HelpMate Robotics, Inc.
Danbury, CT, USA
evans@helpmate.com

Bala Krishnamurthy
HelpMate Robotics, Inc.
Danbury, CT, USA
bala@helpmate.com

Abstract

HelpMate Robotics has developed an autonomous mobile robot courier for material transport in hospitals. These machines are in operation around the world, operating in uncontrolled and unsupervised environments up to 24 hours per day. The history and technology of the HelpMate robot are presented in this paper with the intention of providing a real world perspective on transitioning technology from the laboratory to the marketplace.

1 Introduction

HelpMate Robotics Inc. (HRI), of Danbury, Connecticut, USA, has developed a hospital courier robot that is the benchmark of commercial success in the emerging field of service robots. This paper will explore the decade of development of the HelpMate robot, focusing on the evolution of the technology from laboratory exploration to hardened commercial product.

In trying to define market opportunities for robots in service applications and to match enabling technologies against those opportunities, several possibilities originally stood out: floor cleaning, security, hazardous environments and hospital materials transport. All required autonomous mobile robots navigating in indoor structured environments. It was this technology base and these markets on which we focused early attention.

1.1 The Hospital as a Target Market

Health care is an obvious market for automation because of the high and rising costs that must be tamed if any attempt at universally available care is to be offered in our society. Cost containment is an overriding theme in health care management, and robotics is one technology to reduce labor costs. Operation of hospitals around the clock on a multi-shift basis makes capital justification easier, so this was a natural target.

Development of the HelpMate concept has been described in earlier papers [1,2,3,4,5]. The final machine, shown in Figure 1, is able to navigate in crowded hallways, avoiding people and inanimate obstacles as it encounters them, and using the walls of the hallways as the principal navigation reference. Figure 2 shows the applications for HelpMate.

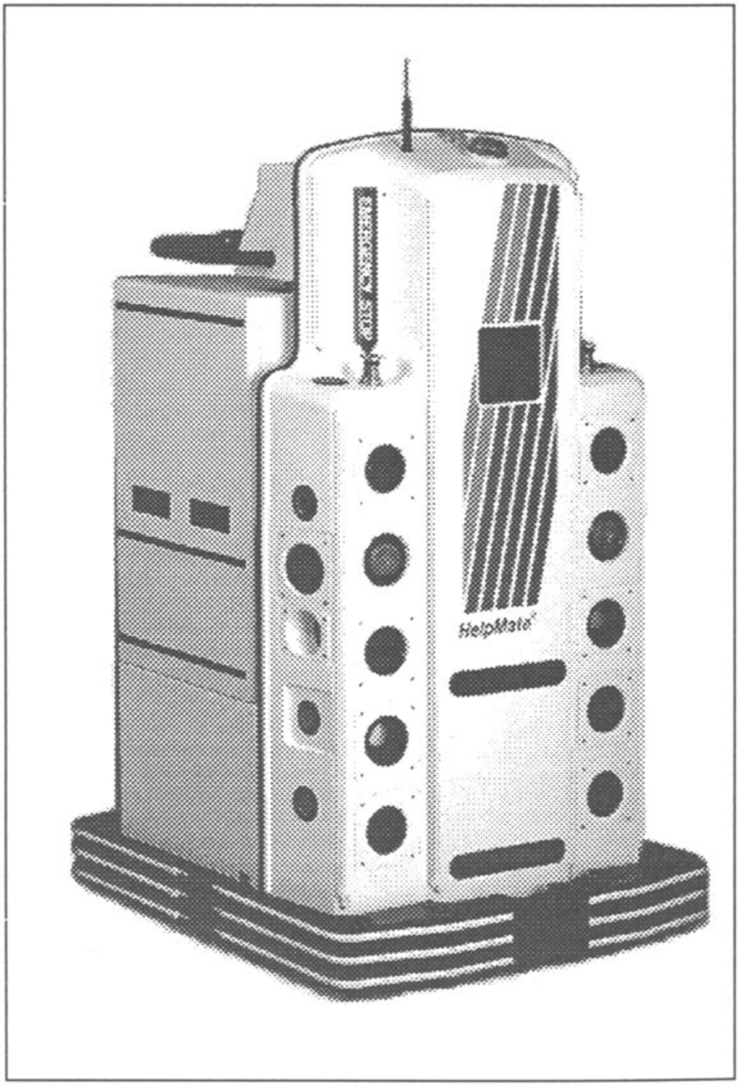


Figure 1: HelpMate Robot Courier

Dietary: late and special request trays
Supply: equipment and material
Lab: speciman and sample transport
Pharmacy: medication and supplies
Med Records: patient files
Administration: mail and reports
Radiology: films
Mail: mail and packages

Figure 2: Applications of HelpMate

2 Navigation: a problem in Sensing

Over the years, we have come to believe that the key to autonomous mobile robot navigation is sensory perception. If you can obtain a good, dynamic, high resolution picture of the world around the robot, then you can successfully use any of several algorithmic approaches to planning a collision free path through the environment.

Much of the early work on navigation algorithms hypothesized known, static worlds, what we would call "blocks worlds" after the highly structured and artificial worlds used in early machine vision research [6]. The heritage is that of trajectory planning for robot arms in free space, with force or compliance control considered as a special case [7]. Work is still presented today on such ideas. Potential field models, clothoids, path planning for non-holonomic vehicles: all presume a totally known and static environment [8,9,10]

But the real world is not static. Hospital hallways, in particular, can be filled with moving people: employees, patients and visitors. So the problem becomes one of dynamic sensing, and control strategies must be dynamic and reactive.

Further, it turns out that there are many "stealth" objects in the real world for any single sensory modality. Sound waves bounce off any hard, flat, angled surface and are absorbed by soft material such as a blanket. Light bounces off mirrors and chromed surfaces in a specular fashion and passes through glass, and light is absorbed by flat black material such as black wool pants. Hence, a multiplicity of different types of sensors maximizes the probability of sensing objects prior to collision. Contact sensors to detect collision are always required.

The HelpMate robot uses sonar, vision, and contact sensors to interact safely with people and obstacles in a hospital world.[3]

This combination of sensors is not unique and was in fact already known in the research community before we started working on HelpMate. During the early and mid-1980's there was a great deal of interest in autonomous mobile robots and some very good work had been done before we started, going back to the 1960's. [11,12,13,14]

In fact, after reviewing the literature and talking with some of the researchers at leading universities, we felt when we started that developing commercial mobile robots would be a straightforward matter of technology transfer of academic work and then solving applications problems. Thirteen years later we are still trying to tie up loose ends, but we finally have over one hundred machines running in the real world, in uncontrolled environments, without supervision, 24 hours per day, seven days per week, providing useful and cost effective transport services. It has been anything but straightforward, as we will review in Section 4 below.

2.1 Sonar

Ultrasonic ranging is a basic modality for HelpMate. Some 28 different transducers are located around the robot, looking for walls, for obstacles and for overhanging objects. There are even transducers in the bumper looking straight down to see any sudden drop off such as a stairway or the edge of a loading dock.

HelpMate uses the Polaroid electrostatic microphone transducers [15]. These sensors were developed for focusing instant cameras. A foil microphone is used for both transmitter and reciever. The microphone is driven by a 300 Volt train of pulses, and a 150V bias is held to allow sensing of the return sound energy. We use the Texas Instruments version of the asic drive chip to activate the transducers. This provides a driving signal of 16 pulses at a single 49.4 KHz frequency. The time of flight measurements begin with the start of the pulse train and end when enough energy is returned to the microphone to exceed a trigger threshold. The gain of the amplifier driving the trigger circuit is increased with time to compensate for loss of signal strength for returns from more distant targets.

The main uncertainty in sonar measurement is direction of the return signal. Researchers usually characterize the Polaroid field of view as 15° which is useful as a general guideline. For example, the "Denning Ring" is the classic array for the Polaroid sensors when used on robots with cylindrical geometry [16], with 24 sensors around the circumference of the robot, one every 15°. However, the microphones have a radiation pattern which is like a typical antenna pattern, the 15° being the 6dB width of the central lobe. Side lobes can be excited as far as 60° and more from normal. With the increasing gain in the detection circuit with time, the chance of triggering on multi-path echos is significant.

Despite these measurement uncertainties, sonar and particularly the Polaroid transducers have been a favorite choice of both researchers and entrepreneurs since 1980 for an obvious reason: price. Over two million dollars went into development of the original sensor and drive circuitry and in volume it is now possible to procure those sensors for less than $10 and less than $30 including the electronics. Nothing else is competitive, so a large number of man years have gone into coping with the problems of the Polaroid sensors [17,18,19]. The same expenditure of effort might have borne more fruit in developing alternatives [20,21,22]

After years of struggle we were able to effectively use the Polaroid sensors. The HelpMate works, so we stay with those sensors.

2.2 Vision

Vision has been the main line of research for robotics and artificial intelligence since the 1960's. Machine Vision has proven to be substantially more difficult than anyone supposed it would be thirty years ago [23,24,25], but it is now both effective and ubiquitous and, in the past several years, it is also become a low cost sensing modality [26].

HelpMate must be very flexible in sensing its environment because obstacles will cause it to deviate substantially from a planned route down the center of a hallway. Vision seemed from the first to be an attractive modality because of the richness of information and the structured geometry of the world in which the robot was to operate.

Over the years, we have experimented with one and two dimensional images, with views near the floor, normal and oblique views of the ceiling, optic flow of vertical features, and structured light vision systems. Two concepts were deployed originally with the first models of the HelpMate, one remains in the current design, the structured light system [27].

Ceiling vision, which is no longer used, uses an image of the hallway ceiling lights to extract information about the relative orientation and position of the robot in its environment [28]. The ceiling vision camera was pointed obliquely upward, giving a long forward view of the ceiling. Ceiling lights and tiles were used to find the center of the hallway and its alignment.

The structured light vision system is used for obstacle detection and avoidance [29]. A camera is pointed at a 45 degree angle toward the floor from near the top front of the robot. Two light projectors send out planes of light parallel to the floor, one at ankle height and one at knee height. The beams cut across the camera field of view, giving a range for sensing obstacles of about two meters for the bottom strobe and one meter for the top one.

The camera will see an empty field of view if there is nothing in front of the robot. If there is an obstacle, the planes of light will intersect with the obstacle, producing a contour line of the surface which the camera will see. Processing the camera image, the distance from the bottom of the image to the first illuminated scan line is a measure of distance to the obstacle, and the horizontal width of the contour line indicates its size.

Optical filtering, electronic shuttering, and frame-to-frame differencing are used to improve signal-to-noise and immunity to ambient lighting.

2.3 Contact Bumpers

For safety, the robot must be able to sense contact. There will always be some obstacles in the environment that will defeat any finite array of non-contact proximity sensors, so there must be a further line of defense.

HelpMate uses Tapeswitch™ sensors along the bumpers and vertically on corners. These switches have an activation force of about six ounces, so they are very sensitive to contact. The bumpers are mounted on elastomeric supports, so they will move under a force of several pounds. Infrared proximity sensors detect motion of the bumper and act as an additional and redundant collision detector.

2.4 Additional Sensory Modalities

The final sensory system that we added to the HelpMate was an absolute position reference system. For long stretches of carpeted corridors there will be enough wheel slippage that the robot can be off of its estimated position by tens of centimeters. This is enough to cause problems in aligning with elevators or in navigating through cluttered halls. To overcome this problem we have used retroreflective tape affixed to the ceiling on the hanger bars for acoustic tiles, a standard ceiling design. A pair of long range infrared proximity sensors are mounted on the shoulders of the robot, pointed at the ceiling. These sensors see the ceiling tape and provide information on both position and orientation at those landmarks.

We have used this ceiling tape system for safety as well as for navigation landmarks. At any point where there is an open stairwell or loading dock that would create a hazardous environment for the robot, we place a wide section of reflective tape on the ceiling, and this is used as by the robot as a warning. The robot also has staircase detectors, so there is a "belt and suspenders" redundant sensing approach for critical safety situations.

3 Navigation: A Problem in Control

With perfect odometry and in the absence of obstacles, navigation is reduced to simple path following. In a real world environment, a combination of different sensor modalities and control programs is needed to compensate for odometry errors and to avoid obstacles.

3.1 Odometry

Odometry or dead reckoning (properly ded reckoning, from deduced reckoning, an old Navy term describing the estimation of position of ships from velocity and time and heading measurements) is the basis for almost all mobile robot systems. Odometry or dead reckoning is used to refer to measurement of the robot's position and heading from wheel encoder readings and other internal position and velocity and acceleration sensors such as gyros or accelerometers or doppler velocity sensors. A definitive survey is provided by Borenstein, Everett, and Feng. [30]

Odometry errors accumulate with time and depend upon the environmental conditions. A rough or slippery or carpeted floor will produce greater errors than a smooth tile floor, for example.

Navigation over a few meters, and potentially over tens of meters, is possible with only odometry. At some point in any system, however, reference must be made to outside landmarks to re-establish a position estimation that is correct with respect to the external environment. This is the process of taking sensory data about the external world and matching that to an internal map of the environment. We refer to that process as localization. Position corrections are required both along the direction of travel (Y direction) and perpendicular to the direction of travel (X direction).

3.2 Localization

A prerequisite for acceptable robot navigation, in addition to adequate sensing, is either detailed information about the environment or robust navigation algorithms that compensate for the lack of *a priori* information. The HelpMate is provided with some *a priori* information and endowed with enough intelligence to deal with the inexact world model and to rationalize the *a priori* information with the dynamic sensed model. The definition of 'some' and 'enough' has taken over 10 years to refine!

Once a world model is available, it is possible to use sensory data to match the position of the robot to that model. This can be as simple as using single sensor data for wall following [31], or as complex as matching a complete local map with a global map [18,32]. Most work in the literature has matched features extracted from the sensor data to corresponding elements of a global model [33, 34]. This is the approach we use for HelpMate.

HelpMate navigates in what we would call a structured environment, with rigid and unchanging geometric features of the building within the range of the sensors of the robot. As such, the natural approach to localization is to use the sonar sensors to measure the range to the walls of the hallways and to use the walls as the primary X position reference. Exactly where the robot should travel with respect to those wall is information in the *a priori* model, which we call the Topography.

Raw sonar sensor data, along with hallway wall information supplied via the Topography, is used to estimate the X position and the heading. A line fitting routine is used to extract a line segment from the list of sensor data representing the wall. This is similar to the approach taken in the mid-80's by Crowley to match line segments extracted from sonar data with those from a model [33]. The wall registration module on the HelpMate runs every 150 ms and generates position and heading corrections off wall segments.

Unless the hall is a particularly long one and is followed by an area where the positioning must immediately be exact or where the walls are often expected to be obscured, the HelpMate needs no correction in the Y direction. This is because once a turn is made into the next hall, any Y error in the previous hall is now translated into an X error, and is usually corrected by the wall registration algorithm within a few meters of travel.

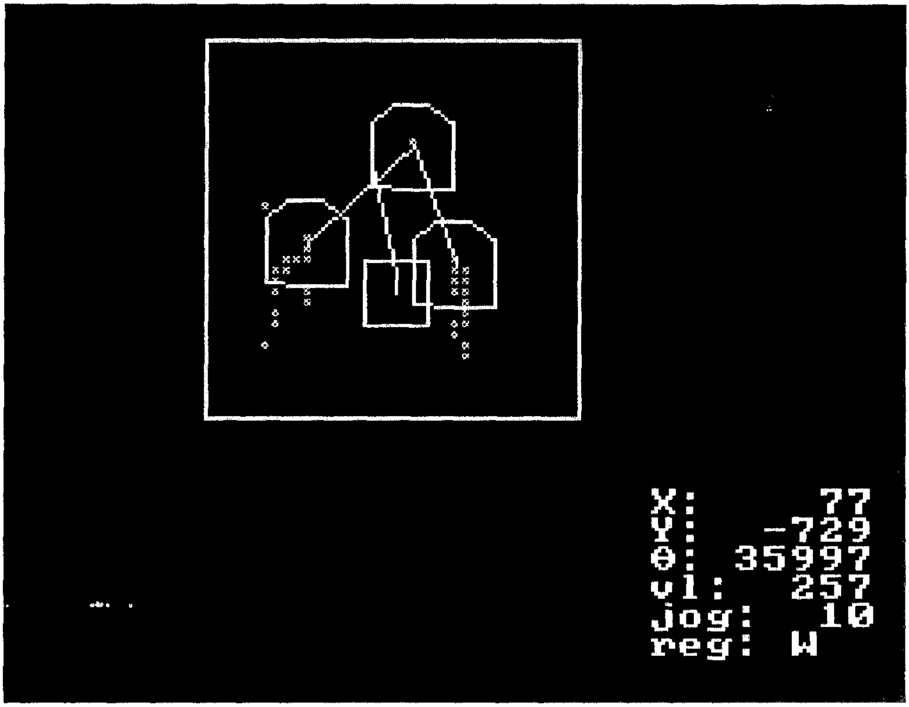
Augmentation of natural landmarks is required in areas where the robot may be required to travel long distances on carpeted areas, resulting in wheel slippage, and then turn immediately into an elevator lobby area where precise positioning is required. In cases such as these retroreflective tape is affixed on the ceiling just before the end of the long hall. The tape is detected by the upward looking IR sensors and the tape registration module, running every tick time, generates a correction to both the Y position and the orientation of the robot.

3.3 Obstacle Avoidance

The primary responsibility of HelpMate is to operate safely, to avoid running into anything, as it navigates down hallways. This requires a multiplicity of sensors, as described in the previous section, and a way of combining and processing that sensory data in a coherent way.

Combining data from multiple sensor modalities is often called sensor fusion. There are many techniques in use, including fuzzy logic [35], subsumption and other state machine architectures [36] and various mapping schemes. HelpMate uses a mapping approach that was derived from work at NIST by Albus and co-workers[37], and work at CMU and Grenoble by Crowley [33] and Elfes[17].

By the time HelpMate was started, the NIST work on hierarchical control had evolved to the concept of interjecting an explicit world model hierarchy between a sensor processing hierarchy and the control hierarchy. The idea was to use the sensor data to servo the model and then to use the model data for control calculations. We adopted this concept for HelpMate, using a Cartesian occupancy grid representation similar to Elfes to combine data from sonar, from vision and from contact bumpers, as shown in Figure 3.



```
X:     77
Y:   -729
θ:  35997
vl:   257
jog:   10
reg: W
```

Figure 3: Graphical Picture of Local Map as HelpMate Avoids an Obstacle

HelpMate's immediate environment is represented in Cartesian coordinates, unlike Crowley's polar map[33] or that of Borenstein and Koren [38]. The local map is a robo-centric occupancy grid representation with 35 X 35 cells, each representing a 128mm square, which works out to be approximately a 4.5 meter x 4.5 meter area around the robot, but asymmetrically weighted with 2/3 of the cells ahead of the robot and 1/3 behind..

This local map is refreshed every 150 milliseconds with new weighted sonar, vision, shield and contact bumper sensor information. The information pooled into this local map is weighted dependent on the rate at which the data is gathered, the reliability of the data and the importance of the data. Vision data, for example, is given a higher weighting that sonar, because the data is less frequent and less prone to error. The contact bumper data is mapped with an even higher certainty since it is fleeting, was obviously something that escaped other sensor detection, and must be retained in memory until the object is safely passed.

Dynamics are handled by overlaying a time based decay which slowly depletes the memory of previously seen objects while retaining information long enough to get around an obstacle that escapes detection by most of the sensors.

All obstacle detection is done using this composite local map. This map is scanned every 150 ms for available openings that lead to the final destination either directly or by deviating around an obstacle. One opening is selected using path heuristics, and position commands that direct the robot towards this opening are generated.

The commands to the drive module from the navigation module consist essentially of forward and angular velocity every tick time. The 'set_velocity' and the 'jog' command interface provided by the drive subsystem facilitates setting of the velocity and the rate of turn in units of degrees per second. This interface is basically that proposed by Crowley, although he adds position and acceleration in each command and expects back an estimation of position uncertainty. We deal with position uncertainty at a higher level as part of navigation.

HelpMate's path detection, selection and tracking calculation modules run every 150ms, and are based upon the configuration space approach where the robot is represented as a point object and the obstacles are grown by the clearance required to safely pass by [39]. A list of possible paths is generated by the path detection module which restricts its search to the boundaries of the hallway, allowing for position and orientation uncertainty. The openings detected are checked against the size of the robot, including the clearance required for safe navigation. The right and left edges of each opening are then represented as polygons large enough to provide the specified obstacle avoidance distance. Paths are plotted to the closest edges of these polygons, as graphically shown in Figure 3 above.

4 The Real World: Problems and Solutions

Between the dream and the reality, as has been often noted, there can be quite a gap. Real world applications are where substantial engineering problems can cause budget and schedule overruns as theory and laboratory demonstration fail in practice. The classic joke runs that the first 95% of the project is not so hard, it is the second 95% that is difficult. In trying to develop a new product for a new market, this is often the exact truth, since a product is not commercially viable until it truly solves a customer need, reliably, at an affordable price, and the problems to be overcome are unknown at the start of the project.

Between early 1988 and the end of 1990, some two and one-half years, we struggled to move autonomous mobile robot technology, which performed well enough in laboratory demonstrations, into the marketplace.

In January of 1991 we finally walked away from our first installation, in Danbury Hospital in Danbury, Connecticut, and asked them to call us if they had problems. In April we went to 24 hours per day and in June the robot was accepted and the hospital began paying rent. Additional sites were operational by the end of 1991, and in 1992 sites with multiple robots were operational. Since 1992 there have been incremental improvements in reliability, in cost, and in functionality, but the HelpMate was essentially operational at that point in time as a successful product.

This section will review some of the problems that were encountered and overcome in the 1988-1990 time period, the transition from laboratory technology to product technology.

4.1 Dynamics Problems

When we started on obstacle avoidance in 1986 and 1987, we worked in a laboratory setting with static obstacles, primarily concerned with getting autonomous floor cleaning machines to avoid pallets, displays, and shopping carts in supermarkets in the middle of the night. The first HelpMate prototype was very good at running mazes in our lab before we got to the field, and it would stop very nicely when someone stepped in front of it and then find a way around the person if they stood still.

The first field test, at the end of 1987, was from the Dietary department to the elevator bank in the next building. We arrived at about 9 AM and had the system programmed and set up to run about 11:30 AM. The service hallway had been fairly empty through the morning, but suddenly, just as the robot started off, the hallway was filled with people! The reason: the cafeteria opens at 11:30 AM for lunch service. The poor HelpMate prototype never made it to the elevator that day.

By 1989 the robot could avoid dynamic, moving obstacles with some alacrity and grace. In fact, the robot was sufficiently responsive to moving obstacles that two problems appeared in the hospital field trials. First, if the robot was traveling down the hall and a person cut across in front of it to go through a door, the robot would be turning away (toward the wall) as the person moved in front of it. This created an appearance that the robot was attacking the person, as both robot and person would get to the door at about the same time. This caused a series of complaints as to the robot's aggressive behavior.

A second problem was interactions with the staff, particularly the doctors, who still show a continuing fascination in playing with the robot and demonstrating its obstacle avoidance capability. We had made the robot so responsive that it was a very entertaining demonstration to jump back and forth in front of the robot, creating a very impressive and comical dance.

The solution to both of these problems was to slow down the robot's responses, to make it slow and sluggish and dull in its behavior, plus a number of situation specific rules in the navigation code to take care of many different cases. This was the start of struggling with what we have called "geometric reasoning" in behavior [adopted from Crowley, 40]. Hundreds of context specific rules were eventually developed to make the behavior of the robot acceptable in a hospital environment.

4.2 Thermal Problems

The first prototype robot, serial number 1, had a foamcore shell and a laptop sitting on top of it. It was never intended to operate without a TRC employee in attendance and was a development platform for sensors and navigation software.

The second prototype, with a fiberglass shell, was completed during the late spring of 1988 and taken to the Robots 12 exhibition in Cobo Hall in Detroit. Time ran short, and the first time the shells were attached to the robot and buttoned up was on the exhibition floor. This show was in June, and during set-up the outside doors were open, the air conditioning was turned off, and the temperature approached 40°C on the exhibition floor.

When we finally started the robot, it was unable to complete the demonstration. It would make it around the booth, which was about 20 m long, at most once before dying. The problem turned out to be overheating of the electronics, resulting in resetting of either the main cpu or loss of interprocessor communication. We eventually ran demos successfully, but we had to remove the back shell of the robot between demos for it to cool down. A redesign of the electronics layout allowed adequate passive convective cooling.

4.3 Static Electricity Problems

The winter of 1988-1989 turned up two problems: static electricity and sonar crosstalk. Many of the service areas in a typical hospital are heated but not humidified. During the winter, the relative humidity can drop below 20%, leading to static electricity problems.

The robot can, to some extent, be considered as an electrostatic generator moving on a dielectric surface. There is no way to discharge a self generated charge until part of the frame or metal bumper touches a grounded metal surface such as the frame of the elevator door. Even worse are people touching some exposed part of the robot with a relative potential of 30,000 volts or more.

Since there is no way to avoid these incidents, the solution is to shield all of the internal electronics, insuring good frame ground paths for any externally induced charge and avoiding any vulnerability of logic gates tied to sensors or communications links.

A recent novel by Clive Cussler had the hero, Dirk Pitt, trapped in a secret laboratory with an army of robots that were going to be released to conquer the world [41]. Pitt succeeded in disabling the robots with a jolt of static electricity to their bodies. The story had some technical substance, but we could not help but feel that the robot army required further engineering before it was ready for the field.

4.4 Sonar Crosstalk Problems

The other winter problem we encountered was sonar crosstalk. The coefficient of absorption of cool, dry air can be more than 1 dB per meter lower than that of warm humid air [15]. The result: coherent ultrasonic energy produced by the sonar transducers is dissipated quickly in the summer and can bounce around in an echoic space that is not humidified for a long time during the winter.

The HelpMate robot has 28 different sonar transducers that are fired in approximately 300 milliseconds. This produces a great deal of sound energy that can bounce around for quite a while, and crosstalk becomes a serious problem. The most visible effect is "ghosting" where the robot will avoid non-existant obstacles encountered along the hall.

This problem occured every year, for several years, starting about Thanksgiving. We would diagnose the problem, propose solutions, prototype and test them in the lab, and deploy them by about February or March. Each year we thought we had successfully solved the problem, but we had of course only gotten through the coldest months of the year and the problem was disappearing on its own. The next year it was back.

We eventually came up with techniques for verifying true echoes that make the occasional remaining mid-winter ghosts a tolerable annoyance.

4.5 Electromagnetic Compatibility Problems

The FCC and the EC have standards on radio emissions which are very stringent [42,43]. Essentially, the robot has to be as quiet (as a radio emitter) as a local radio station at a distance of only two meters, over the entire electromagnetic spectrum. And it has to be invulnerable to radiated power one million times more intense.

These are not trivial requirements to meet, since every cpu and every clock circuit and every logic chip is a source of radio energy, and every wire linking a board to another board or to a sensor is an antenna. Filtering and shielding were required on every subsystem.

Like many other companies, we took over a year to meet these requirements, a painful process, with substantial redesign of most of the components of the robot.

4.6 Reliability and Service Problems

This is something every entrepreneur has to deal with in introducing a new product. Our mindset in starting the development process was one of rapid prototyping, trying something, building a breadboard as fast as possible, testing it, discarding what failed and trying something else as quickly as we could. The result was a functioning but fragile technology base when we were finally in the field.

The other winter problem we encountered was sonar crosstalk. The coefficient of absorption of cool, dry air can be more than 1 dB per meter lower than that of warm humid air [15]. The result: coherent ultrasonic energy produced by the sonar transducers is dissipated quickly in the summer and can bounce around in an echoic space that is not humidified for a long time during the winter.

The HelpMate robot has 28 different sonar transducers that are fired in approximately 300 milliseconds. This produces a great deal of sound energy that can bounce around for quite a while, and crosstalk becomes a serious problem. The most visible effect is "ghosting" where the robot will avoid non-existant obstacles encountered along the hall.

This problem occured every year, for several years, starting about Thanksgiving. We would diagnose the problem, propose solutions, prototype and test them in the lab, and deploy them by about February or March. Each year we thought we had successfully solved the problem, but we had of course only gotten through the coldest months of the year and the problem was disappearing on its own. The next year it was back.

We eventually came up with techniques for verifying true echoes that make the occasional remaining mid-winter ghosts a tolerable annoyance.

4.5 Electromagnetic Compatibility Problems

The FCC and the EC have standards on radio emissions which are very stringent [42,43]. Essentially, the robot has to be as quiet (as a radio emitter) as a local radio station at a distance of only two meters, over the entire electromagnetic spectrum. And it has to be invulnerable to radiated power one million times more intense.

These are not trivial requirements to meet, since every cpu and every clock circuit and every logic chip is a source of radio energy, and every wire linking a board to another board or to a sensor is an antenna. Filtering and shielding were required on every subsystem.

Like many other companies, we took over a year to meet these requirements, a painful process, with substantial redesign of most of the components of the robot.

4.6 Reliability and Service Problems

This is something every entrepreneur has to deal with in introducing a new product. Our mindset in starting the development process was one of rapid prototyping, trying something, building a breadboard as fast as possible, testing it, discarding what failed and trying something else as quickly as we could. The result was a functioning but fragile technology base when we were finally in the field.

We went from a mission success rate of about 70% in mid 1988 and trips of up to 30 minutes or more to a success rate of 98+% with trips of 15 minutes or less over the same routes by the end of 1991. The mean time between failure in 1988 was almost measured in minutes, in 1991 we were in the few hundred hours and now we are somewhere around 2000 hours for MTBF in terms of hardware reliability. Service, which was a major problem for us in the early years, has ceased to be a concern and will soon be contracted to a third party with a nationally distributed network of service technicians.

Hardening will be continued; current industrial robots are in the 20,000 hour MTBF range, a failure rate of once every five years for two shift operation. We will continue to work toward such a level of reliability.

4.7 Man-Machine Interface Problems

The original model for a man-machine interface was an automated teller machine (ATM) with a screen showing menus of options and a numeric keypad for making selections of options and for entering numeric data such as floor numbers. Since some hospital workers do not read, consistency and simplicity are requirements.

We originally used a "beep-beep" acoustic signal to draw attention to the robot and displayed messages on the screen such as "Please unload compartment one, then press the green button". We found that the performance of the robot was variable, and in many cases unacceptable, in terms of trip times. This was traced to the nursing units, where we found the nursing staff was ignoring the robot, leaving it sitting in the hall with a rapidly cooling meal in its backpack.

We asked the nurses why they were ignoring the robot and they said they didn't know it was there. We asked if they hadn't heard the "beep" and they said, "Oh, everything in the hospital beeps. We don't pay any attention to beeps."

This led to the addition of a voice output module to the robot, so now messages are given as a recording. Both male and female messages are available, and translations have been made into Japanese and several European languages.

The voice output system resulted in a dramatic improvement in the acceptability of the robot, and even untrained users are able to successfully receive the correct payload. The robot encounters visitors, patients and staff as it moves through the hallways. Messages such as "I am about to move, please, stand clear" and "My way is blocked. Please, move the obstacle" have been very important in gaining acceptance and support for the robot.

Many people try to talk to the robot. In the future, voice recognition and improved voice generation will provide a dramatic improvement in sociability.

5 Robotic Transport: System Requirements

A single HelpMate robot, on its own, despite the powerful navigation and obstacle avoidance algorithms, is inadequately equipped to function in a real world hospital environment. The HelpMate in itself is unable to call and ride on elevators and open doors without the help of the companion system products and technologies developed at HRI and integrated into our robot transport systems. A single robot transport system requires elevator controllers, door openers and annunciators, and communications between the robot and each one of these devices.

HelpMate is unique in its ability to call and ride elevators to travel between floors and buildings. The first installation of HelpMate at Danbury hospital in 1988 used infrared transmitter/receiver pairs mounted on the robot and in the elevator and in the ceiling of every elevator lobby. Using this infrared link, the HelpMate communicated with the central controller, called the elevator to its floor, entered the elevator while holding the door open, then closed the door and made a floor call to the desired floor. Once at the requested floor, HelpMate navigated out of the elevator, closed the door, and signed off with the elevator controller.

Although the algorithm proved robust and is still in use with enhancements, communications via the infrared sensor proved unreliable, and required too accurate a positioning in the elevator lobbies. The usual crowd in the elevator lobbies often made it impossible to be positioned within communication range of the infrared sensor.

Experimentation with radio communications began in 1991, and that is our current communication mode. Our elevator controller has transitioned from being an Allen Bradley programmable logic control (PLC) system to a standard single board personal computer running elevator control software written in C++ under MS DOS.

Door openers and annunciators (a light and chime to signal someone on the far side of a secure door that HelpMate is just outside), also originally employed infrared as the communication mechanism, and they have proved to be quite adequate and robust. A wide angle transmitter on the robot using multiple led sources broadcasts a signal with a very simple encoding to distinguish the robot from other IR sources in the hospital. This is very much like a remote control for a television set. The demand for installation cost reduction has forced us to look at radio communications for these devices, and radio annunciators will soon be offered as an option.

HelpMate robots use the Arlan series of radios, offered by Aironet, a subsidiary of Telxon, providing RF spread -spectrum communications. The initial implementation used the 100 series radios which provide point-to-point serial communications at 9600 baud. The newer radios offered by Aironet are the 600 series that facilitate ethernet communications with an effective throughput of several hundred kilobaud. Lucent, Symbol and Proxim are also now offering spread spectrum wireless communications for hospitals, and we are working to be compatible with any and all such systems.

Incorporation of the 600 series radios into the HelpMate+ required integrating a third party TCP/IP stack with the Beta versions of the Arlan 600 series. This proved to be far more complex than anticipated, and hardware and software modifications were necessary to the HelpMate robot and the elevator controller software to benefit from this new technology.

The adoption of radio communications in the year 1992, albeit serial at 9600 baud, paved the way for the concept and development of other cooperating system devices such as the Supervisory controller and a Robot Monitor. It was during 1992 that the first multiple robot sites were operational at Danbury and at Stanford University Medical Center.

Fleet control and systems issues, addressed in subsequent sections, have been the focus of our efforts since 1992.

6 Evolution in System Software

Over the past decade the HelpMate internals and user interface have undergone many changes and improvements. Understandably, our initial concern was to prove that the robot could indeed navigate and avoid obstacles in real world environments. Once that was accomplished the focus turned to improving the installations and making the robot easier to use and to service in the field.

6.1 Initial Implementation: A Traditional Application Programming Language

The primitive behavior element for HelpMate is getting down a single hallway without running into things. A language was developed for specifying the navigation of a sequence of hallways to reach a destination. The language was named HAL, for HelpMate Application Language, and consisted of about 5 keywords with arguments to carry out the functions of navigating and turning in halls along with velocity and acceleration settings. The initial versions of the HelpMate software included a simple kernel, a file manager and a program executer to parse, interpret and execute program steps.

HAL was enhanced further to include statements that assisted in the operation of elevators, automatic doors and annunciators. Our field trials consisted mostly of keying in the programs by hand, downloading them to the robot and having them executed on the robot. A typical user installation could have hundreds of such programs to enable the robot to travel between several departments and all of the nursing stations as destinations.

While experimenting with sensors, algorithms and parameters in the early days it was essential that we not waste valuable time going through the cycle of compilation, linking and downloading to try an idea out. We overcame this by creating a menu driven user interface for the engineers via which we could change the application parameters. Using this interface we could change parameters like the composite map decay rate, the maximum drive deceleration, the sonar firing sequence, or even add a new sonar sensor anywhere on the robot. This facilitated quick parameter modifications while experimenting with new sensors or algorithms.

The application parameter interface is hardly being used today, even by the installation engineers, and any parameters still in use are being incorporated into the topography rule file described below. This then guarantees that all application specific information resides in a single repository. Having this interface available in the early days was invaluable, however.

6.2 Development of the Topography Data Base: Data and Rule Driven Behavior

Installation of HelpMates in hospitals required that we come up with an automated way of generating these programs rather than requiring programs for each route. We have accomplished this by enhancing a commercial CAD program with a custom shell specific to our needs and having our installation engineers use this tool to input hospital topography information. In addition to the layout or topography provided, the installation engineer is free to add station names and impose speed or route specific rules on either the entire drawing or on a few halls. Locations of doors that need to be opened or annunciators that need to be triggered are also included. Stations are nodes, or locations, where the robot waits to be serviced.

This rule file and the drawing files generated by the AUTOCAD program are interpreted and converted to a file known as the Topography by an application called TopGen. The Topography file is generated both as an ASCII file for human review and as a binary file for downloading to the robot. Note that this is the only site specific information used by the HelpMate to generate the menu driven user interface tailored to the site, to generate the routes to travel between stations, and to modify the behavior of the robot in a way specific to that particular site. Figure 4 shows the sequence of steps in building the Topography.

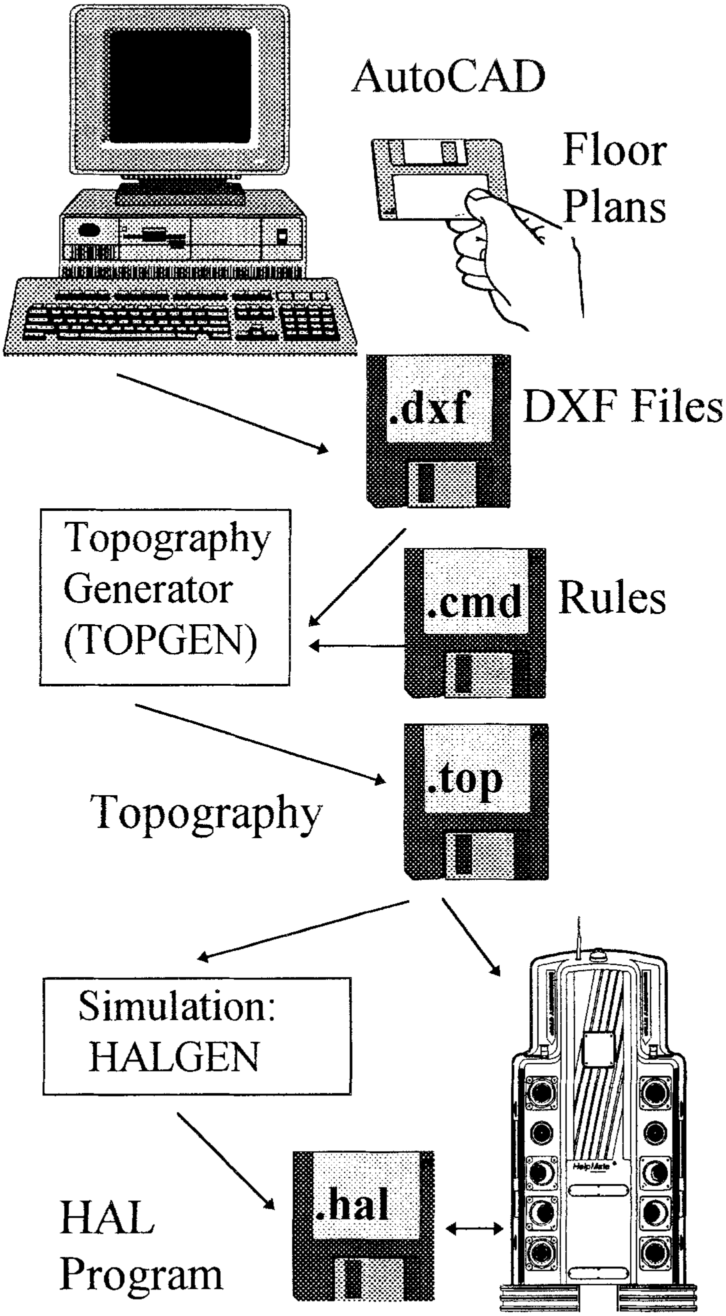


Figure 4: Topography Development

6.3 User Interface

The man-machine interface uses a menu display, with the user making selections using a numeric keypad. On dispatch using the menus the HelpMate is able to plan a route to the station specified. This is done by examining all the possible paths from the current station to the destination and picking the path with the smallest weight. After optimization, path generation in a very large installation such as Baylor University Medical Center with 55 stations in 5 buildings with up to 19 floors with 5 elevators and 2.8 km of halls takes 13 seconds on a 68332 processor. This is an extreme case, because the floors are mostly linked between elevators and hence the number of potential paths is very large. Modest facilities without linkages between elevators take only a second or less to compute. The output of the path generation module is a HAL program such as the one described above.

The path generation module can be run on a desktop PC for debugging and exhaustive route checking prior to installation. This application is called HALGEN. It takes as input a Topography file and a script file indicating the programs to be generated, and creates an ASCII HAL program that can be viewed or even downloaded to the HelpMate for execution.

Typical installation time for a 400 bed hospital takes of the order of 15 man-days, of which about 5 man days are in creating the Topography and 5 man-days are spent on site in testing and training. This is a manageable amount of time, thanks to the installation tools provided to the engineer.

6.5 Debugging and Management Tools: Flight Recorders, Run Logs and Diagnostics

It soon became apparent that we needed to concentrate on debugging aids for run time errors and apparently inexplicable situations. We created a HelpMate 'flight recorder' that saved all data, events and decisions made by the master processor. This was essentially a large data store of sensor data received with time stamps, the effect of this data on the composite local map, the pathways detected, the one chosen, and the final drive commands sent out for each tick. Memory restrictions only facilitated storage of this cyclic detailed log for about 5 minutes. So, as long as we got to the robot immediately after the situation happened, we could upload the log, look at the ASCII data, and decipher the problem.

Perusing the flight recorder in ASCII format proved too tedious. A tool we called ROMON (Robot Monitor) was developed to depict the data in graphical format. Figure 3 above shows the output of ROMON. This application runs on our development PC's, takes as input the flight recorder data and displays a pictorial image of the robot, its environment and the navigation decisions. ROMON helps visualize the problem situation and pinpoint the area in the flight recorder that needs more careful analysis.

Flight recorders are rarely taken by the installation engineers now, except under rare occasions. Automatic methods of saving flight recorders have been developed. Future enhancements call for the flight recorders to be collected and archived automatically either on the hard drive or sent to a server on the network for evaluation.

As robots moved into commercial use, run logs were added to track application data. These logs are kept permanently in the robot's memory. The data collected can be classified as either diagnostics or trip related data. The diagnostics data log captures data every time the power up diagnostics fail, along with detailed failure information and a time stamp. Other diagnostics logs indicate how often, for example, the ceiling IR sensors have failed, the vision subsystem has failed, or an encoder error has caused trip cancellation. The trip log dutifully records the number of times the HelpMate was dispatched to each station and the number of times it was successful. This information along with the number of hours it has been in use provides valuable round trip time information to provide data on justifying cost savings for the customer.

Layers of diagnostics were developed to aid in the diagnosing of hardware and sensor related problems. Initially raw sensor data was viewed using a primitive menu interface. This was refined in a separate menu-driven diagnostics package that allowed the sophisticated user to issue any command or sequences of commands to any of the subsystems from the HelpMate master processor, the 68000. This layer was enhanced by adding a graphical view of the robot along with the raw sensor data. During initial development of the HelpMate, these tools were critical for engineers working in the field trying to understand the behavior of the robot, they are now used occasionally by manufacturing and by field service.

Current HelpMates run power-up diagnostics that immediately diagnose and report any problems that may adversely affect the performance of the robot. This power-up diagnostics checksums the system software and the Topography, and runs tests on every sensor in the system. Failures are reported to the user with information on the failure mode. Drive and sensor failures are pinpointed and reported back to the hospital staff who are able to relay the information to HRI field service engineers and greatly assist in speedy problem resolution. These start-up diagnostics insure safe operation of the robot.

7 Fleet Management

Effective fleet management requires three functions. First, traffic control algorithms to ensure efficient usage of systems resources; second, tools to monitor the real time performance of the robots; and, last, data collection tools to record and analyze the effective throughput of the system.

7.1 Traffic Control

Traffic control is indispensable for smooth and robust operation of multiple robots cooperating on delivery tasks with overlapping paths. Hospital hallways are typically cluttered with meal carts, stretchers, IV posts, laundry carts and wheelchairs in addition to being high traffic areas which can be crowded with staff and visitors. These conditions make it virtually impossible for HelpMates to work together in the same hallway without unduly delaying each other or appearing to act quite stupidly by blocking each other's way, necessitating intervention by people or computer controlled supervision.

Contention for elevators is a similar problem. Elevator resources, especially in hospitals, represent the lifeline of the healthcare delivery system and must therefore be managed carefully. Since elevators are scarce resources, situations that call for awkward multiple robot maneuvers and confrontations must be avoided at all costs. To add to the above, elevator lobbies in hospitals are notorious for being cluttered with equipment, staff, patients and visitors. Two HelpMates maneuvering in the elevator lobby vying for a common resource results in unnecessary frustrations and delays for the impatient hospital staff and patients.

Peer to peer communications between robots to resolve such conflicts were explored. Two robots could conceivably disentangle themselves from tricky situations, using local robot to robot communication and deadlock dissolution algorithms. We noted that the complexity of the algorithms increased rapidly in order to prevent time consuming and inconsistent behavior when more than 2 robots were involved.

The railroad industry, with similar conflicts, adopted a simple rule: any single section of track was a one way zone that could be used by only one train at a time. This idea is called zone control. When two trains must pass or otherwise be in the same zone, sidings or spurs are provided. A siding is simply a detour off the main track which converges with the main tracks some distance away. It exists solely for a train to pull over and wait until after another train has safely passed by. A spur is a dead end section of track branching off the main line.

Industrial automatic guided vehicles (AGVs), materials transport carts that follow fixed routes around a factory or warehouse, adopted the basic idea of zone control from the railroads. They refined the possibilities of traffic control with bumper blocking, in-floor zone blocking or computer zone blocking mechanisms [44,45]. Bumper blocking is used when the vehicles are travelling along the same direction, following each other. In-floor zone blocking and computer zone blocking require physically dividing the guidepath into logical zones and controlling access to them. Physical site modifications are a part of the AGV system installation, and therefore pose no problem. Asking HelpMate application engineers to install signals or traffic lights at intersections is an additional time and cost burden to be avoided if possible.

HelpMate transport systems deploying more than one robot require the functionality of a central traffic control computer, called the Robot Supervisor, with radio network communications to all other system component devices. An IBM PC compatible computer running DOS, MS Windows, Win95 or Windows NT, located in a secure area, functions as a traffic control by regulating resource allocations and ensuring system integrity.

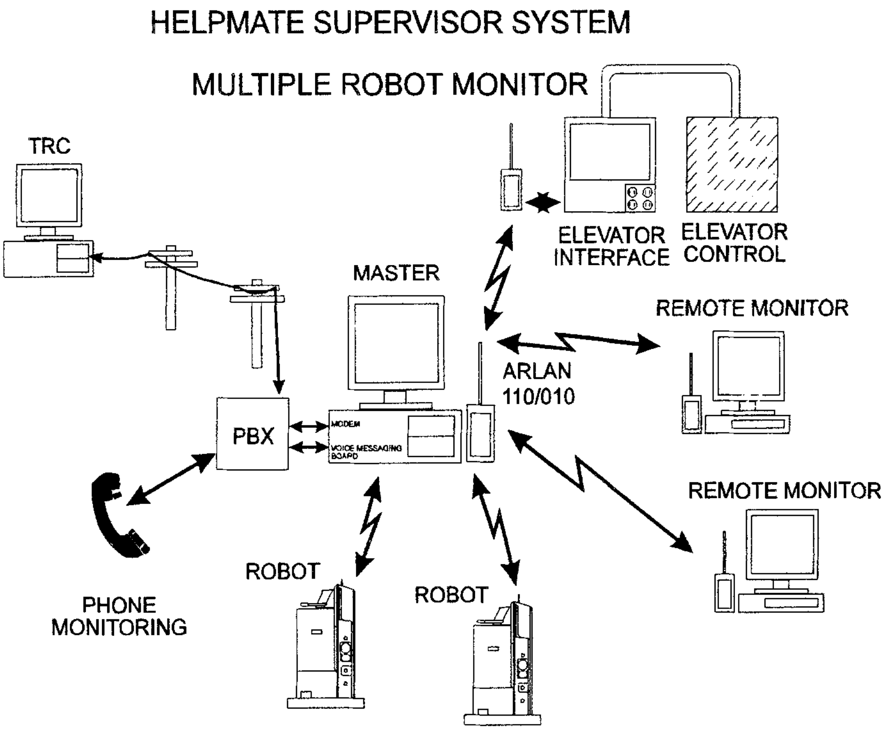


Figure 5: Multiple Robot Supervisory Control System

A combination of spurs and a variation on the AGV idea of computer zone blocking is used in multiple robot traffic control. Coordination of multiple robots is based on the premise that the site topography can be are divided into distinct areas and access to those areas is only available on demand to individual robots. Additional halls, akin to spurs and used solely for deadlock resolution, are added into the layout of the hospital at the discretion of the installation engineer.

A deadlock situation occurs when two or more HelpMates prevent each other from completing their missions. The supervisor algorithm detects deadlocks once the robots are stopped at a junction zone and a robot is requesting resources that have already been granted to another.

Deadlock resolution is dependent on there being a layaway node nearby to which to re-route one of the robots. The resource granting algorithm ensures that at least one layaway is available for such deadlock dissolution for each junction zone, thus preventing, for example, three robots arriving at a three way intersection and occupying all three layaways. The resolution takes place by sending a re-route request via a nearby layaway to one of the robots. This causes the roboi to re-route itself to the layaway and await re-entry until the other robots clear the requested zone. It will then be able to continue on to its final destination.

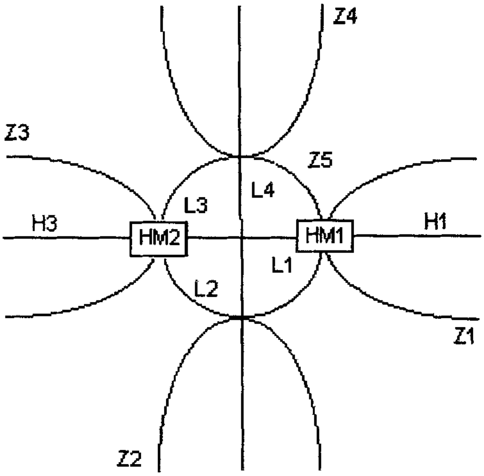


Figure 6: Zone management and deadlock resolution

In the above figure, HelpMate HM1 wants to navigate from hallway H1 to hallway H3 while HelpMate HM2 is going in exactly the opposite direction. Upon arriving at layaways L1 and L3, they both request control of zone Z5 and the layaways across the zone, each of which has already been granted to the other robot. This creates a deadlock. The supervisor will reroute the first robot to arrive to one of the open layaways at the intersection, L2 and L4. This will free one of the robots to proceed, and subsequently the other robot will return to its desired path.

7.2 Real Time Robot Monitor

The Robot Monitor application was developed to provide the end user with tools to monitor the status of robots, with the eventual goal of providing central dispatching and rerouting. In addition to providing status on all robots, the application is able to alert the user to any emergency situations. This application is written in C++, runs on an IBM PC under Windows or Windows NT and uses the Topography files supplied by the installation engineer to display site specific information.

Robot status, location and mission information are of prime importance in fleet management. The user interface is composed of dynamically updated windows displaying selected robot and elevator information. The main window contains controls that show the HelpMate's current mission, its current location and destinations, the current battery voltage, the current time, and a log of the past 100 navigation events.

Event text and action may be customized by the installation engineer to suit customer requirements. Action options available are entering the event into the event log, speaking a message or dialing a beeper number.

The application is capable of tracking a specific robot as it moves around in the hospital. The topography window shows the layout of an installation floor and the location of all HelpMates within the area of that window. When robot tracking is requested the topography window is panned as the robot moves out of context.

7.3 Data Collection and Analysis

The HelpMate Monitor generates an ASCII log file of important events. This information is useful in determining, for example, how many deliveries were made or for what percentage of time the elevators were being used. The format of the file is set up so the data file can be imported into Lotus, Excel, Access, dBase, or similar programs to generate detailed reports.

The largest current installation as of early 1997 is four robots in one hospital. This will rise to 7 robots by the end of 1997 and eventually we expect to see large fleets of robots in use.

8 The Future

After a decade of hard work, running close to a man-century of effort, we now have well over 100 robots running around the world, many working 24 hours per day, 7 days per week, in uncontrolled and unsupervised environments.

HelpMate provides simple courier services, requiring a hospital worker to load the robot and tell it where to go and another hospital worker to unload the robot at the destination point. Loads to about 1 m^3 and 100 kg can be carried by the robot. At some point automated loading and unloading will be provided with arms or special purpose mechanisms on the robot.

Eventually we expect to be selling fleets of robots to hospitals. Those fleets will include machines of different shapes and sizes to provide all manner of materials transport duties, including movement of the large dietary, laundry, supply and trash carts, movement of specimens within clinical laboratories, and movement of mail in the offices.

Patient transport may be technically possible, but most hospitals believe that transporters provide information and comfort that are uniquely human capabilities. However, in certain circumstances patient transport will be accomplished in the future with robot assistance.

The thrust of our future work is then toward larger robots, smaller robots, fleet management and automated loading and unloading. Cost reduction, reliability improvement, and the addition of features to solve specific customer problems will continue to be the focus of our efforts on the current machine.

The basic HelpMate robot will not change very much. New generations of machines will have improved sensors such as lidar sensors to image the ambient environment and will have improved controls for faster and smoother behavior.

More significant will be the evolution of the system software. We have moved from a traditional sequential programming language to data and rule driven behavior. In the future we will have intelligent and self optimizing systems that learn the environment on their own, optimally tune their behavior, diagnose themselves and provide remote diagnostic and service reports. Figure 7 shows this evolution.

By far the most interesting capability for the future will be voice recognition and speech generation. People already name their robots and visitors and staff talk to them. When they can talk back, even at the most primitive level, they will be considered to be far more intelligent and capable to all who interact with them.

	Behavior Specification	**Application Setup**	**Diagnostics**
Generation 1 (Development)	Procedural Language: HAL	Application Parmeters via Engineering Menu	Engineering Menu
Generation 2 (Current)	Data and Rules: Topography	Application Parameters via Topography	Maintenance Menu, Start-Up Diagnostics
Generation 3 (Future)	Self-Optimizing Auto Mapping of Environment	Self-Tuned	Remote and Self Diagnostics

Figure 7: Evolution Of HelpMate System Software

References

[1] Evans, J., Krishnamurthy, B, et.al. “Creating Smart Robots for Hospitals”, Proc., Robots 13, Washington, DC, 1989

[2] Evans, J., Krishnamurthy, B., et.al. “HelpMate: A Robotic Materials Transport System.” Robotics and Autonomous Systems, 5, 251. 1989.

[3] Robertson, Gordon I., “HelpMate™ Delivery Robot Operates Safely Amongst the General Public.” Proceedings, 22nd ISIR, SME, Detroit, Michigan, 1991.

[4] Engelberger, J.F., “Health-care robotics goes commercial: the HelpMate.” Robitica, **11**, 517. 1993.

[5] Evans, J., “HelpMate: An Autonomous Mobile Robot Courier for Hospitals.” Proceedings, IROS 94.

[6] Marr, David, Vision. New York, W.H. Freeman, 1982.

[7] Brady, Michael, et.al., Robot Motion: Planning and Control. Cambridge, MIT Press, 1983.

[8] Tilove, R. B., 1990, "Local Obstacle Avoidance for Mobile Robots Based on the Method of Artificial Potentials." 1990 IEEE International Conference on Robotics and Automation (ICRA90), Cincinnati, Ohio, May 13-18, pp. 566-571.

[9] Kanayama and Miyake, "Trajectory Generation for Mobile Robots", 3rd International Symposium on Robotics Research, ISRR-3, Paris, October, 1985.

[10] Barraquand, J., and Latombe, Jean-Claude, “Nonholonomic Multibody Mobile Robots: Controllability and Motin Planning in the Presence of Obstacles.” Proc. 1991 IEEE ICRA, Sacramento, CA, April, 1991, p2328.

[11] Nilsson, N.J., “A Mobile Automaton: An Application of Artificial Intelligence Techniques.” Proceedings, IJCAI-1, 1969.

[12] Briot, M. et. al., “The Multi-Sensors Which Help A Mobile Robot Find Its Place” Sensor Review: 15-19, Jan. 1981

[13] Giralt, G., Chatila, R., and Vaisset, M., “An Integrated Navigation and Motion Control System for Autonomous Multisensory Mobile Robots.” Proc., First Int. Symposium on Robotics Research, ISRR-1, Breton Woods, NH, August 28, 1983.

[14] Flynn, Anita M., “Redundant Sensors for Mobile Robot Navigation.” MIT AI Lab Technical Report AI-TR-859, 1985.

[15] Biber, C., et. al., “The Polaroid Ultrasonic Ranging System.” Audio Engineering Society 67th Convention, New York, 1980. Available from Polaroid.

[16] Kadanoff, M.B., "Navigation Techniques for the Denning Sentry." Proceedings SME Robotics Research Conference, Scottsdale, AZ, Aug 18, 1986.

[17] Moravec, Hans P. and Elfes, Alberto, "High Resolution Maps from Wide Angle Sonar." Proceedings, 1985 IEEE ICRA, St. Louis, MO, March 1985.

[18] Drumheller, M., "Mobile Robot Localization Using Sonar." AI Memo 826, MIT AI Lab, January 1985.

[19] Leonard, J., "Directed Sonar Sensing for Mobile Robot Navigation." Ph.D. Thesis, Oxford Department of Engineering Science, 1990.

[20] Holland, John., "CA-2 Ultrasonic Collision Avoidance System., Preliminary Specifications," Cybermotion, Inc., Roanoke, VA, Oct. 1989.

[21] Kay, Leslie, "Transducers." U.S. Patent 4,704,556. Inventor of Sonic Glasses for the blind ca. 1973.

[22] Kuc, Roman and Viard, V.B., "Guiding Vehicles with Sonar: The Edge Problem." Proceedings, IEEE 1988 Ultrasonics Symposium, Chicago, IL, 1988.

[23] Coles, S.L., Raphael, B., Duda, R., et. al., "Application of Intelligent Automata to Reconnaissance." Stanford Research Institute Technical Report, November, 1969.

[24] Moravec, H.P., "The Stanford Cart and CMU Rover." Proc. IEEE, **71**, 7, p872, July, 1983. Also CMU Tech Report, same title, 1983.

[25] Thorpe, C., et.al., "Vision and Nivigation for the Carnegie-Mellon Navlab." PAMI, **10**, 3. IEEE, May, 1988.

[26] Low cost vision systems have been developed at USCand MIT, for example.

[27] King, Steven J., "HelpMate: An Autonomous Mobile Robotic Transport System." Proc. Electronic Imaging East 90. BISCAP, Boston, Oct. 29, 1990.

[28] Evans, J.M., Weiman, C.F.R.W., and King, S.J., "Mobile Robot Navigation Employing Ceiling Light Fixtures" , U.S. Patent 4,933,864.

[29] Evans, Weiman, and King, "Visual Navigation and Obstacle Avoidance Structured Light System." U.S. Patents 4,954,962 and 5,040,116.

[30] Borenstein, J., Everett, H.R., and Feng, L., "Where am I?" Sensors and Methods for Mobile Robot Positioning. University of Michigan Technical Report UM-MEAM-94-21.

[31] Kanayama, Y., et.al., "A sonic range finding module for mobile robots." Proceedings, 14th ISIR, Gothenburg, Sweden, October, 1984.

[32] Kak, A., et.al., "Hierarchical Evidence Accumulation in the PSEIKI System and Experiments in Model Driven Mobile Robot Navigation." In Uncertainty in Artificial Intelligence, **5**, Elsevier Science Publishers B.V., North-Holland, p 353.

[33] Crowley, J.L., "Navigation for an Intelligent Mobile Robot." IEEE Journal of Robots and Automation, **1**,1, p31, 1985.

[34] Cox, I.J., "Blanche-An Experiment in Guidance and Navigation of an Autonomous Mobile Robot." IEEE Transactions Robotics and Automation, **7**, 3, p193. 1991.

[35] Holland, John, "Sensor Fusion"; and Blachman, S., et.al., "A Fuzzy Logic Certainty Engine for Evaluating Environmental Threats with Mobile Robots." Proceedings, Sensors Expo, Sensors Magazine, Chicago,1992.

[36] Brooks, R.A.,and Connel, J.H., "Asynchronous distributed control system for a mobile robot." Proceedings, SPIE, **727,** Mobile Robots, October, 1986.

[37] Albus, J.S., et.al, "Theory and Practice of Hierarchical Control." Proc. 23d IEEE Computer Society International Conference, September, 1981.

[38] Borenstein, J., and Koren, Y., "Real Time Obstacle Avoidance for Fast Mobile Robots in Cluttered Environments." IEEE 1990 ICRA CH2876-1. Cincinnati, OH, p572., May 1990.

[39] Lozano-Perez, T., and Wesley, M., "An algorithm for planning collision-free paths among polyhedral obstalces." Communications of the ACM, **22**, 10, 560. 1979.

[40] Crowley, James L., Private Communication, 1991.

[41] Cussler, Clive, Dragon. New York, Simon & Schuster, 1990.

[42] US Regulation- FCC Part 15 requirements for unintentional and intentional radiators.

[43] EC Regulations- EN55011, EN50081-2, EN50082-2, IEC1000-4-1, and IEC1000-4-3

[44] Castleberry, Guy A. The AGV Handbook, 1991, Braun-Brumfield, Ann Arbor, MI, USA.

[45] Miller, Richard K. Automated Guided Vehicles and Automated Manufacturing, 1987, Society of Manufacturing Engineers, Dearborn, MI, USA.

Intelligent Wheelchairs and Assistant Robots

Josep Amat
Institut de Robòtica industrial (IRI)- CSIC/UPC
Barcelona (SPAIN)
Amat@esaii.upc.es

Abstract : This work presents an overview over the main technological aids oriented to the rehabilitation of the physically disabled so that they can get some independence. These aids range from wheelchairs up to the assistant robots developed in the last years

1. Introduction.

Technological developments in the last years have not only allowed an increase of the automation level in industry, they have also enabled the manufacture of many consumer goods. Such goods considered by society as basic elements to improve the quality of life, are equipments such as automobiles, audio and video devices, electric appliances or even personal computers.

Many of these equipments, direct or adequately modified can also constitute valuable aiding elements to people with physical disabilities produced by diseases, accidents or also to persons with physical limitations due to old age.

The current availability of equipments that constitute today valuable aids for disabled people, come from an increasing number of specialized companies that respond to the needs of a demand more and more important. On the other side, these companies have available an increasing range of components and devices originally addressed to industrial automation, but that, conveniently adapted, also allow the automation of lighting systems, ventilation, and access to or manipulation of the most common home elements. These elements start to be used by persons that suffer from important deficiencies in their motor capabilities [1].

The companies specialized in providing technological equipment to physically disabled persons to increase their independence, offer everyday new technological resources, ranging from new materials to robotics. These companies also rely on the support of many research centers, making possible with this cooperation to significantly increase the market supply of this kind of aids and to appreciably reduce their cost. Thus, more potential users can reach these products, fig. 1.

2. Technological aids for mobility

One of the most usual disabilities of old people is their lack of mobility. This disability make them unable not only to move outside, in public thoroughfare, but even to carry out the necessary short runs at home.

The use of wheelchairs, that started to be developed at the beginning of this century, has enabled to provide a very efficient solution to the need of mobility, without any other external aid, provided the impaired person can use his or her upper limbs.

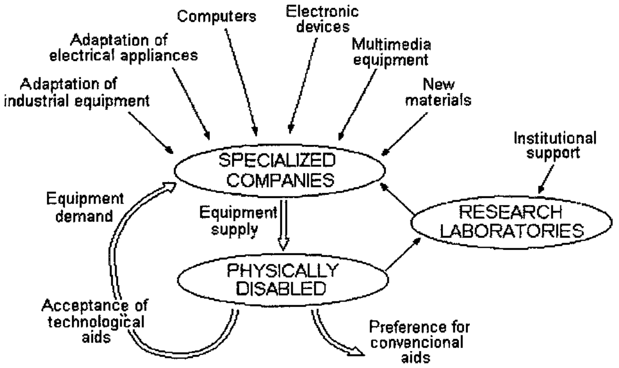


Fig. 1 Impact of technological advances in the supply of aids for disabled.

The use of wheelchairs has expanded enormously in the second half of this century, but only recently, in the last years, they have started to be motorized. Consequently, they also can be used by people with motor impairment in their upper limbs, severely enough to impede their manual propulsion. It is specially the development of technologies tied to automation and even those developed for mobile robots, what have allowed to build wheelchairs endowed with a control unit to facilitate its driving. With these more advanced wheelchairs, even people with very low remaining motion in their hands can drive them without any external aid.

Nowadays, motorized wheelchairs can be driven by means of intelligent controllers. These are microcomputer based controllers enabling the chair to carry out straight line trajectories independently of terrain irregularities, to change direction incrementally or to rotate in place, fig.2. Each user needs his own adapted interfaces, according to his deficiency, to control the wheelchair [2]. For instance, these interfaces could be adapted to detect the head movements when the user can not use his hands.

With the aim to control the trajectories with the support of a microcomputer, these wheelchairs are endowed with angular encoders in their steering wheels. The encoders supply the data required to measure runs of the order of millimeters, and consequently it is possible to calculate, at every moment, the wheelchair trajectory and adjust it to the set points given by the user. This procedure allows to detect the wheelchair deviations produced by the terrain unevenness or its irregularities, and to compensate them. In the same way, it allows to memorize trajectories and to perform automatically their inverse runs. This facility allows to go out from constrained spaces, avoiding the great amount of maneuvers that would be necessary driving the chair manually. This is the case, for instance, of leaving a narrow bathroom or an elevator.

These intelligent wheelchairs are still little accepted by users, but not due to their cost, since the additional cost of electronics with respect to the cost of the motorized wheelchair is not very relevant. Acceptability is limited by a technological factor, that still predispose negatively most of the potential users.

Such intelligent controllers can also benefit from other advances reached in the field of autonomous navigation or mobile robots [3]. For instance, endowing the wheelchair with absolute positioning systems, GPS type, or with obstacle detection systems. Such systems enable to have available motorized wheelchairs with aids for navigation in urban environments, even for the severely disabled.

The efforts carried out with the aim to endow wheelchairs with the possibility to go up and down stairs has achieved, technically speaking, positive results, fig. 3. But, their high cost and their bulkiness, together with the availability of other existing aids to go up vertical unevenness, has had as a consequence that these kind of wheelchairs are scarcely used.

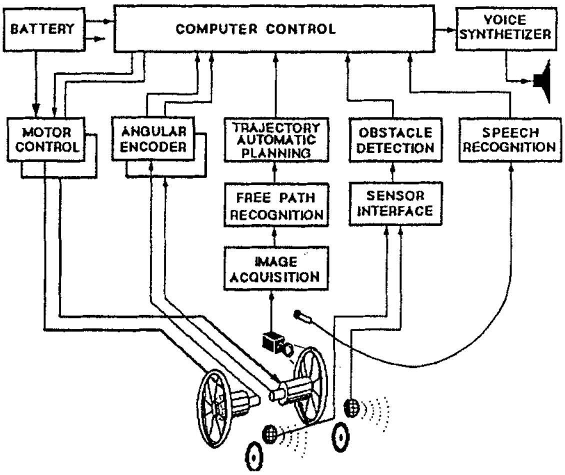


Fig. 2 Control structure of an intelligent wheelchair

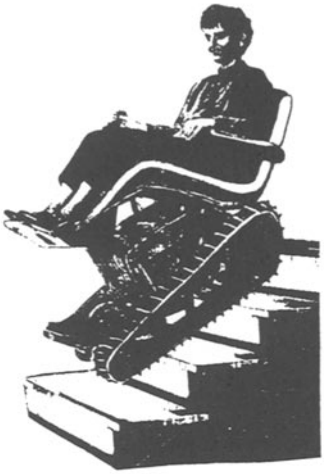


Fig. 3 Wheelchair with caterpillar to up and down stairs

On the other hand, some efforts have been done in the design of new chairs and the study of the materials to use, to build wheelchairs adequate to practice some kind of sports or physical activities. Nowadays, with very light wheelchairs having an ergonomic design, it is possible to practice athletics, basketball, cycling or even to fly with delta wings, fig. 4.

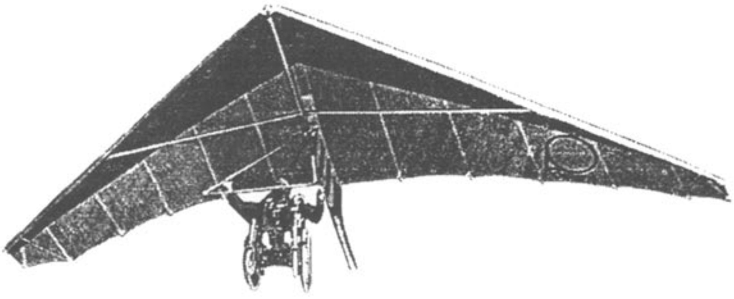


Fig. 4 Possibility for a disabled to practice sports

3. Prosthetic elements

Robotics advances are also applicable to the development of aids oriented to the rehabilitation of people with muscular atrophy or amputees, which, due to this impairment have missed the mobility of one or more of their limbs.

The development of arms and legs, either prosthesis or orthesis, started in the seventies and has already attained remarkable performances. With current technology it is possible to build arms or legs with a physical appearance, totally mimetic to human's. This fact makes them, from this point of view, totally acceptable. A technological difficulty, presently, comes from the power supply requirements, specially for lower limbs which consumption is far higher. On the other side, another difficulty is due to their control, according to the user's will, every moment of his daily life.

When the physical deficiency comes from a recent amputation or even from a muscular atrophy, it can be possible to control the joints of a prosthetic arm using the user own myoelectric signals, those generated by the brain to activate the muscles in healthy people [4]. In this case the myoelectric signals are acquired by means of electrodes, are amplified and afterward processed to identify the kind of movement desired by the user. Fig. 5.

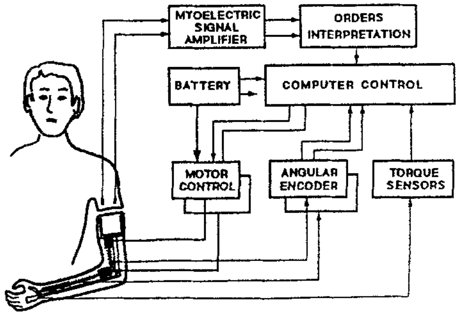


Fig. 5 Schema of the control system of a prosthesis by means of myoelectric signals

Some of the problems not still solved are the capabilities to differentiate signals from noise, and to interpret many orders given by the brain. Currently, only the most significant orders are interpretable, orders such as up and down, approach and retrieve the arm or open and close the hand. Imprecision in the interpretation of these orders forces the user of such devices to initiate a process of learning his new capabilities, which though they are very few with respect to a healthy arm, enable the user to recover an important level of his autonomy.

When the physical impairment affects also the generation of the myoelectric signal, being either due to an injury in the spinal cord or in the brain itself, it is necessary to foresee an adapted interface. This interface activated by the user's remaining movements (head, face or mouth movements) constitutes the means of communication with the computer in order to control the movements of the arm until the desired actuation is attained.

Apart of the situations in which the motorized prosthesis are indispensable, as in the case of amputees of the two arms, their acceptability is still very low, even considering the degree of perfection attained in their aesthetics. The Waseda hand, built at Waseda University [5] in 1985 offers an excellent similarity to the human skin, fig. 6, but the servitude that the weight of its power supply represents, and the noise produced by its motors to execute the hand movements are important problems. For these reasons, in many cases, orthopedic passive arms are preferred. The geometry of such arms is configurable by the user, which utilizes this orthopedic arm as a complement to his healthy arm. He learns to modify the position of the artificial arm with almost imperceptible movements of his arm or body in order to reach the best position of the prosthesis for each task to carry out.

The obtention of satisfactory use of robotized arm prosthesis, makes it necessary to still perform research efforts in the design of low size actuators, more efficient and noiseless, as well as in the development of more intelligent control systems.

In this line some other prototypes have been developed. Among them we can mention the "Sams" hand (Southampton Adaptive Manipulation Scheme) developed in 1994. This hand controls independently the movements of the forefinger and the thumb from the other three fingers. This increase of degrees of freedom provides a higher versatility of the hand movements. This greater versatility carries with it a hierarchical control structure in which every joint has available a specific controller that receives also information from the force and sliding sensors available in the hand. These controllers are coordinated by a higher level controller that executes and supervises the actuation programmed by the user, from high level orders that enable to attain a more intelligent control .

Another procedure followed to recover the movement of upper limbs in case of muscular atrophy, in which the motor capability is lessened in such a way that the muscles can not even hold the weight of the own arm, consists in the use of an exoskeleton. In this case, the control of the joints from the endowed force sensors, enable to compensate the weight of the arms and to utilize the user's remaining movements to attain the exoskeleton desired movements.

When the degree of atrophy does not allow to detect the movements the user wishes to perform, it is necessary to use an adequate interface, operating either from orders given through the head movements or orally, using a voice recognition system.

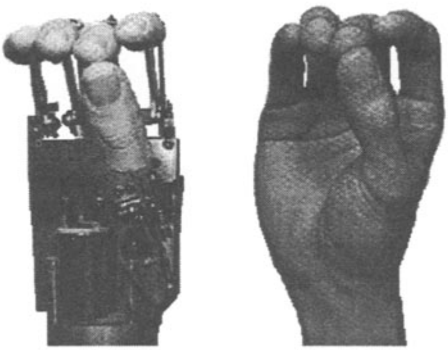


Fig. 6 Prosthetic hand with human appearance built at Waseda University

Fig. 7 Mobility attained using orthopedic legs

The development of lower limb prosthesis presents the additional drawbacks of requiring a power supply of higher capacity, and the fact of making the generated locomotion movements compatible with the body balance. Some prototypes have already been designed, able to execute a sequence of movements with an orthopedic leg, from the movements given by the not injured leg while walking over a flat soil, or even climbing or descending stairs.

In the same way, the coordination of two orthopedic legs has also been attained. The generation of a sequence of movements enabling the user to walk or to run, fig. 7, has also been proved [6]. Nevertheless, this kind of prosthesis have still more difficulties to reach its acceptability, in front of the solutions based on the use of wheelchairs.

4. Assistant robots

The possibility to rely on the use of robots to aid a disabled to get certain independence, even to severely disabled such as tetraplegic, was considered in the eighties. The Veterans Administration Medical Center had available already in 1986 an advanced assistant robotic prototype.

The goal of such robots is to replace the lack of motion capability of an impaired person to be able to approach and to manipulate objects in his environment, without the need to continuously rely on an assistant. And even, to be able to perform autonomously a certain number of daily life activities, such as eating grooming or toileting.

Basically, three different kind of assistant robots can be considered: those mounted over the user own wheelchair; those installed fixed close to the user, and those installed over a mobile base, to be able to move within a limited environment.

4.1 Assistant robots mounted on the wheelchair

The advantage of having a robotic arm mounted on the own wheelchair is that the user can move freely and use his auxiliary arm to manipulate objects at any place in his home. But, on the other hand, this option has the drawback of requiring to always transport this device with its significant volume and weight. Fact that sometimes can limit the user's accessibility.

Probably, among all the robots installable on wheelchairs, the one that has attained a certain acceptance is "Manus" developed by TNO in the Netherlands, from 1981 to 1990, in the frame of a European project within the TIDE program [7]. The robot was commercialized in 1991. This robot, fig.8, has seven degrees of freedom, and has been designed so as to attain a great accessible area, being even able to get objects from the floor. Its cylindrical structure with a telescopic base is actuated by means of electrical actuators. The arm can be folded so as to minimize the occupied space when not in use. The control is performed by means of a joystick and with the aid of a simplified functions keyboard its use is much simpler, even performing relatively complex tasks.

Its acceptance and diffusion in Europe, as well as in other countries around the world, has enabled to organize a user's group to facilitate the interchange of experiences among users with different remaining motions. The gained experience allows to solve some of the problems detected and to improve its performances.

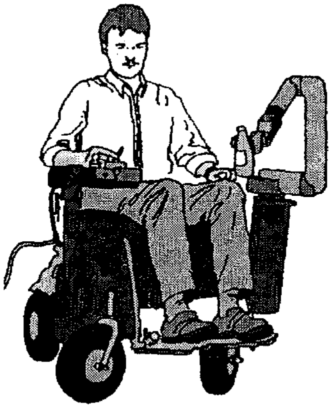


Fig.8 The Manus *assistant robot.*

Another prototype of robotic arm mounted over a wheelchair is "Inventaid" developed by the Papworth group in 1992. This robot has six degrees of freedom and is moved by means of pneumatic actuators. Its control system is very elemental, having the user to control the movements joint by joint. This structure represents a major simplicity of the control system, but requires a higher user's ability. But, the experience has shown, that with an adequate training, it is possible to perform efficiently very different kind of tasks, having certain complexity. These results have been obtained by users motivated by the utilization of this kind of technological aids.

On the other hand, the use of pneumatic technology limits the force that the arm can perform, and even the robotic arm can manipulate objects up to 4 Kgs., it is very adequate and safe to operate very close the user.

Another robot with these characteristics is "Magpie" [8] developed in cooperation by Oxford Orthopedic Engineering and Nulfield Orthopedic Center in 1994. It is a powerless articulated arm, that is, the movement of the arm are obtained from the actuation of some part of the user's body, such as the head or a foot, and the propagation of these actions through simple mechanical transmissions. This mechanical arm, is more limited, but allows to perform tasks such as approach objects, or to feed oneself without external aids.

4.2 Stand alone robots

This kind of assistant robots have been conceived to operate very close to the user, but since they are installed on a fix base, independent from the wheelchair, they do not have any weight or consumption constraint. Therefore, these robots can count on any kind of peripheral devices to provide a higher versatility and more intelligence to the robot interface for its control. Thus, the robotic unit can be provided with a vision system designed to guide the robot towards the user, from more precise orders, or a vision system to locate and to recognize the most frequent elements in the environment. It could also be provided with a voice recognition and a voice synthesizing system or the adequate interface to control other elements of its environment [9] , fig. 9.

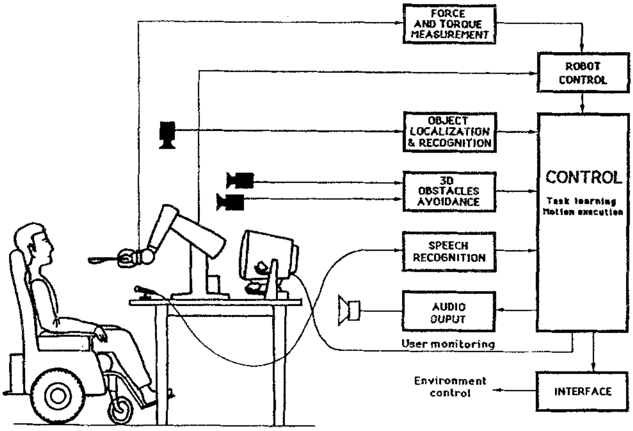


Fig.9 General structure of an assistant robot

One of these robots is Tou, developed at the Universitat Politècnica de Catalunya from 1989 to 1992. It is characterized by its soft structure, to guaranty the user safety and at the same time to offer a more friendly presence and touch to the user [10].

Its architecture is constituted by a set of cylindrical shaped foam rubber deformable modules: Each cylinder has two pairs of antagonistic wires, that produce its deformation in two orthogonal directions, actuated by electrical motors. This architecture enables to deform the arm to obtain the adequate curvature for the end effect to reach the point desired by the user, fig. 10.

This robot has available different kind of interfaces, according to the user's needs, to interpret a set of basic orders that are: a voice recognition system, an adapted keyboard or a joystick. A vision system aids the robot guidance towards the objects of the environment for their grasping. Since the robot flexible structure carries with it a high imprecision in its movements, this guiding support facilitates the user's arm guidance.

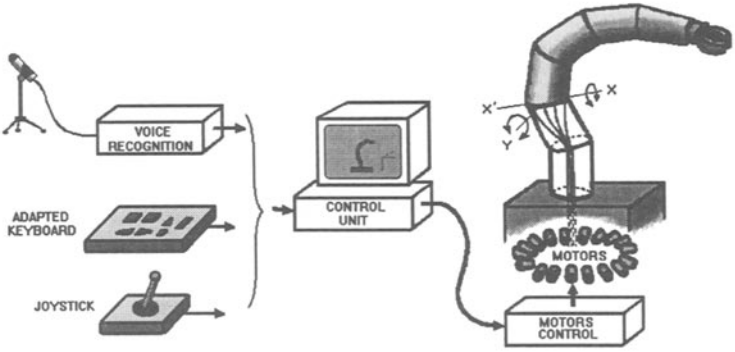


Fig.10 Structure of Tou

Another kind of stand-alone robots, even it is endowed with wheels to move it more easily when necessary beside the user is "Handy". This assistant arm was developed at the University of Keele [11] in 1993 and has been successively improved. It is mainly oriented to feed the disabled user. Its movements are pre-programmed and the user can fix his or her own rhythm to bring the food from the plate to the mouth, with an interface adapted to his remaining movements.

The assistant robot "Isac" (Intelligent Soft Arm Control) constitutes another aid with these performances. It was developed at the University of Vanderbilt (1991) and manufactured afterwards in Japan. This arm is pneumatically actuated and it is constituted by inflatable elements, equivalent to inflatable muscles, called rubbertuators [12].

This arm is also provided with a vision system that corrects the trajectory towards the different objects on a table, to facilitate user's operation.

4. 3 Assistant robots on a mobile base

To increase the arm accessibility without requiring a too big structure that predisposes negatively the user to utilize it at home or at work, other projects have been developed providing the robot with mobility within the required work space, using either rails or mobile platforms.

The prototype developed at the Department of Veterans Affairs in Palo Alto, (CA) in 1986 is DEVAR (Desktop Vocational Assistant Robot). It consists of a Puma-260 robot [13] installed on rails, thus reaching a wide working area. Fig. 11.

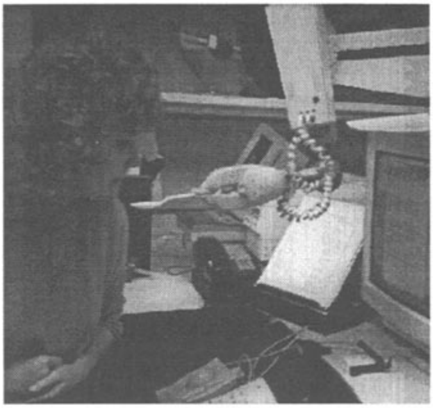


Fig.11 The Devar workstation

This arm is controlled with a joystick and the computer keyboard if the user's remaining movements are enough, or by voice. The robot can either approach food to the user's mouth or to take papers from a file of books from a shelter. It manipulates a CD or can use the microwaves to heat food or to approach some pills.

RAID (Robotic for Assisting the Integration of the Disabled) is the European alternative to DEVAR, developed in the frame of the TIDE program and coordinated by Armstrong Ltd. UK. The main robotic tasks are the manipulation of office objects, documents, books, diskettes located in the supports designed for the workstation. The robot, the system and the wheelchair are controlled by the same interface located in the wheelchair [14].

In spite of the successive improvements of the initial prototypes and the attained effectiveness, these two systems are not yet commercial products due to their high cost and the environment complexity.

Another version of assistant robot developed by the Veterans Administration R&D Center in 1988 is MOVAR (Mobility Vocational Robot) that consisted in installing the same Puma robot on an omnidirectional three-wheeled vehicle, endowed with three mechanum wheels oriented at 120° one from the other to obtain the omnidirectional movement. This mobile robot has a laser scanner for its location in the working room, proximity sensors to avoid collisions and a TV camera mounted on the robot arm to visualize the objects the user wants to manipulate.

This prototype has been used to experiment and to demonstrate the capabilities of a mobile robot with these characteristics to increase the autonomy of disabled people, but its normal use is still far away due to its high cost.

Another robot developed at the SSSA in Pisa, 1992-93 is URMAUD (Unità Robotica Mobile per l'Assistenza ai Disabled). The arm has 8 degrees of freedom to provide both a higher mobility and flexibility and to facilitate its folding. The three fingered hand is endowed with tactile sensors. The manipulation of objects is complemented with a TV camera that visualizes the working area [15]. This robot is the one used in the project MOVAID (MObility and actiVity AssIstance for the Disabled) developed in the frame of TIDE (1994-97). Its goal was the integration of a complete system to be operative in a domestic environment, including not only the arm and the mobile base, but also the control of the whole assistant resources at home.

This development has tried to use all the technological resources to make possible to any kind of disabled to get a high degree of independence in the five basic environments: the kitchen, the bedroom, the living-room the study and the bath-room. Some representative robot tasks are to open the door of the refrigerator, taking some food, heating it in the microwaves and putting it on the table. The availability of a moving base allows to manipulate objects in different rooms.

5. Conclusions

In this chapter a survey of the evolution suffered in the field of rehabilitation, from conventional wheelchairs up to autonomous assistant robots has been presented.

The results of the works presented show that with current technological resources it is possible to develop many types of aids that can be adapted for people with different degrees of impairments. But, there is still important problems to be solved such as: power storage, that limits autonomy and forces to re-charge batteries; the miniaturization of devices, since their external aspect conditions their acceptance; and the development of more intelligent interfaces to simplify still more the use of these equipments by persons not ready to use automated equipment, and that frequently have not only motor disabilities but also visual impairment.

Cost is also a decisive factor in the acceptance of these products. The major diffusion of this technology and products and the important and increasing market for them will presumably produce a reduction of costs in the next years.

6. References

[1] Amat, J. 1994 "*Technology for independence*" First International Conference on Robotics in Medicine, Robomed'94, Barcelona, Spain

[2] Gelin, R., Detriché, J. and Soulabaille, Y. 1994 "*A navigator on a wheelchair*" International Conference on Rehabilitation Robotics, ICORR'94

[3] Schraft, R. D., Wagner, J., and Schaeffer, C. 1998 "*Mobility Aiding Systems*" Technological Aids for Disabled, Ed. Inst. d'Estudis Catalans, Barcelona, Spain

[4] Okada, Y. And Kato, I. 1978 "*Intention control of mechanical arm prosthesis*" 3rd. CISM_IFToMM, Symposium on Theory and Practice of Robots and Manipulators

[5] Kato, I and others. 1987 "*The Waseda Hand*" Internal report, Waseda University, Tokyo, Japan

[6] Rosier. J, and others. 1991 "*Rehabilitation Robotics, the Manus concept*" IEEE 5th International Conference on Advanced Robotics, Pisa, Italy

[7] Hennequin, J., Platts, R., and Hennequin, Y. 1992 "*Putting technology to work for the disadvantaged*" Rehabilitation Robotics Newsletter, Vol. 4, N. 2

[8] Kumar, V. And Bajcsy, R 1996 "*Design of customized rehabiliattion aids*" 7th International Symposium on Robotics Research. Springer, Munchen, Germany

[9] Casals, A. 1994 "*Assistant arms for daily living*" First International Conference on Robotics in Medicine, Robomed'94, Barcelona, Spain

[10] Casals, A., Villà, R. and Cufí, X. "*Tou, an assistant arm: design and control*" IEEE 6th Int. Conference on Advanced Robotics, ICAR'93, Tokyo, Japan

[11] Jakson, R. D. 1993 "*Robotics and its rôle in helping disabled people*" Engineering Science and Educational Journal

[12] Kawamura, K. "*Prospects of research on intelligent robotics systems using flexible actuators at the Intelligent robotics Lab*" Research Report, Vanderbilt University

[13] Perkash, I. and others 1990 "*Clinical evaluation of a vocational desktop robotics aid for severely physically disabled individuals*" Report R&D Dep. of Veterans, Palo Alto-CA

[14] Dallaway, J. L and Jakson, R. D. 1992 "*RAID- a vocational robotic workstation*" International Conference on Rehabilitation Robotics, Keele University, U.K

[15] Dario and others 1995 "*MOVAID, a new European joint project in the field of rehabilitation robotics*". 7th Int. Conference on Advanced Robotics, Barcelona, Spain

Robots in surgery

Alícia Casals
Dep. Enginyeria de Sistemes, Automàtica i Informàtica Industrial
Universitat Politècnica de Catalunya
Barcelona (Spain)
casals@esaii.upc.es

Abstract: Surgical robotics constitutes a relatively recent research field, but in a few years has expanded enormously. This is due to the fact that advances in other robotics areas can be used to improve the working conditions in surgical procedures or in aspects very close to surgery and medical robotics in general. As medical robotics applications increase it is possible to appreciate the huge potential of using robots in this field. Technological progresses in subjects such as sensors, mechatronics, actuators, control strategies and computers have opened the door to robots for entering in the operating theatre. This chapter constitutes a short overview of the current state of robots in surgery, surveying different applications already in use and analyzing the perspectives that research and technology offer to progress in this field.

1. Introduction

The progress in surgery procedures has evolved from open surgery, that is a completely manual operation, to minimally invasive surgery, in which the surgeon relies, in a higher or lower level, on technological aids, such as medical images, microdevices, sensors, computers and mechanical manipulators [1], [2]. New advances are being done towards the development of dedicated machinery for non invasive therapies, such as external beam radiotherapy, ultrasound lithotripsy or endoscopy.

Human robot cooperation has shown to be the best compromise to solve many advanced robotics problems since this cooperation benefits from the best of both (Fig. 1). Although robots can be easily programmed in simple robotic applications, advanced robotics programming runs into difficulties due to uncertainty and dynamic environment changing conditions. While humans offer good performances in what refers to intelligence, intuition, adaptability and learning capabilities, they lack of sufficient precision for certain actuations. Humans get tired in long interventions lessening their efficiency and reliability. On the other hand robots are extremely useful tools due to their repetitivity, speed and reliability.

Minimally invasive surgery has already become a normal practice today. Currently new endoscopical procedures are being investigated in order to increase its benefits, both for the surgeon and for the patient. These benefits are: less damage to surrounding tissues, less post-operative recovery time and no scarring. The extended use of these techniques has demonstrated that there are still important limitations in its practical applications due to the surgeon lack of direct visibility,

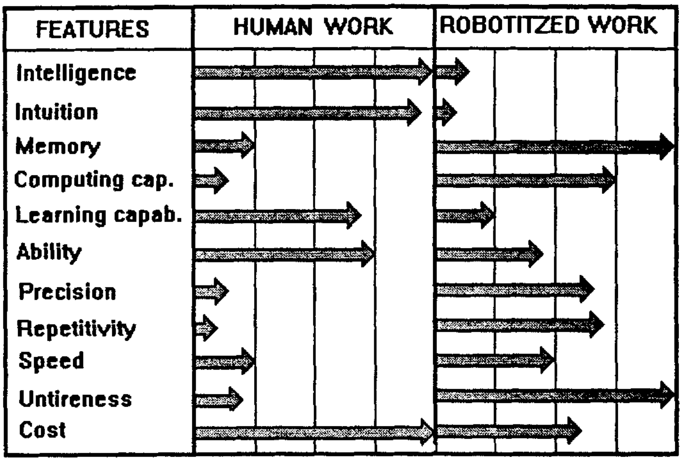


Fig. 1 Comparison of human and robot performance

sensing and free operating space [3]. This leads us to investigate on how to improve surgical procedures.

The main steps to follow in a computer aided surgical procedure are: medical images analysis from different sources; intervention planning, from the diagnosis obtained from images and other data; and finally, intervention execution being either computer supervised and assisted, robot assisted or even task executed by a pre-programmed robot with the supervision of the surgeon. Looking at figure 2 we can see how conceptually different technological areas can support this procedure. The first requirement is the surgeon knowledge and expertise about the problem and the possible surgical procedures. The surgeon works from registered and processed images (registration consists on the integration of the different kind of images, MRI, ultrasonic, CT, X ray ...). From this new pre-operative image or images, computer graphics and simulation techniques can aid to plan the operation. Computer techniques such as trajectory following and world modeling constitute additional tools to preprogram the intervention. Afterwards, during the intervention, new sensor measures and images can be correlated with preoperative data to supervise and guide the pre-planned procedure. The availability of dedicated medical instruments, perception and robotics leads to the concept of CAS, Computer Aided Surgery, that benefits from the best performances of the surgeon and those of technology and robots.

2. Surgical Robots

The great variety of surgical interventions and the different complexity levels of these procedures present a very wide range of problems that require different kind of technological support. For this reason, we can consider very different types of surgical robots and robotic aids regarding the kind of tasks to perform, the level of human-robot cooperation or the level of interaction of robots with the environment.

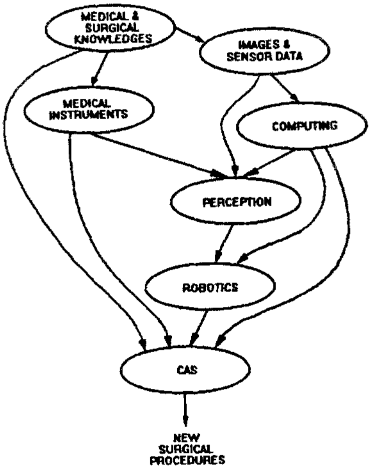


Fig. 2 The Computer Assisted Surgery Concept

These robots, or robotic systems, can be classified into three different groups according to their functionality.

- Local guidance
- Teleoperation
- Pre-programmed robots

2.1 Local guidance

The guidance of the surgical instruments by the surgeon based on the preplanned intervention relies on the use of some sort of electromechanical support. Among these devices there are three different categories [4]: localizers or passive arms, semi-active devices and synergistic devices or active guidance.

2.1.1 Localizers

Localizers are devices that simply measure the coordinates of an instrument or a pointer moved by the surgeon. They are mecanically passive devices that allow the surgeon to connect the real world with the information world of medical images. Localizers can range from optical tracking devices to passive robotic arms.

Optical tracking devices are based on the 3D measurement of some reference points. The reference points are marks located either in the patient body or in the own instrument or probe. The marks use to be LEDs, (three or more) which position are detected by two or three CCD cameras mounted in a fixed structure. From the cameras data the 3D position of the reference points can be calculated by means of stereovision (Fig. 3). Optical trackers are relatively simple and useful devices designed to assist the surgeon in guiding the surgical instruments through the right path towards the desired point.

Passive arms are human powered robots guided by the surgeon, which maintains full control over the entire procedure. The surgeon guides the instrument towards a reference point, visualized in real time over the patient's image, or following a given trajectory. The final precision in reaching this point depends on how far the point of interest is from the reference marks. These powerless robotic arms work under the concept of impedance control, the arm moves freely guided by the surgeon gestures. If the joints are provided with brakes, the surgeon can guide, from the visualized image, the instrument to the desired position, that can have been previously planned, and then lock the arm in that position. This technique is applied in [5] using a 6 degrees of freedom robot for percutaneous surgery in urology. In this application the urologist manually holds the needle along the trajectory to track, that is visualized in the image, and the passive arm maintains the required alignment of the needle injector. This procedure improves needle placement accuracy, intervention duration, patient safety, sterility and surgeon radiation exposure.

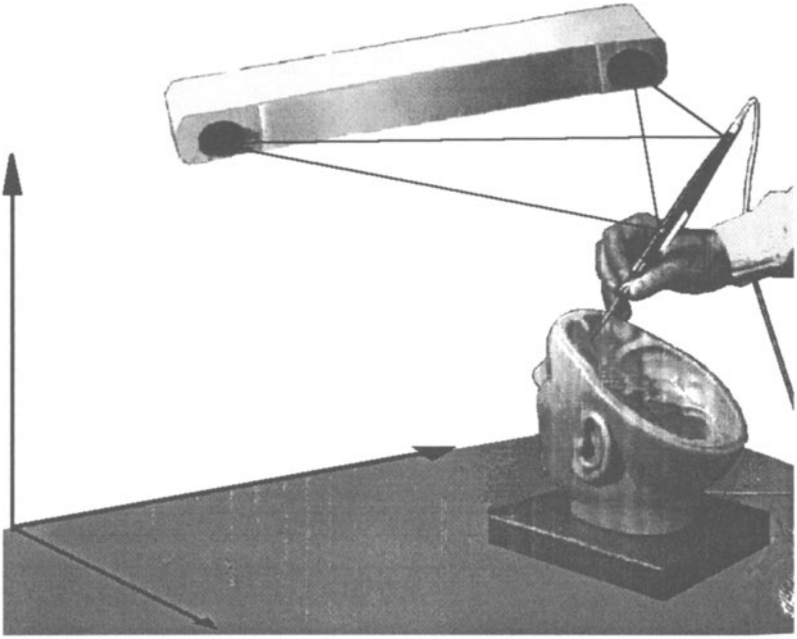


Fig. 3 Optical localizer

2.1.2 Semi-active devices

When it is possible to program partially a trajectory, the availability of a pówered robot allows a new kind of cooperation. For instance, in neurosurgery [6] it is common the need to track a straight line, either to inject some dose, to extract tumors or for electrode implantation. In this case, the robot can be used as a mechanical guide through which the surgeon introduces a drill, a probe or an electrode until it reaches a predefined mechanical stop. Thus the robot guaranties the precise entrance and advancing movements through the brain.

2.1.3 Synergistic devices

The concept of synergistic devices is defined in [4]. The aim is to match to an adequate degree the surgeon-robot cooperation by taking profit of the robot accuracy and reliability while letting the surgeon to decide each action based on his intelligence, knowledge, expertise and ability. In this operating mode the surgeon guides the surgical instrument, which is held by the robot, at his will. But, these movements are supervised by the computer to avoid the instrument entering within dangerous zones, those that the surgeon has previously defined in the simulation or preplanning phase. In the preplanning phase the surgeon defines virtual walls to avoid for instance to move the instrument too close to a nerve while cutting a bone. When the robot, guided by the surgeon reaches a predefined surface the safety control strategy can force the robot to stop, as it if collides with a virtual obstacle, or to damp the movement emulating a collision with a virtual pillow. Fig. 4 shows this concept applied to laparoscopic surgery.

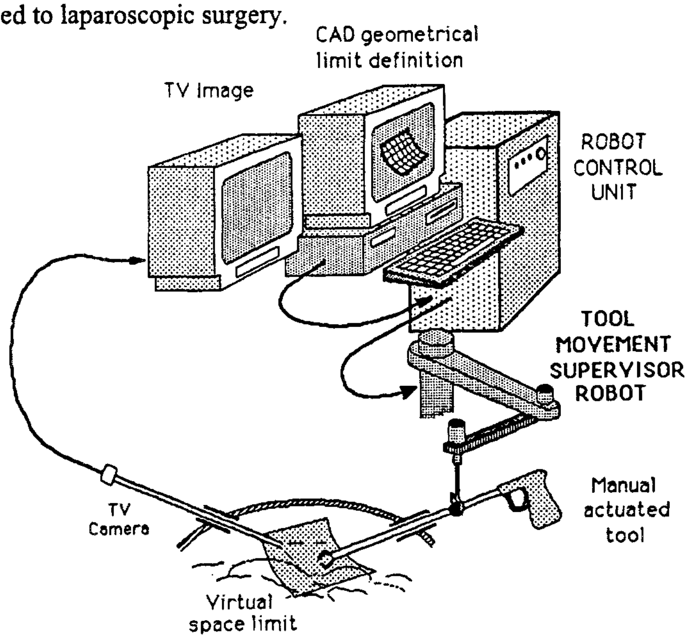


Fig. 4 Geometrically bounded movements

2.2 Teleoperation

Teleoperation constitutes also a very good mean to share the best performances of both human and robots. The surgeon guides the robotic arm, through an adequate hands-off interface (headstrips, footpedal, master arm, joystick, voice...). While the surgeon decides the best actions to carry out during all the surgical procedure, the robot control unit executes the ordered movements and actions providing the required accuracy, or even supervising the movements, thus avoiding any dangerous error caused by possible inaccurate or wrong surgeon movements towards protected areas. There already exist different commercial robots operating in this mode. They have been mainly applied in laparoscopic surgery, where the surgeon needs a third hand to move the laparoscope, beside the instruments. Nevertheless, this concept is applicable to very different surgical fields.

Teleoperation can be unilateral, that is, the robotic arm follows the movements ordered by the surgeon, which is assisted by the computer to avoid dangerous situations. When besides this one way orders, there are some feedback data towards the surgeon, to "feel" the patient organs or body elements, we have bilateral teleoperation, or a telepresence system.

2. 3 Pre-programmed robot

It would be desirable to have a system able to operate autonomously being supervised by the surgeon. The ways a robot can be programmed to execute a task are: first, to preprogram the task previous to its execution, assuming all the steps to follow can be predefined and no deviations from the programmed tasks are foreseeable. This is not a very common situation as a consequence of the uncertainty due to sensors, to the patient movements during the intervention, that change the situation in the operating area, or to multiple and unexpected situations in which the surgeon has to take new decisions on-line.

The second way to program the robot is sensor based control. In this case the robot is programmed to execute a trajectory or a sequence of trajectories or actions, but interacting with sensors that provide the information of how its execution is performing. For instance, a trajectory can be executed provided a given force is not surpassed, or maintaining the applied force between two predefined values. In these cases, different robot control modes are used, position control, force control or hybrid. Additionally, impedance control is required for the situations in which the surgeon needs to take the control. Then, the robot remains free to be moved at the surgeon will.

Anyway, this active mode has shown to be only possible when operating over solid structures as bones, but never in soft tissues that deforms while they are manipulated. Furthermore, only part of the intervention can be pre-programmed, mainly some defined trajectories. The surgeon has to prepare the intervention, placing the robot in position and supervising continuously the pre-programmed actions.

3. Simulation

Simulation has shown to be a useful tool in different aspects of surgery. First, for training, and second, and very important in surgical robotics, to pre-plan or to program an intervention.

Training simulators allow surgeons to train new procedures for as long as required introducing any kind of practical and pathological conditions that can appear in reality. This facility offer an environment that can not be simulated with phantom or even in cadavers, further avoiding the use of animal models or the intervention of patients. This aid is specially useful when dealing with deformable organs. The work in [7] is based on the modeling of the liver with its anatomical details. Using the elasticity theory it is possible to simulate the effect of each surgeon action and evaluate its effects. The surgeon receives visual and force information from this simulation.

The complete operating room is simulated in [8], for knee surgery, to foresee all the possible mobile objects during an intervention, and thus analyze before hand the consequences of every robot movement.

Computer graphics, modeling and simulation constitute the base for a robotic system to be able to execute a pre-planned and pre-programmed part of an intervention. These techniques can also assist the surgeon to plan in the computer the best strategy to follow, previous to the intervention, given a concrete pathology of the patient in order to improve his efficiency and reduce intervention time.

Orthodoc [9] is one of the first robotic prototypes that work from the CT images to pre-plan a robotic intervention, in this case for total hip replacement surgery. Other systems based on this principle have been also developed or are under development taking advantage of the state of technology on what refers to CAD/CAM systems well established in industrial environments. The process, as shown in figure 5, consists on creating a 3D model of the bone to define how it has to be mechanized as well as to create, based on this model, the exact shape of the prosthesis to be implanted, in order to fit exactly into the femoral cavity. Once the model of bone and prosthesis are available, the surgeon can simulate the implant on

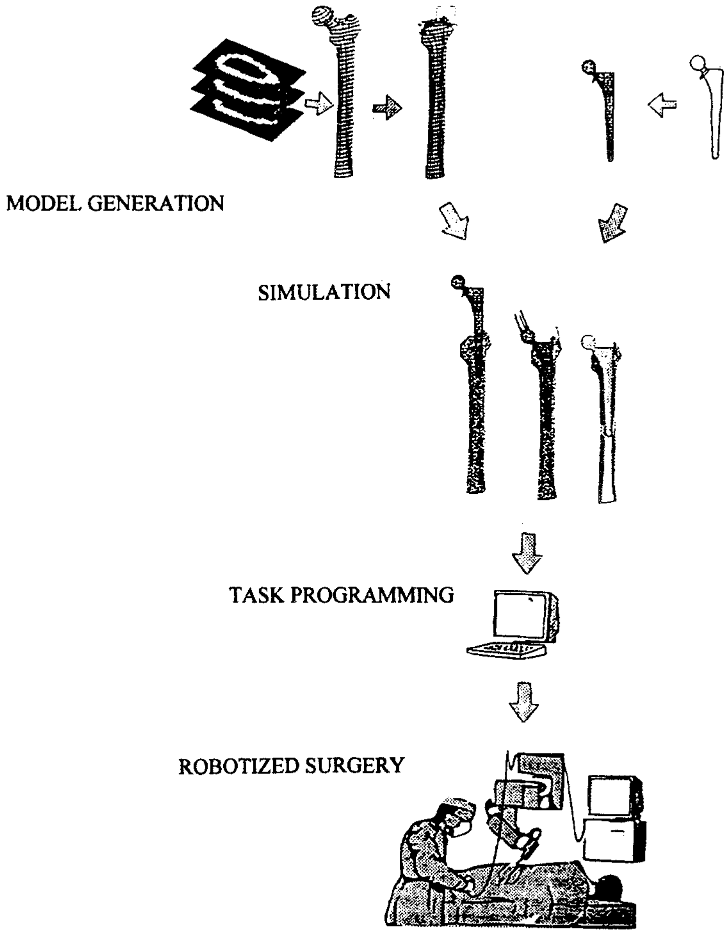


Fig. 5 The CAD/CAM concept applied to robotics in surgery

the computer screen and pre-program the robot for bone machining. All these procedures can be performed previously and out of the operating room. During the intervention the surgeon prepares the operation exposing the bone and fixing it, to define the reference frame for the robot to work. The surgeon supervises the process comparing the real process, cutter position superimposed in the CT images, with the pre-planned task. All these procedures require the possibility of the surgeon to interact with the system when necessary. Experimentation and already real procedures executed with humans in the last years show, in the same way that in industry, that robotized mechanization is far more precise and safe than manual work.

With the use of computer graphics it is also possible to generate synthetic images that can be superimposed over the real images obtained from a patient and to test different strategies and actuations without the need to physically carry them out. An example of its application in laparoscopic surgery is shown in figure 6. These images visualize the process of the inginal hernia repair interventions with a polypropylene mesh and the synthetic image generated. In this case, it would be possible to simulate the introduction and application of the mesh on the screen, checking the mesh size and its right position and orientation over the desired element to obtain the expected results. In real practice it is often necessary to guess the approximate size and shape of the mesh from the laparoscopic image, cut the mesh and intrude and apply it. When the applied mesh does not fit in size or shape, the surgeon has to try several times before reaching the expected result.

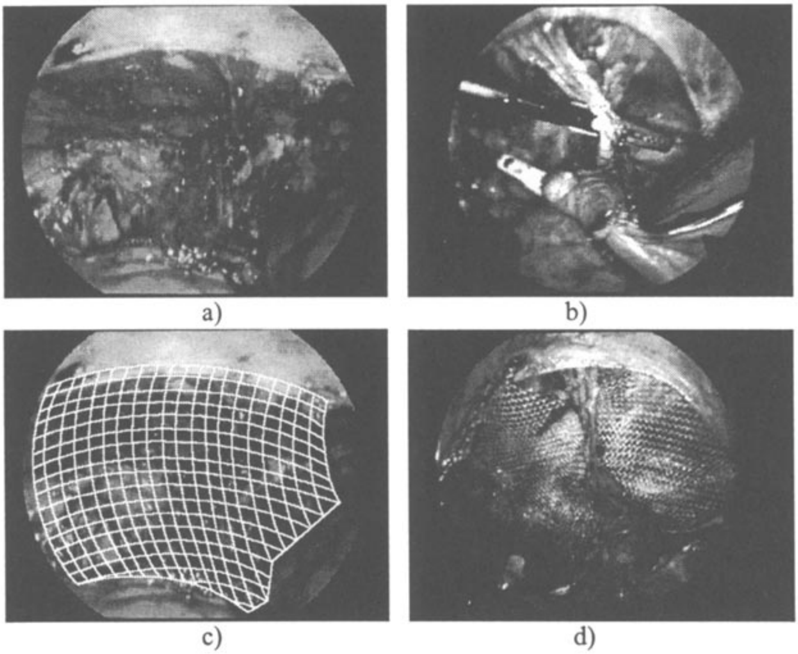


Fig. 6 Inginal hernia repair. a) Image of the inginal zone, b) a step forward in the repairing process, c) synthetic image, d) after the application of the mesh

4. Current applications and future trends

Robotics has already shown its effectiveness in surgical applications in very different areas, having for each of them very different requirements. In spite of this progress and proved results, the use of robots in surgery is far less generalized than it could be. The main reason is safety, an essential issue when working in the operating room. Due to environment and working conditions laws regulation makes it difficult to experiment in procedures even though, they have proved its efficiency, performance and reliability in other environments.

Without trying to be exhaustive, a sample of some of the most relevant applications shows robotics status and possibilities.

4. 1 Applications

One of the most advanced areas is **orthopedics**, that, as already mentioned, has imported CAD/CAM techniques and procedures from industry and has proved its efficiency in surgical procedures. Its success is due to the fact that the robot can operate more precisely, using numerical control techniques, than the surgeon, that needs to operate manually in restricted environments with limited accessibility and with low perception. The system consisting of Robodoc, the mechanical device and Orthodoc, its computer support, is a good example of this kind of robots used in applications such as femur implants or hip replacement.

Computer assisted spine surgery has proved, through a significant number of trials, to be more accurate and safe than manual [10]. Transpedicle screw insertion, for rigid segmental fixation, has been experimented using a technique that combines pre-operative CT imaging with intra-operative passive navigation.

Laparoscopic and other minimally invasive techniques have also proved to be a field of robotics application and many works have been developed either in the guidance of the laparoscopy [11] or to improve surgeon perception by means of 3D imaging or force feedback. In [12] a study is done on the requirements of manipulators for minimally invasive surgery: endoscopic devices, the surgical instruments, the interfaces or the sensors.

Neurosurgery is a field that has motivated a great amount of research since robotics can greatly contribute to improve the interventions due to the high level of precision these procedures require. In this area estereotaxis, that is the precise location of some body reference points, has been largely studied and applied to calibrate the robot [13]. One of the first relevant robots in stereotactic neurosurgery is Minerva [14]. This kind of surgery consists in the introduction of a probe of 2 to 3 mm through a hole drilled in the skull, in order to reach a point inside the brain. Some of the common operations are: Thalatomy, which is performed to reduce the Parkinson disease, Evacuation of hematoma abscess, Biopsy of tissues for histological examination or Implantation of radioactive sources for irradiation of tumor.

The problems due to stereotaxis frame bulkiness and its application to the patient has motivated that an important effort be put on the study of registration techniques and detection of natural relevant points in the patient body to avoid the use of the reference frame. Thus, frameless stereotactic methodologies have been studied for a long time[15], [16] either tracking natural points or with the aid of localizers.

Stereotaxy is useful to perform brain biopsies or procedures for disfunctions because compared with manual operations it lesions or stimulates smaller cerebral areas. But, it presents some limitations for removal of lesions of more substantial volume. For that reason, new surgical instruments are developed integrating image processing techniques, from a microscope camera, and a stereotactic robot [17].

Eye and ear surgery are also characterized by the need of operating in very small areas, and frequently with access difficulties. On the other hand, accuracy is essential for such interventions. An example of eye surgery is Keratotomy, operation that consists on making 4 or 8 incisions on the cornea [18] to produce an increase of its curvature for correcting strong myopia. These incisions require a precision of 0,01mm and be performed maintaining the bistoury perpendicular to the cornea surface. Thus robotics accuracy, and repetitivity improves manual performance through teleoperation. The surgeon can operate more comfortable from magnified images thus avoiding tireness and stress. For such applications the man-machine interface is a key factor to efficiently control the robot. Laser treatments for other applications like retinal photocoagulation, diabetic retinopathy, and other pathologies, are other eye procedures where robotics and CAS techniques can contribute to improve efficiency.

In [19] the problem of operating inside the ear in a stapedotomy procedure is faced up. The difficulties to control drill protrusion of the stape bone are: small size of the working zone, small and narrow access, low tactile feedback, compliance of the stapes and its unknown thickness, that can vary between 0,1mm and 2,5mm. To improve current procedures a drilling tool system has been developed. It consists of an automatic micro-drill provided with strain gauges to control the applied force and torque and position sensors to control position and velocity.

While robotics can contribute enormously in increasing performance in orthopedics, in machining hard materials, bones, using CAD / CAM techniques, other areas of surgery deal with deformable objects. Interventions in **soft tissues** such as human organs: liver, brain, heart or the skin can not rely directly on these techniques but can also benefit from robotics [20]. Soft tissue surgery differs from orthopedics in that it tends to be more disordered. Transurethral resection to the prostate, for instance, has been experimented using a special purpose robotic frame to remove a conical section of tissue. For this application the surgeon positions and locks the resectoscope using a flexible snake type arm.

4. 2 Microrobotics

Microrobotics and micromechanisms developments have been the key towards the advances in endoscopic procedures for minimally invasive surgery, therapy or diagnose. Micro-machines development find new problems due to the appearance of phenomena absent in bigger devices. The study of different kind of micro-devices [21] has allowed the progress in doses implants, stimulators and other actuators. Some examples of the designed devices are micropumps, microturbines, micromotors or microhoses, to activate such implants or endoscopic devices. The development of endoscopic devices for diagnostic analysis and actuation has lead to the study of new mechanisms to enter through a troccar or a natural entrance to the body. An active forceps built from SMA (Shape memory alloy) pipes is described in [22]. The use of such flexible devices facilitates accessibility in laparoscopic surgery. Other devices are based on movement propagation through modular elements, worm movements are a good solution to advance through pipe shaped parts [23].

In [24] a review is done about the different fields where micromechatronics can be applied in medicine. In this work, the problems associated with the access, sensing, actuation or the implants, to a living organism are analyzed. The compatibility of artificial devices with natural organs or body parts forces to enter the bio-mechatronics field. In this line, the research in bio-sensors, bio-artificial organs and neural interfaces is pointed out.

4. 3 Telesurgery

The development of techniques for minimally invasive surgery makes this kind of surgery easier to robotize than open surgery. This is due to the fact that these surgical instruments are more a robot end effector than the classical tools and the surgeon hands, and their actuation is easier to automate. The difficulty, with respect to industrial robot applications, comes from the uncertainties of the working environment that prevents to program a priori a complete task. Robotics in this case offers the possibility of the control from human gestures, that is, teleoperation [25].

The requirements in teleoperation tasks do not rely uniquely in providing the desired mobility to the tools needed for a given surgical intervention, which do not implies technological problems, but also in the availability of sensors feedback to the operator, in this case the surgeon. This feedback perception allows to operate safer and with more efficiency.

Therefore, the possibility to introduce tele-robotics in minimally invasive surgery is conditioned to the development of the adequate perception systems that allow to manipulate indirectly the tool. Once this problem is solved, that is the tools are mechanically manipulated from the surgeon gestures, he or she can already operate from outside the operating theater. This possibility can mitigate both, accessibility problems that frequently appear, or even to have available in a given situation a specialist that is physically far away. An example of the first type of interventions is described in [26] for micro-blood-vessel suturing. On the other side, an on-line interactive ISDN network for telemedicine and telepresence is described in [27]. This kind of operation can then be done with the same guarantees than the operations done with the direct presence of the surgeon close to the patient.

Both research lines are already alive and some experimentation has been done. The possibility that these specialists operate at any distance and that they are only required in the more critical moments can bring in the future an increase of quality. It will be possible both, to work with an additional support and also to have the possibility to remove part of the work equipment outside the operating theater, making the working areas larger and so more comfortable for the surgeon.

5. Conclusions

From the experience already existing it can be appreciated that robots are now useful, or more realistically, they are a real need in surgery to improve quality and safety of interventions and to make possible new surgical procedures not thinkable some years ago. The effective use of these robotics technologies requires important adaptations to be applied in this medical environment: size, accessibility, sterility, safety are still important issues to increase its effectivity and be more acceptable by the medical community. New emerging technologies such as bio-mechanics or the development of haptic devices and friendly human-machine interfaces are among others important areas to advance on, to expand the use of robots in this field.

References

[1] Finlay, P. A. 1994. "*Small is beautiful*". 1st European Conference on Medical Robotics, Barcelona, Spain

[2] Troccaz, J. 1996.*"Robots in surgery"* IARP Workshop on Medical Robots, Vienna, Austria

[3] Casals, A., Amat, J., Laporte, J. 1995. "*Robotic aids for laparoscopic surgery*". The Seventh International Symposium on Robotics Research, Springer - Munich, Germany

[4] Troccaz, J., Peshkin, M., and Davies, B. 1997. *" The use of localizers, robots and synergistic devices in CAS*". CVRMed-MRCAS, Springer - Grenoble, France

[5] Stoinanovici, D. et al. 1997. "*An efficient needle injection technique and radiological guidance method for percutaneous procedures*" CVRMed-MRCAS, Springer - Grenoble, France

[6] Lavallé, L. et al. 1992. *"Image guided robot: a clinical application in stereotactic neurosurgery"* IEEE International Conference on Robotics and Automation, Nice, France

[7] Cotin, S. et al. 1996. "Towards realistic surgery simulation including visual and force feedback" IARP Workshop on Medical Robots, Vienna, Austria

[8] Götte, H., et al. 1996. "*A new less-invasive approach to knee surgery using a vision-guided manipulator*". IARP Workshop on Medical Robots, Vienna, Austria

[9] Paul, H. A. et al.1992. *"Development of a surgical robot for cementless total hip arthroplasty"* Clinical Orthopedics and Related Research, N. 285, december 1992

[10] Merloz, P et al. 1997 "Computer-assisted versus manual spine surgery: Clinical report" CVRMed-MRCAS, Springer - Grenoble, France

[11] Casals, A., Amat, J. and Laporte, E. 1996 *"Automatic guidance of an assistant robot in laparoscopic surgery"* IEEE Intern. Conference on Robotics and Automation, Minnesota, USA

[12] Schraft, R. and Wapler, M. 1995 "*Manipulators make minimally invasive surgery easier*" 7th International Conference on Advanced Robotics. Sant-Feliu de Guixols, Spain

[13] Fankhauser, H. Et al. 1993 "*Robot for CT guided stereotactic neurosurgery*" Proc. Of the XIth Meeting of the World Society for Stereotactic and Functional Neurosurgery, Ixtapa, Mexico

[14] Flury, P et al. 1992 "Minerva, a robot dedicated to neurosurgery operations" 23th International Symposium on Industrial Robots, Barcelona, Spain

[15] Kall, B. A. et al. 1993 "*Three-Dimensional display in the evaluation and performance of neurosurgery without stereotactic frame: More than a pretty picture?*" Proc. Of the XIth Meeting of the World Society for Stereotactic and Functional Neurosurgery, Ixtapa, Mexico

[16] Henri, C. J. et al. "*Registration of 3-D surface data for intra-operative guidance and visualization in frameless stereotactic neurosurgery*" CVRMed'95, Springer - Nice, France

[17] Giorgi, C. 1996 "*A stereotactic microscope for robot assisted surgery*" WAC: Robotics and Manufacturing, Montpellier, France

[18] Guerrouad, A. and Vidal, P. 1991 "*Advantage of computer aided teleoperation in microsurgery*" Fifth International Conference on Advanced Robotics, Pisa, Italy

[19] Brett, P. N. et al. 1996 "*The advanced surgical robotic micro-drilling system applied to the stapedotomy procedure*" IARP Workshop on Medical Robots, Vienna, Austria

[20] Davies, B. 1994 "Soft tissues surgery" ". 1st European Conference on Medical Robotics, Barcelona, Spain

[21] Menz, W. 1994 "Robotics in minimally invasive neurosurgery". 1st European Conference on Medical Robotics, Barcelona, Spain

[22] Nakamura Y. Et al. 1997 "Active forceps for endoscopic surgery" Fifth International Symposium on Experimental Robotics", Barcelona, Spain

[23] Slatkin, A. B. and Burdick, J. B. 1995 "*The development of a robotic endoscope*" Fourth International Symposium on Experimental Robotics, Stanford, California

[24] Dario, P. et al. 1996 "*Michomecatronics in medicine*" IEEE/ASME Transactions on Mechatronics, Vol. 1, N. 2 June 1996

[25] Becjzy, A. K. 1995 "*Virtual reality in telerobotics*" 7th International Conference on Advanced Robotics. Sant-Feliu de Guixols, Spain

[26] Mitsuishi, M. et al. 1997 "*Tele-micro-surgery: analysis and tele-micro-blood-vessel suturing experiment*" Fifth International Symposium on Experimental Robotics", Barcelona, Spain

[27] Bejczy, K. A. and Fiorini, P. 1995 "*An On-Line Interactive ISDN Network Connection for Telemedicine / Telescience*". Report Jet Propulsion Lab

Part Four

Legged and Climbing Robots

Legged Walking Machines
Friedrich Pfeiffer, Steuer Josef, Thomas Roßmann

Climbing Robots
Gurvinder S. Virk

Legged Walking Machines

Friedrich Pfeiffer
SteuerJosef
Thomas Roßmann
TU München
pfeiffer@lbm.mw.tu-muenchen.de

Abstract:

A survey of walking machines and their todays performance is given. Paper considers two-, four- and six-legged robots and presents especially the author's six- and eight-legged machines.

1. Introduction

Artificial compared to biological walking includes some drawbacks due to the fact that technical performance of robots is still far away from biological performances. Biological design for drives like muscles, for leg construction, for gait generation is much more effective, lighter, more adaptive and with lower consumption than technical solutions. Biological control adapts perfectly to environmental changes and requirements. It is self-learning, highly adaptive and self-repairing. The sensor situation in biology is overwhelming. Biological sensors provide any magnitudes of walking in a lavish way which is superior to all technical sensor systems.

Nevertheless it makes sense to start research on walking machines with our human means, and the results achieved are promising. To have a measure for technical performances we choose a representation of weight-to-size and weight-to-power data for biological and for technical systems. The Figures 1 and 2 illustrate the situation.

With respect to the weight-to-size features we are able to arrange the walking systems along a curve of lightweight constructions like insects or slim mammals and along a curve of massive constructions, where most of the artificial walking machines are arranged. Nevertheless we should keep in mind that inspite of that good position of our technical robots these curves do not tell the whole story. Aspects of walking performance like jumping, walking through rough environment, acceleration and deceleration capacity and the like are missing. With respect to these features technical walking would be significantly inferior.

One of the essential problems in realizing walking machines consists in our still poor drive systems. The possibilities are confined to electric, pneumatic or hydraulic motors and to some special systems trying to copy the muscle behaviour. Figure 2 indicates the situation. Electric drives cover the largest range of applications, pneumatic systems are limited to large weights but rel-

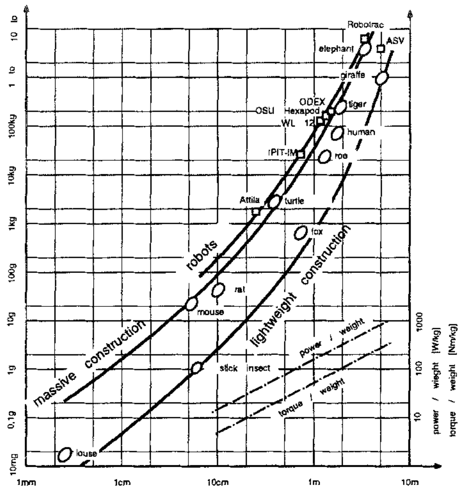


Figure 1. Weight-to-Size Performance [23]

atively low power, and hydraulics applies best to heavy machines with high power consumption. The realized walking machines confirm this trend.

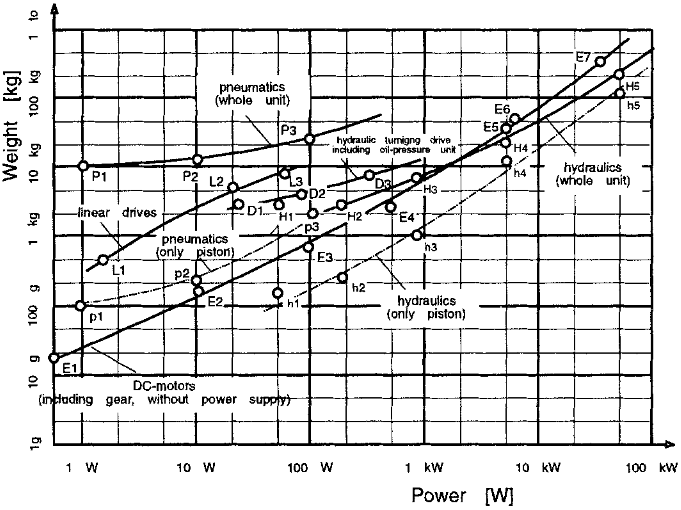


Figure 2. Weight-to-Power Performance for Technical Walking Machines [23]

The following describes some selected machines for two-, four- and six-legged walking, and additionally some other machines. From this we proceed to general guidlines for developing walking robots and discuss solutions having been realized at the author's institute [5,15–20].

2. Walking Machines – A Survey

2.1. Two-Legged Machines

The probably most advanced two-legged robot is WABIAN, which has been presented by the Waseda University, Tokyo, in the year 1997 (Fig. 3). It can walk approximately like a human being, and it is altogether very near to human features. Some dozens of man-years and some million dollars characterize this development, which in addition must be seen before the background of more than twenty years research activities on walking and grasping at Waseda.

A much simpler design has been realized at Harvard University in the US and was presented in 1994 (Fig. 4). It moves on a circle being attached to a rotating arm. One of the goals consists in an investigation of controlling hopping around this circle. The hopping speed is remarkable.

Completely different aspects have been persued with the SHADOW PROJECT from the London University in the United Kingdom, 1993 (Fig. 5). The robot is realized by a wooden leg-skeleton where each leg is driven by 14 air-muscles. Two valves control each of the muscles. The overall control is achieved by neural nets.

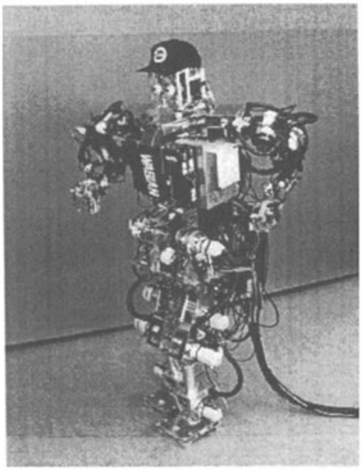


Figure 3. WABIAN, Two-Legged-Machine

The following list gives a rough summary of other two-legged robots (this summary is of course not complete).

Other Two-Legged Robots

- Robot Bipede 1 (1994)
  Ecole Nationale Sup. de Physique de Strasbourg, France
  (0.3 x 0.8 x 0.3 m, 15 kg, 70 W)

- CURBI (1986)
  Ohio State University, USA
  (0.18 x 0.89 x 0.26 m, 9,16 kg, 60 W)

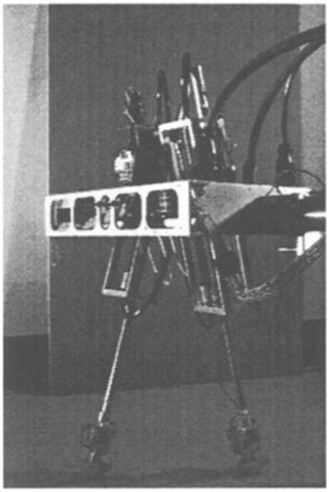


Figure 4. Harvard Experimental Biped, Two-Legged-Machine

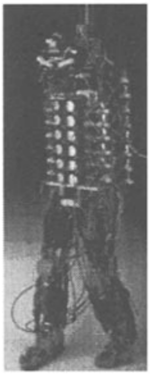


Figure 5. Biped Walker, Two-Legged-Machine

- SAICO (1997)
  Universidad La Salle, Mexico
  (0.29 x 1.1 x 0.56 m, 12 kg, 85 W)

- HITBWR-III (1988)
  Harbin Institute of Technology, China
  (0.6 x 1.0 x 0.4 m, 40 kg, 400 W)

- HONDA-Biped Robot (1996)
  Honda, Japan
  (1.8 m, 210 kg)

- Piernuda (1996)
  Facultad de Ingenieria Unam, Mexico
  (0.67 x 0.95 x 0.15 m, 5 kg)

- Ninjya (1980)
  Miyazaki University, Japan
  (0.7 x 0.7 x 0.3 m, 12 kg, 20 W)

- KDW BIPED (1987)
  Changsa Insitute of Techn., China
  (0.2 x 0.8 x 0.22 m, 16.3 kg, 250 W)

2.2. Four-Legged Machines

In 1994 the University of Massachusetts, Amherst, USA, realized a four-legged robot which can walk in either direction (Fig. 6). Typically as most of the four-Legged machines, this robot does not follow biological solutions but the technical idea of versatile walking. The walking pattern of some crabs is not too far away. THING is a small, lightweight robot moving with batteries.

The ROBUG IIs of the University of Portsmouth (UK 1989, Fig. 7) represents the type of wall-climbing robots, which attach itself firmly to the wall by pneumatic devices. Such wall-climbing machines are realized all over the world with the idea of cleaning sky-scrapers or underwater constructions. One of the problems consists in the power supply with high-pressure air requiring long high-pressure flexible tubes with increasing weight for high or deep constructions.

The four-legged robot RIHMO, realized in the Institute for Industrial Automation, Madrid, Spain, in 1993 (Fig. 8), possesses a cartesian pantograph mechanism for each leg. Pantograph mechanisms are popular solutions due to their stability and their simple drives, which in the case of RIHMO consume only a small amount of power.

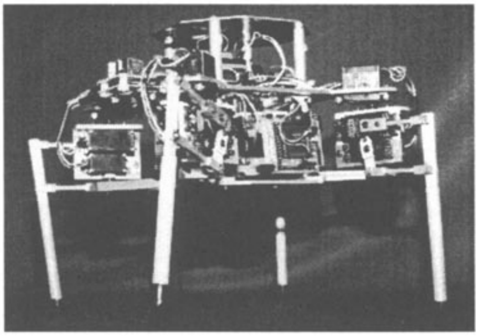


Figure 6. THING, Four-Legged-Machine

Other four-legged machines are listed below.
Other Four-Legged Robots

- UNH Quadruped Robot
  University of New Hampshire, USA
  (0.6 x 0.6 m, 25 kg)

- TITAN VI (1990)

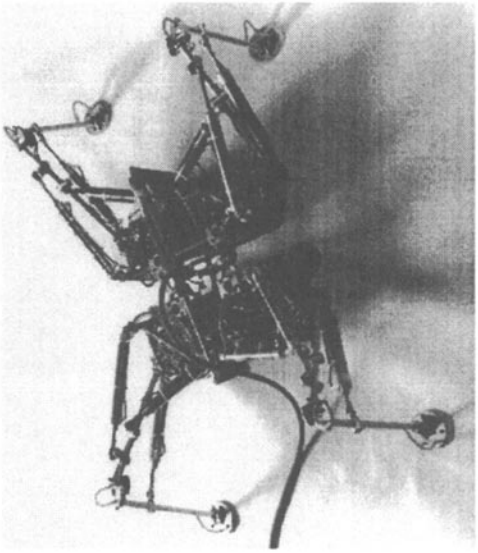


Figure 7. ROBUG IIs, Four-Legged-Machine

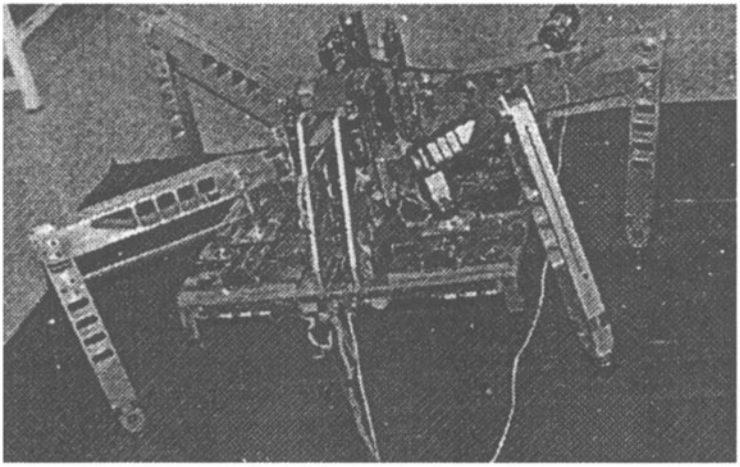


Figure 8. RIHMO, Four-Legged-Machine

Tokyo Insitute of Technology, Japan
(1.5 x 1 x 1.5 m, 190 kg)

- TITAN VIII (1997)
  Tokyo Insitute of Technology, Japan
  (0.4 x 0.6 x 0.25 m, 19 kg)

- NINJA-II (1994)
  Tokyo Institute of Technology, Japan
  (1.8 x 0.5 x 0.4 m, 45 kg)

- roboTRAC
  Universität Duisburg, Germany, ETH Zürich, Switzerland
  (0.55 x 0.15 x 0.27 m, 1.3 kg, 10 W)
- MK-4 (1986)
  Warsaw Univ. of Technology, Poland
  (100 kg, 3 kW)
- MENO II (1995)
  Univ. of Southern California, USA
  (0.5 x 0.3 x 0.5 m, 12 kg)

2.3. Six-Legged Machines

Many very early investigations started with six-legged machines, and all heavy robots are also based on a six-legged design. Six-legged machines are statically and dynamically stable or at least include no particular problems. Gait patterns are easily to realize and well-known. Often applied are wave patterns generating tripod and quadropod walking. One of the earliest and heaviest machine is the famous Adaptive-Suspension-Vehicle designed by Waldron 1982 at the Ohio State University (Fig. 9) [22]. It weighs 3200 kg, possesses a power supply of 35 kW by a gasoline motor which drives a hydraulic system for the legs' motion. An operator can sit in the machine and control it. The legs are realized by a pantograph mechanism. All leg planes are vertical thus reducing considerably the influence of gravitation.

SILEX from ULB Service des Construction Mechanique et Robotique, Bruxelles, Belgique, represents a class of six-legged machines, which can walk in either direction in a similar way like some of the four-legged robots. It was built in 1991 (Fig. 10). The legs are also pantograph mechanisms with three degrees of freedom each. Control is hierarchical with three levels.

Another big machine has been developed in Tampere, Finland. PLUSTECH weighs 3500 kg and is powerd by a Diesel engine (Fig. 11). The hydraulically driven legs possess three degrees of freedom each. The robot performs hard work in the woods by cutting and transporting trees.

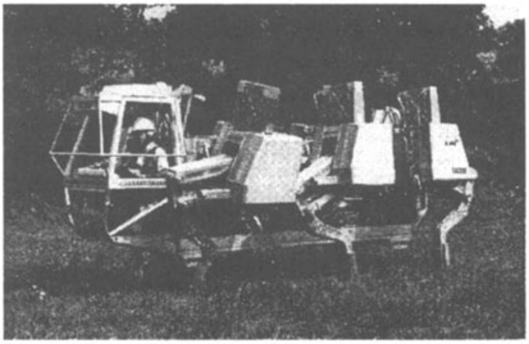


Figure 9. Adaptive-Suspension-Vehicle, Six-Legged-Machine

Other Six-Legged Robots

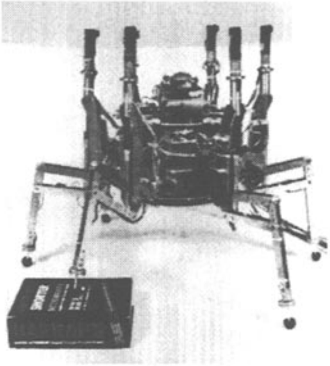


Figure 10. SILEX, Six-Legged-Machine

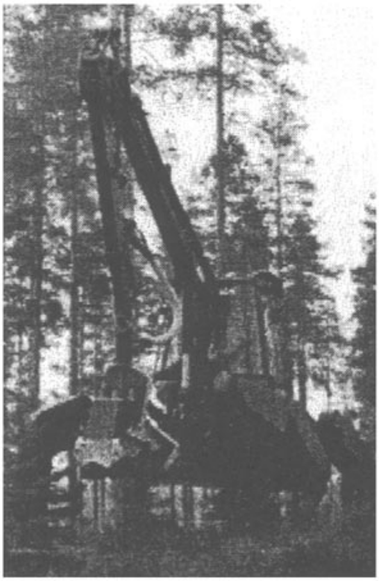


Figure 11. PLUSTECH, Six-Legged-Machine

- Ambler (1988)
  Carnegie Mellon Univ. Pa, USA
  (3.5 x 5 x 4.5 m, 2000 kg, 1.9 kW)

- Boadicea (1996)
  MIT, Massachusetts, USA
  (4.8 kg)

- CWRU ROBOT II (1992)
  Case Western Reserve Univ., Ohio, USA
  (0.76 x 0.2 x 0.46 m)

- Daedalus (1993)

Carnegie Mellon Univ., Pa, USA
(1.8 x 2 x 1.8 m, 120 kg, 100 W)

- Dante II (1993)
  Carnegie Mellon Univ., Pa, USA
  (2.4 x 3 x 3.6 m, 770 kg, 1 kW)

- TARRY (1992)
  Gerhard-Mercator-Univ., Duisburg, Germany
  (0.4 x 0.15 x 0.3 m, 2.2 kg, 30 W)

- HEG 1060
  California Cybernetics Comp., USA
  (0.3 x 0.3 m, 37 kg)

- Robot hexapode (1989)
  Lab. de Microinformatique, Lausanne, Switzerland
  (0.3 x 0.13 x 0.08 m, 0.4 kg, 5 W)

- S1 (1992)
  Univ. of Leipzig, Germany
  (1.8 x 0.5 x 0.9 m, 20 kg, 100 W)

- Roboty (1995)
  Taygeta Scientific Inc., Monterey, Ca, USA
  (0.38 x 0.2 x 0,18 m)

- IOAN (1994)
  ULB Service des Constr. Mech. and Robotique, Bruxelles, Belgium
  (0.4 x 0.15 x 0.1 m, 1.2 kg, 16 W)

- KARLA (1992)
  FZI, Karlsruhe, Germany
  (0.7 x 0.4 x 0.4 m, 4 kg, 25 W)

- LAURON II (1995)
  FZI, Karlsruhe, Germany
  (0.7 x 0.3 x 0.7 m, 16 kg, 70 W)

- Katharina (1994)
  IFF, Magdeburg, Garmany
  (0.9 x 0.5 x 0.9 m, 21 kg, 130 W)

- Autonomous Modular Walking Vehicle (1996)
  Univ. of Waterloo, Ontario, Canada
  (0.61 x 0.3 x 0.46 m, 4.5 kg)

- Leonard (1994)
  Lab. de Microinformatique, Lausanne, Switzerland
  (0.24 x 0.2 x 0.2 m, 1 kg, 4 W)

- MASCHA (1971)
  Univ. of Lomonnossow, Moscow, Russia
  (0.7 x 0.21 x 0.7 m, 20 kg, 130 W)
- MECANT (1988)
  Helsinki Univ. of Technology, Finland
  (2.1 x 1.6 x 2.4 m, 1100 kg, 35 kW)
- TUM-Walking-Machine (1992)
  Techn. Univ. of Munich, Germany
  (0.8 x 0.4 x 0.8. 23 kg, 300 W)

2.4. Machine Performance

The rough overview illustrates the fact that six-legged robots have been realized to a significantly larger extent than two- or four-legged machines. The reasons have already been mentioned: easy "static" walking, no problems with stability, well-known gait patterns, control realization with reasonable effort. Two-legged walking requires dynamic stability generating quite a lot of yet unsolved problems. Four-legged walking lies in-between, it still can be realized more or less statically which for most existing machines has been done.

Without exceptions the discussed machines ly in the "massive areas" with respect to their weight-to-size and weight-to-power performance, indicating the necessity of research for better design. Lightweight construction optimization will be one point already achieved in a few machines, today's drives and power-transmissions still need progress to be able to meet the biological example. It is interesting though that Waseda's WABIAN is very near to a human, and the two heavy six-legged machines are very near to the elephant. Obiously the technical components of today are better suited to realize medium- or large-sized robots than very small ones which mainly indicates the shortcomings of now-a-day's drives.

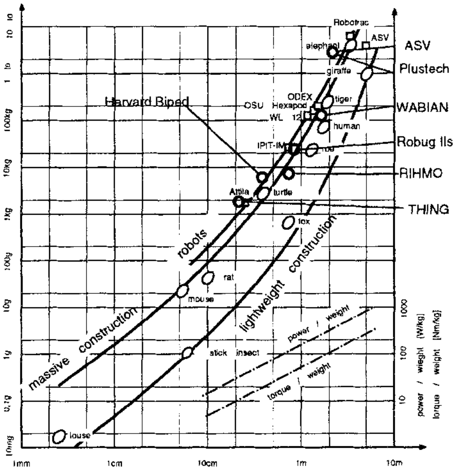


Figure 12. Weight-to-Size Performance

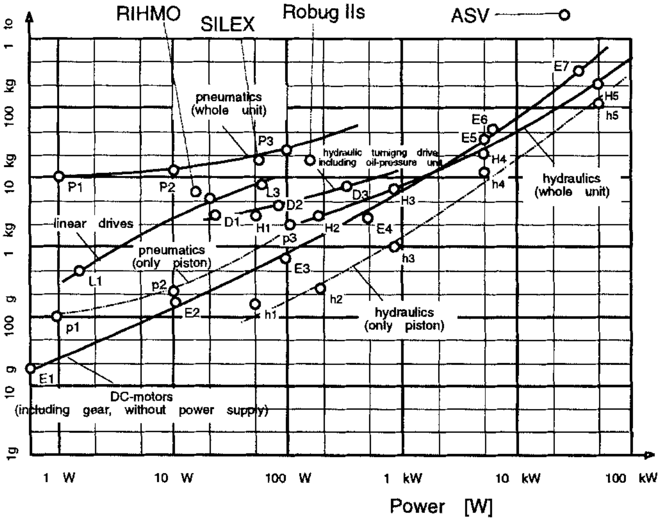


Figure 13. Weight-to-Power Performance for Technical Walking Machines

3. Design Principles for Walking Machines

In simulating, designing and realizing walking machines, and, as a matter of fact, many other machines, it will make sense, to watch certain rules and sequences of events to assure a successful development. This must not be as perfect as industrial program management, but it helps to establish an efficient research progress and convincing results [14].

The overall organization follows Figure 14, which contains three main blocks: layout and design, hardware-in-the-loop tests and finally the real machine. All steps for realizing such a walking machine are connected with tests and evaluations with respect to performance according to the basic ideas and requirements, which come so-to-say from first creative intuitions and goals, with respect to confirmations of the goals and the requirements, and finally with respect to possible improvements.

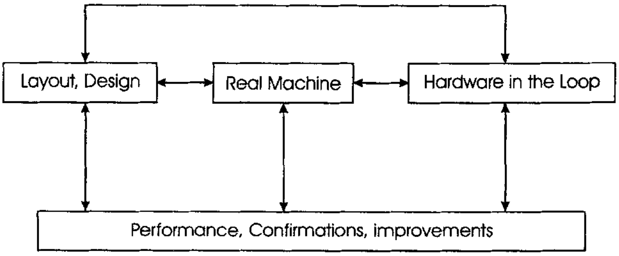


Figure 14. Realizing a Walking Machine

Considering walking machines the intuitive phase of creating ideas and concepts very much was interconnected with research findings of biologists. They know about mechanics of walking, and they know quite remarkable structures of biological walking control. At the very beginning we were of course aware of the fact that it makes no sense to copy biology. But on the other hand we are convinced that a good combination of biological design ideas and technological possibilities will make a good walking machine.

Therefore, in an early step we started to design a six-legged machine oriented at features of the stick-insect. Layout and design includes on a software basis everything of the future machine: concept and configuration, dynamics and control, gait pattern and stability; simulations again and again to find out the best performance, the best load-to-weight-ratio, the best components from the mechanical side and from electronics. One has to perform trade-offs for sensors and actuators, for design configurations and for form and materials. The selection of commercial components from external sources follows optimization processes with criteria like load-to-weight, stiffness, stability.

Figure 15 illustrates this research phase. At the end we have a rather complete impression how the machine will operate, what are the difficulties, where do we have to expect the biggest problems.

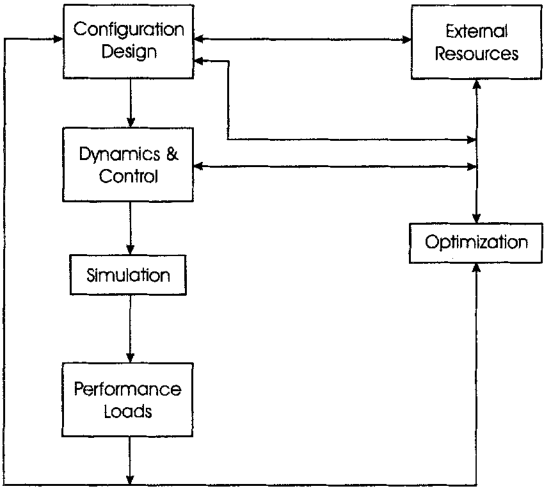


Figure 15. Layout and Design

At this stage it make very much sense to prepare hardware-in-the-loop tests (Fig. 16). For a walking machine with the above preparations it is more or less straightforward. In simulations we make the machine walk in the computer with the exception of one leg. This one leg we realize in hardware, establish a test set-up and combine the motion of this one leg in hardware with the simulated motion of the other five legs on the computer.

As a consequence we are able to operate the complete machine, one leg in reality, five legs in simulations or, in the case of the eight-legged tube crawler, one leg in reality and seven legs simulated. We can test the performance and the control of one real leg and of the complete machine and find out most of the errors and drawbacks made during layout and design. Especially control performance can be tested, control hardware can be adapted and problems

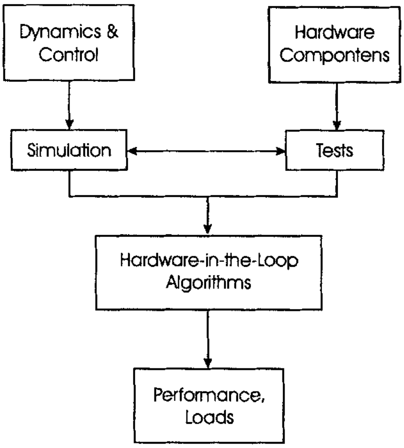


Figure 16. Hardware-in-the-Loop Tests

like EMC (electro-magnetic compatibility) can be considered. Altogether the two stages layout, design and hardware-in-the-loop tests clear up most of the problems beforehand. The final machine realization looses much of its risks.

On the basis of these beforehand-activities the last step of constructing the real walking-machine starts with a set of drawings for both, the mechanical and electronic workshop. Questions of manufacturing, external ressources, integration and assembly have to be answered and handled. From virtual sensors and virtual actuators we proceed to real sensors and actuators. Machine performance will be compared with the design idea, and the whole process of machine testing can be started (Fig. 17).

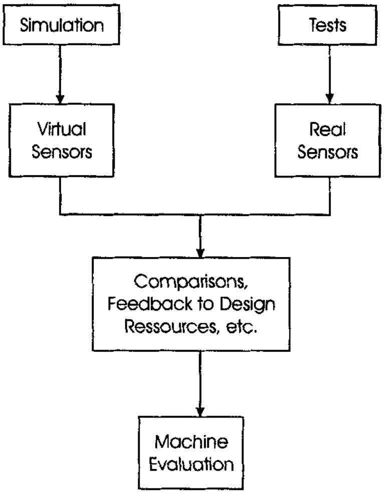


Figure 17. Machine Realization

4. Two Examples

At the author's institute two walking machines have been realized, one six-legged machine with biologically oriented design principles and one eight-legged mchine for tube inspection and the like. A short description is given. For more details see [5,17,23], which include also the theoretical background for walking machines.

4.1. A Six-Legged Machine

The basic geometry of a leg is taken from the stick insect Carausius Morosus [2,3,4]. Figure 18 gives an indication of the walking machine and of the specific leg design as derived from biology. All leg components move in one plane which itself rotates around an oblique axis fixed at the central body. The orientation of this axis is given by the two angle (ψ, φ), the rotation around it by the angle α. The two leg segmentts (for the stick insect *Femur* and *Tibia*) move in the plane with the two rotation angles (β, γ).

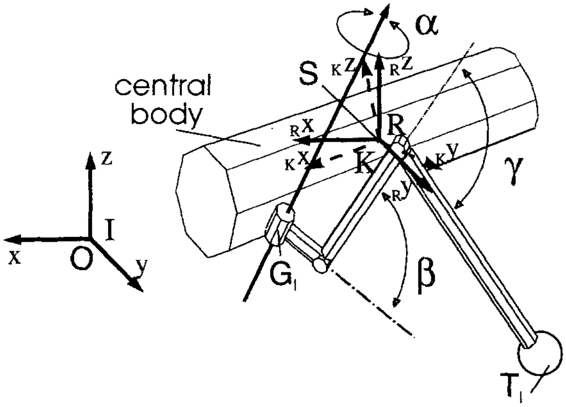


Figure 18. Six-Legged Walking Machine and Details of Leg Design

The coordination of the six legs during walking is mainly laid down in a gait pattern system. Our biological model, the stick insect, applies for walking two patterns, namely the tetrapod gait and the tripod gait. The first one more for low and the second one for higher speeds. Gait patterns are characterized by three parameters *pi*, *pc*, *df*. The ipsilateral phase *pi* is the normalized phase lag between two neighbouring legs on the same side of the body. The contralateral phase *pc* marks the normalized phase lag between a leg and its opposite neighbouring leg. The duty factor *df* defines the portion of contact time with the ground related to total walking cycle time of one leg. One gait pattern can conveniently be characterized by one point in the (*pi*, *pc*, *df*)-space (Fig. 19). In addition this type of presentation is a good basis for stability analysis for different gaits.

The evaluation of design criteria as applied in biological systems greatly profits from the fact that with a prescribed gait pattern the determination of the torques in the leg joints and the forces on the ground leads to an under-determined system of equations of motion. To make this system solvable one may apply an additional optimization process which gives an opportunity to test several criteria and to adapt these criteria to measured forces from the stick insect [5].

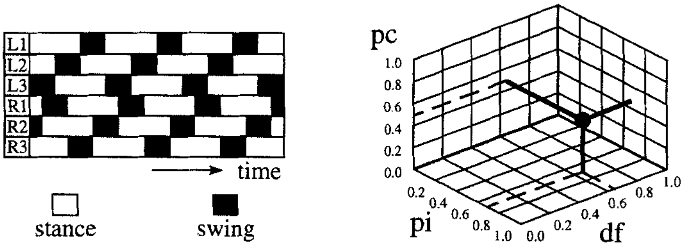


Figure 19. The Tetrapod Gait

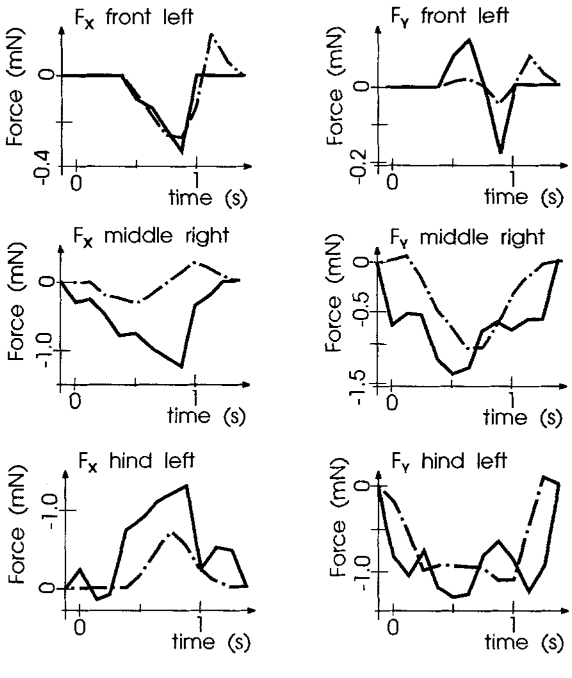


Figure 20. Measured and Calculated Contact Forces in the Walking Plane

The walking stick insect is modelled as a multibody system with rigid components and altogether 24 degrees of freedom (one central body with 6 DOF, six legs each with 3 DOF). Depending on the walking state a maximum of 36 forces at the feet and torques in the joints are unknown. Adding to the equations of motion an optimization process with unknown criteria we are able to find out the biological design principles by testing various criteria [5,16]. The resulting forces and torques are then compared with biological measurements. A best fit was achieved with a criterion requiring minimization of the legs' bending loads and of the interaction forces in the walking plane with a relationship of (60 %) : (40 %). Figure 20 depicts a comparison for that case.

The central components of every walking machine are the legs. Therefore, walking machine design means leg design. Moreover, leg design means design of its most important joint, the β-joint (see Fig. 18). Figure 21 illustrates the basic design features. The main idea of such a direct drive system consists in a direct connection of the flexspline with the driven casing, which results in a very small weight of the complete drive system [5,10]. Another important feature with regard to weight reduction is the shortest possible distance between the two main axes of the α- and β-joint, which could only be realized by the dense construction of the β-combination of direct current motor and harmonic drive system. In addition and for realizing a relative angle of 70° between segment 1 containing the α- and β-joints (Figs. 18, 21) and the second segment an excentric fastening of segment two to the β-axis was chosen as an optimum solution (Fig. 22).

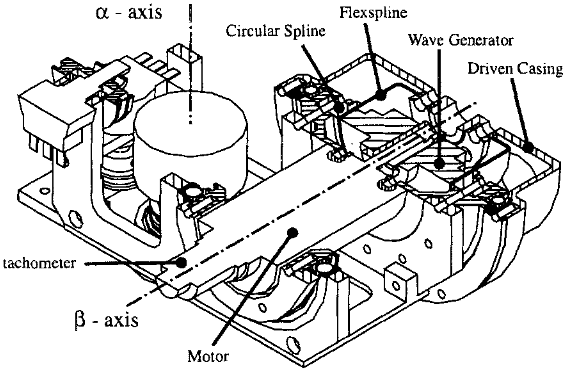


Figure 21. Design of the β-Joint

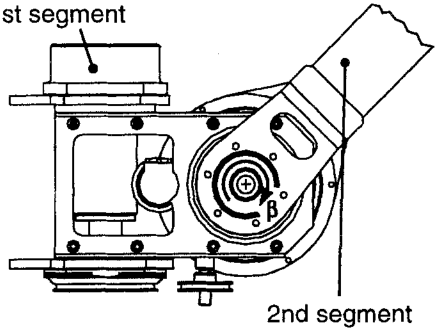


Figure 22. Excentric Fastening of the second Leg Segment

In further reduction any component of the leg construction to its minimum

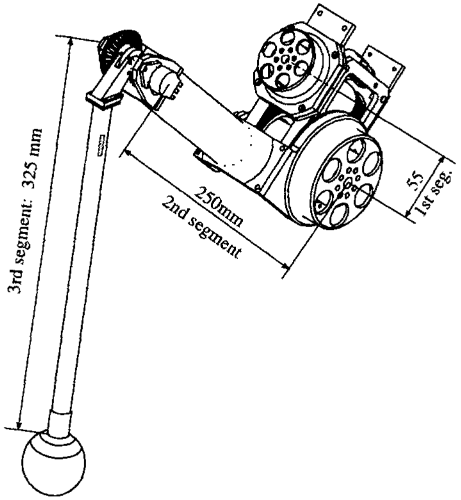


Figure 23. Leg with 3 Degrees of Freedom (2.8 kg)

shape taking into consideration the stress limits of stability we could perform a leg with a total weight of together 2.8 kg but a load bearing capacity of nearly 18 kg at the most (Fig. 23). The six-legged walking machine has three equal legs on each side where the legs from one side to the other are mirror symmetric. Each leg has the same direct drive systems. In addition to the tachometer measurement all joints are equipped with potentiometers for angular measurements. For the joints (α, β) these potentiometers are driven by pulleys, for the γ-joint it is driven directly.

The weight of the complete walking machine comes out with 23 kg, about 17.5 kg for the six legs including special fastenings, 1.5 kg for the central body and about 3.5 kg for the electronic and electric equipment. A specially designed fastening component for each leg allows an adaptation of the α-axis orientation to different walking requirements in a very wide range, which will be systematically tested experimentally. Figure 24 gives an idea of the walking machine design.

The main ideas for the control system of our six-legged robot were taken from neurobiological research with stick insects [2,3,4]. The technical realization follows in its performance very closely the biological behaviour. A global leg coordination module (LCM) is an information level where each leg informs its neighbouring legs about its state as laid down in the decision functions of each single leg controller (SLC) [23].

The leg coordination module (LCM) is responsible for setting the landing and lifting points of each leg (In the following AEP = *anterior extreme position* and PEP = *posterior extreme position*). In controlling these points the global behaviour of the walking process can be influenced. Although this level is doing a global task, the control mechanism works locally. In Figure 25 this mechanism is depicted. In this Figure it can be seen, that neighbouring legs can shift the AEPs and PEPs by a small amount. Thus e.g. legs can inhibit

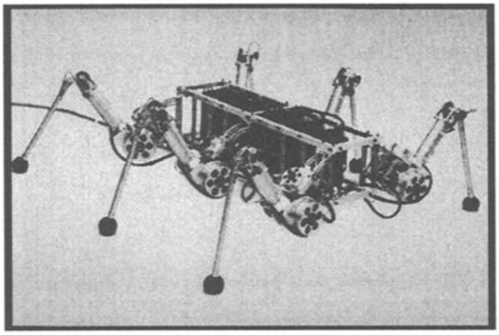


Figure 24. The Six-Legged TUM-Walking Machine

adjoining legs from lifting of the ground in postponing their PEPs. Each leg gets specific information from the other legs namely the walking phase, the velocity and the AEP and PEP-Values. This information is sufficient for each LCM to compute its new AEP and PEP. These values are sent to the middle control level. There is no central supervision.

The control influences used in this apprach have been measured and isolated by neurobiologists. Up to eight control mechanisms can be implemented in the LCM, the function principle of two of the most important mechanisms numbered I and II are shortly explained in the following: Given that the rostrally neighboured leg is not yet in STANCE phase, the mechanism I inhibits the lifting of the leg in shifting back the PEP by a certain increment. Similarly, the mechanism II inhibits a start of the lifting phase when the contralateral adjacent leg is not yet back in STANCE phase.

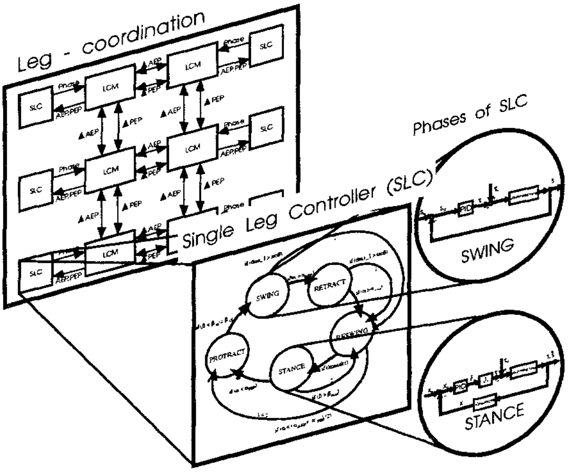


Figure 25. Control Concept of the Six-Legged Walking Machine

The single leg controller (SLC) is the heart of leg motion performing all decisions necessary to move the leg and to control the various phases. The order of the different phases in a normal step is STANCE, PROTRACT, SWING and RETRACT (see Fig. 25). The SLC switches between the phases in dependency of the AEP, the PEP and some specific events (e.g. hitting an obstacle). It does some on-line path planning at the beginning of the PROTRACT phase. Moreover the SLC gives to each leg some local intelligence especially needed to manage obstacles, impacts or other unforeseen events.

The single leg controller detects and surpasses obstacles, controls body height and corrects slippage effects. The capability of obstacle avoidance is achieved by means of a special detection mechanism and a different approach to general path planning. During SWING phase the SLC monitors the bending load in the leg segments. Whenever the corresponding strain gauge signal exceeds a certain threshold value the obstacle avoidance mechanism is activated. A short RESWING phase is executed followed by a new SWING phase trying to pass the obstacle.

The path planning algorithm for the three leg angles α, β, γ thereby differs from standard path planning used in robotics. Usually, end effector trajectories are described by time histories of work space or configuration space coordinates. In our approach we describe the dependency of the outer joint coordinates β, γ in terms of the leg angle coordinate α.

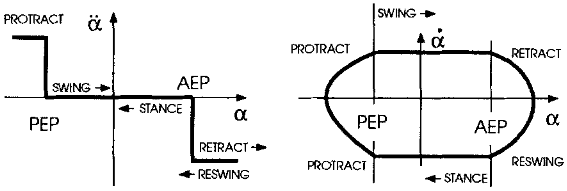


Figure 26. Three Step Controller for the α-Joint (Leg Plane)

In addition to the two upper levels the leg needs a lowest level control system which typically, and again near to biological performance, consists in a feedforward nonlinear decoupling scheme combined with a feedback linear controller. The low level controller for the AIR phase (which includes PROTRACT, SWING, RETRACT and RESWING) resembles a manipulator controller with on-line path planning. The controllers for the AIR and STANCE phases differ in the controlled coordinates.

During the STANCE phase the leg is in an active support phase and is controlled in cartesian coordinates. In the AIR phase the leg angles are controlled. The acceleration $\ddot{\alpha}$ is given by a three step controller approximating thus the biological behaviour of the controlling neurons (see Fig. 26). The angles β and γ are computed at every step from the momentary angle α. These two angles are controlled by a linear PD-controller. SWING marks the return movement of the leg to the next ground point and PROTRACT and RETRACT/RESWING denote the high acceleration transition areas from status STANCE to SWING or vice versa, respectively. We furthermore demand piecewise constant angular accelerations which are switched at the anterior extreme position (AEP) and the posterior extreme position (PEP). Fig. 26 shows acceleration versus angle and the corresponding phase portrait of the swing movement of the leg plane. The acceleration of the angle α in the STANCE phase is not exactly zero, because it results from the kinematics of the robot central body due to the switching in a cartesian system.

4.2. A Tube Crawling Robot

Tube systems differ in their pipe diameters, lengths, the mediums inside, the complexity of the tube arrangement etc. Different kinds of robots have been developed for inspecting and repairing tubes from inside [12,13]. They are driven by wheels or chains or they float with the medium. All types of robots have their specific difficulties, for example problems of traction or low flexibility and do not satisfy all requirements expected by the users. The aim of this project is the development of a robot moving forward by feet to study the possibilities and difficulties of legged locomotions in contrast to other systems. The higher flexibility of legs can be used to extend the technical possibilities of moving in tube systms (Fig. 27).

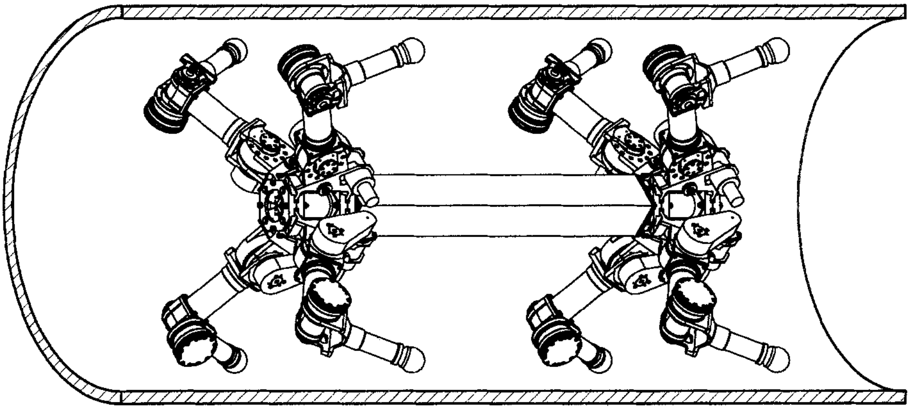


Figure 27. Construction of the Pipe Crawling Robot

The robot shown in Figure 27 has eight legs arranged like two stars. The attachments of the eight legs are located in two planes that intersect at the longitudinal axis of the central body. These planes are called leg planes. Each leg has two active joints, which are driven by DC-motors. Their axes of rotation are orthogonal to the leg planes. This provides each leg with a full planar mobility. The leg is mounted to the central body with an additional passive joint, which allow small compensating movements in the third direction.

The crawler has a length of about 0.75 m and is able to work in pipes with a diameter of 60 – 70 cm. In each of the eight legs, the distance between the two active joints (hip and knee) is 15 cm and the length of the last leg segment (from knee to foot) is 17 cm. The highest possible torque of the hip joint is 78 Nm short term and 40 Nm permanent. The corresponding values of the knee are 78 Nm and 20 Nm. In a stretched out position a leg is able to carry 6.5 times its own weight (less than 2 kg) permanently and 12 times for short time operations. Its mechanical design is based on the six legged walking machine.

The total weight of the crawler is about 20 kg including the electronic parts.

The robot is controlled by five Siemens microcontrollers 80C167 CAN, which are installed on the crawler itself. One controller acts as a central unit. Each of the remaining four units controls two opposite legs. The controllers are able to communicate over a CAN bus system.

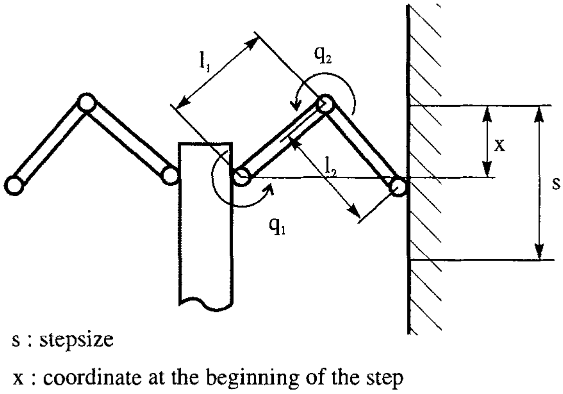


Figure 28. Kinematics in the upper Leg Plane

Each leg has two potentiometers to measure the joint angles and two tachometer generators to measure the angular velocity of the motors. For measuring the contact forces to the pipe a special lightweight sensor was developed. With its five axes it does not depend on the exact contact configuration. For future extensions the electronic architecture allows the implementation of further sensors like inclination meters.

An optimization with respect to leg geometry and stiction forces at the feet has been performed with the goal of better design (see Fig. 28). This optimization was computed for different sets of parameters e.g. tube diameters or friction coefficients. It is not useful to discuss the different results in more detail. Some aspects about the general behaviour of $F_{\max}$ are [18]:

- For each fixed leg position, the maximum friction force $F_{\max}$ does not increase with l_2.

- As the leg position changes from the fore to the rear extreme position, for a fixed l_2, the force $F_{\max}$ varies nonmonotonically. Typically, it initially increases, then passes a local maximum and decreases, and then passes a local minimum and increases again. As μ and the clearance grow, the local maximum tends to move towards the rear extreme position of the foot. For comparatively small μ the local maximum of $F_{\max}$ is its global maximum. As μ increases, the situation changes, and the global maximum is reached at the rear extreme position.

- For high friction coefficients and large clearances, the rate of the growth of $F_{\max}$ during the step considerably exceeds the rate of the decrease of $F_{\max}$ with l_2. This leads to the following result: if the second link becomes longer, it is possible to shift it backwards and thus to yield higher $F_{\max}$. Hence, the elongation of the leg's second link is advisable if the robot is intended for motion inside tubes of large diameter with high μ. This is true for gas pipe-lines where lubrication of the surface is absent. If the robot is designed for oil pipe-lines, where the tube surface is lubricated, another choise of the length of the second link can turn out to be most rational.

The presented control structure enables the robot to move in straight and curved pipes independently of the position inside the tube or the inclination of the tube (from horizontal up to vertical pipes). Considering the experiences with the six legged walking machine a structure was chosen that is divided into two hierarchical levels. The upper level encloses the mechanism of coordination. The lower level controls the position and forces (it executes operating functions). Based on this division it is possible to realize a function orientated structure and to leave the solution of problems to the concerned components.

The gait pattern influences the dependencies between the legs and thus affects the coordination and the control structure. Because of the limited leg mobility, a load shift is only feasible from the legs of one leg plane to the legs of the other leg plane. This provides the crawler with full mobility in this plane. Three dimensional movements must be approximated by acting in orthogonal spaces. In other cases the crawler is able to move straight on only (except for special contact positions).

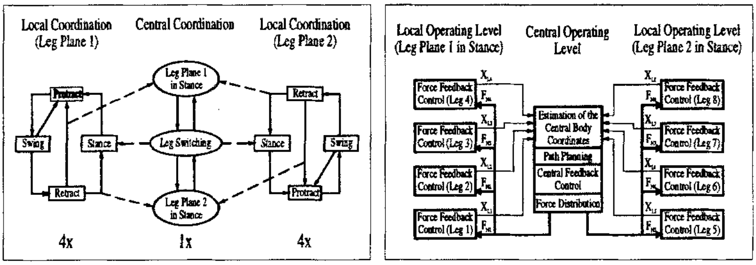

Local Coordination (Leg Plane 1)
Central Coordination
Local Coordination (Leg Plane 2)
Leg Plane 1 in Stance
Protract
Retract
Swing
Stance
Leg Switching
Stance
Swing
Retract
Protract
Leg Plane 2 in Stance
4x
1x
4x
Local Operating Level (Leg Plane 1 in Stance)
Central Operating Level
Local Operating Level (Leg Plane 2 in Stance)
Force Feedback Control (Leg 4)
Force Feedback Control (Leg 3)
Force Feedback Control (Leg 2)
Force Feedback Control (Leg 1)
X_{L4}
F_{N4}
X_{L3}
F_{N3}
X_{L2}
F_{N2}
X_{L1}
F_{N1}
Estimation of the Central Body Coordinates
Path Planning
Central Feedback Control
Force Distribution
X_{L8}
F_{N8}
X_{L7}
F_{N7}
X_{L6}
F_{N6}
X_{L5}
F_{N5}
Force Feedback Control (Leg 8)
Force Feedback Control (Leg 7)
Force Feedback Control (Leg 6)
Force Feedback Control (Leg 5)

Figure 29. Level of Coordination and Operating Level

The diagrams of Figure 29 show the principles of the coordination level and the operating level for the load phase.

- The *central coordination level* coordinates the phase characteristics of the two leg planes. Decisions on switching of the legs under load are made by this component. The legs do not have any autonomy here with the advantage of higher safety from falling. In this aspect the concept differs from other solutions [12,13]. Furthermore, the problems which can only be mastered by a reaction of the whole robot schould be solved in this level (e.g. the legs of one plane can not find any contact).

- The *local coordination level* controls the step circle of a single leg, especially the sequence of leg motion phases (stance, protract, swing, retract). It also reacts to disturbances like avoiding small obstacles.

- The *central operating level* controls the position and the velocity of the central body which are estimated from the joint angles of the legs. This is done by changing the leg forces to achieve accelerations for correcting the control errors. For this purpose the local operating level is used. It receives the corresponding setpoint commands. These commands must be created with respect to restrictions like satisfying the condition of sticking or the limitations of the electrical and mechanical components.
- The *local operating level* controls the applied forces during the contact phase and the motions of a single leg during the different air phases. In contrast to the last ones, which are really local problems (legs without contact can be assumed as decoupled), the forces of legs touching the environment are strongly coupled and therefore a strictly local realization cannot consider all effects in each configuration. Therefore local means as local as possible.

The main problem is the controller design for the load phase of a leg plane. The crawler is a system with geometrical and kinetical nonlinearities. Its several components have many degrees of freedom and are strongly coupled. In accordance with the described structure of the operating level the controller can be presented by the block diagram shown in Figure 30.

A decentral PID control of the leg forces and the central control of the crawler position was developed by using a multi model design, which is based on linearizations around several leg positions [20]. The qualification of this design was tested by simulations. Nevertheless the system behaviour of this design depends on the actual leg configuration and therefore it cannot be optimal in any case. According to this another design will be presented here, which is based on an input-output-linearization of the inner circuit [21]. The disadvantage of this method is the more complicated and more complex structure. To get system equations which can be handled without loosing the physical context the following simplifications are made, which do not change the characteristic behaviour of the system:

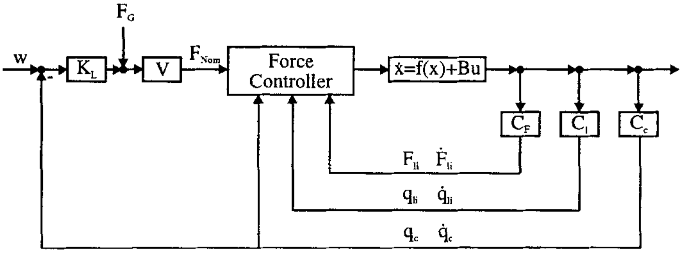


Figure 30. Block Diagram of the Operating Level

- Motions in the passive joints are not observable and not controllable by the legs of the corresponding leg plane. Therefore these motions are decoupled and must be considered in the controller design. This leads to a planar model with 11 degrees of freedom.

- The damping of the rubber balls (feet) is neglected.
- The masses of the segments are added to the central body and therefore the moments of inertia referred to the leg joints are constant and decoupled from the central body coordinates. Caused of the light weight design the influence of this simplification is less than one per cent.
- The friction in the gears will be compensated by using an observer. The compensation is assumed to be ideal and therefore friction is not considered any further.

Furthermore the central body velocity and the actual direction of gravity are assumed to be known. In reality these variables must also be determined by an observer.

A simulation program, which includes all the relevant properties of the robot, was developed. By means of this program it is possible to get informations about the system behaviour and to determine the motor power reserves. Since the elastic eigenfrequencies of the system parts are very high, a modelling as a rigid body system is sufficient. The system components are the central body, the rotors of the motors, the shafts of the gears and the segments of the legs. Different to industrial robots the stiffness of the gears is negligible for the system behaviour. The reasons are the extreme light weight design, the very short lever arms and the small moments of inertia of the segments. The friction of the Harmonic Drive Gears depending strongly on the torque has great influence on the control and on the loads of the motors (coulomb friction in meshing). For consideration of this effect, "normal torques" are established to calculate tangential friction torques that act against the direction of the rotation. To include sticking without load (effects like No-Load Starting Torque and No-Load Back Driving Torque) an initial tension of the gears is introduced. For sticking under load the transmitted torques are added to the initial tensions. In addition to the mentioned phenomena, the following ones are part of the simulation model: The contact between legs and ground is realized with a spring-damper element, which represents the rubber balls at the end of the legs. The temperatures of the motors are integrated with a two body model with unlimited caloric conductibility. With these temperatures the torque reserves of the motors can be determined, which are only limited by burning out. Furthermore the motors are changing their behaviour in a not negligible manner caused by the dependence of their coil conductivity on temperature.

For testing the mechanical design and the designed controllers a single leg test setup was built. The leg mounted on a fixed frame can walk on a conveyor-belt, which is motor driven and can be run with different velocites. The mechanical parts and the control hardware is equivalent to that one used in the robot.

For the test setup an extra simulation program is developed. The model is similar to that of the whole robot. In Figure 31 comparisons of simulations results and measurements are shown. The diagrams on the left side belong to

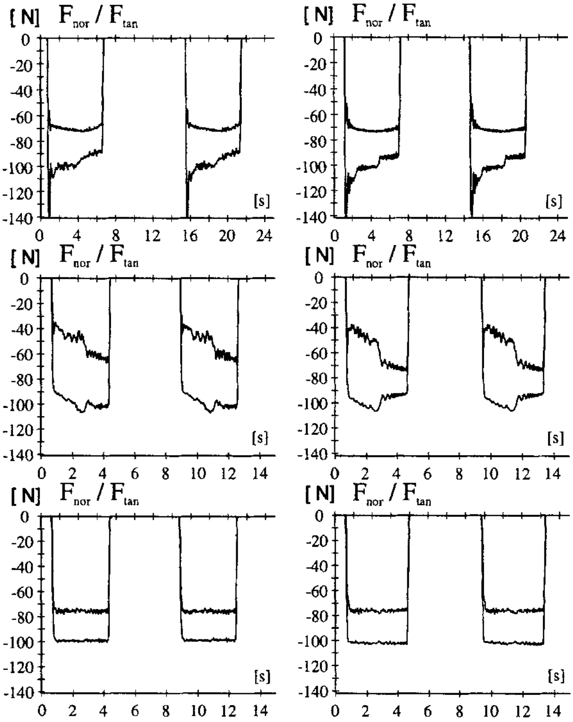


Figure 31. Comparison of Measurement and Simulation

the measurements. The two curves in the graphs correspond to the normal and tangential forces of two steps on the conveyor-belt. In each line a different controller was used. The first one shows steps at a slow speed using a PID controller.

Two undesirable properties can be seen. The first one are the high peaks at step beginning and the second the decreasing normal forces in the middle of the steps. This is caused by the gear friction in the knee joint, which changes the direction of rotation. The second and the third line use the controller based on feedback linearization. The difference is that for the third the friction observer is used. The second one is only displayed to illustrate the great influence. It can be seen the compensation works very well. The observer could be used for the PID controller also. In this case it is able to inhibit the decreasing of the force but not the peaks at the beginning. As an excerpt it can be seen that the last controller is qualified for the problem. The curves also show a very good conformity between simulation and measurement.

5. Summary

A survey of walking machines is given. Additionally two specific walking machines, a six-legged and an eight-legged one are presented. It turns out that artificial walking has made considerable progress in the last two decades, but that its perfomance is still far away from biological walking quality.

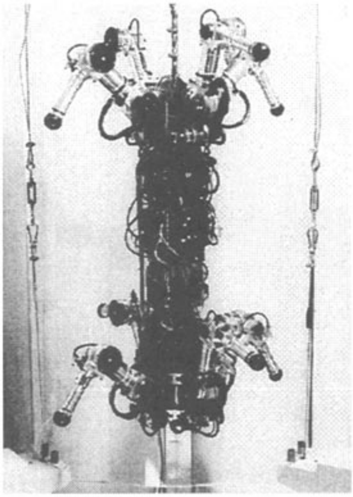


Figure 32. The Tube Crawling Machine (Mass - Length etc.)

For two special machines design and control principles are described. A six-legged machine follows closely biological design principles where especially a three-layer-control concept realizes very nicely the walking pattern of a stick insect. An eight-legged machine was realized for tube crawling operation. Its control concept realizes observers for gravity and friction and a feedback linearization for the complete system. An essential feature consists in a complex force control strategy for controlling the feet-tube wall-contacts.

General remark: More detailed informations on the walking machines as presented in chapter 2 may be called from
`http://www.fzi.de/divisions/ipt/WMC/preface/walking_machines_katalog.html`

References

[1] Bremer, H.: Dynamik und Reglung mechanischer Systeme, Teubner Verlag, Stuttgart, 1988.

[2] Cruse, H.: The Function of the Legs in the Free Walking Stick Insect, Carausius morosus, Journal of Comparative Physiology, (1976), p. 112.

[3] Cruse, H.: What mechanisms coordinate leg movement in walking arthropods?, Trends in Neurosciences 13, (1990), pp. 15-21.

[4] Cruse, H.; Dean, J.; Müller, U.; Schmitz, J.: The Stick Insect as a Walking Robot, Proc. Fifth Int. Conf. on Adv. Robotics, Robots in unstructured Environment, Pisa, Italy, June 1991, pp. 936-940.

[5] Eltze, J.: Biologisch orientierte Entwicklung einer sechsbeinigen Laufmaschine, no. 110 in Fortschrittsberichte VDI, Reihe 17, VDI-Verlag, Düsseldorf, 1994.

[6] Glocker, C.: Dynamik von Starrkörpersystemen mit Reibung und Stößen, Reihe 19, Nr. 182, VDI-Verlag, Düsseldorf, 1995.

[7] Glocker, C.; Pfeiffer, F.: Stick-Slip Phenomena and Application, Proc. of Nonlinearity & Chaos in Engineering Dynamics, Symposium, I., ed., 1993.

[8] Glocker, C.; Pfeiffer, F.: Muliple Impacts with Friction in Rigid Multibody Systems, Nonlinear Dynamics, Kluwer Academic Publishers, (1996).

[9] Graham, D.: A behavioural analysis of the temporal organisation of walking movements in the 1st instar and adult stick insect (carausius morosus), Journal of Comparative Physilogy, (1972).

[10] Harmonic Drive GmbH: Harmonic Drive Gear Component Sets, HFUC Series, Tech. Rep., Hamonic Drive GmbH, 1993.

[11] Herrndobler, M.: Entwicklung eines Rohrkrabblers mit vollständigen Detailkonstruktionen, Master's thesis, Lehrstuhl B für Mechanik, TU München, 1994.

[12] Neubauer, W.: Locomotion with Articulated Legs in Pipes or Ducts, Proc. of the Int. Conf. on Intelligent Autonomous Systems, Pitssburgh, USA, 1993, pp. 64-71.

[13] Neubauer, W.: A Spider - Like Robot that Climbes Vertically in Ducts, Proc. of the 1994 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Munich, 1994, pp. 1178-1185.

[14] Pfeiffer, F.; Roßmann, Th.; Steuer, J.: Theory and Practice of Walking Machines, in "Human and Machine Locomotion", CISM, 1997.

[15] Pfeiffer, F.; Cruse, H.: Bionik des Laufens - technische Umsetzung biologischen Wissens, Konstruktion, (1994), pp. 261-266.

[16] Pfeiffer, F.; Eltze, J.; Weidemann, H.-J.: Six-legged technical walking considering biological principles, Robotics and Autonomous Systems, (1995), pp. 223-232.

[17] Pfeiffer, F.; Eltze, J.; Weidemann, H.-J.: The TUM-Walking Machine, Intelligent Automation and Soft Computing, 1 (1995), pp. 307-323.

[18] Pfeiffer, F.; Roßmann, T.; Chernousko, F.L.; Bolotnik, N.: Optimization of Structural Parameters and Gaits of a Pipe-Crawling Robot, IUTAM Symposium on Optimization of Mechanical Systems, Bestle, D.; Schiehlen, W., eds., Kluwer Academic Publishers, 1996, pp. 231-238.

[19] Pfeiffer, F.; Weidemann, H.-J.; Danowski, P.: Dynamics of the Waling Stick Insect, Proc. of the 1990 IEEE Int. Conf. on Robotics and Automation, Cincinatti, Ohio, May 1990, pp. 1458-1463.

[20] Roßmann, T.; Pfeiffer, F.: Control and Design of a Pipe Crawling Robot, Proc. of the 13th Worl Congress of Automatic Control, I. F., ed., San Francisco, USA, 1996.

[21] Slotine, J.-J.E.; Li, W.: Applied Nonlinear Control, Prentice Hall, Englewood Cliffs, New Jersey, 1991.

[22] Waldron, K.; et al.: Force and Motion Management in Legged Locomotion, IEEE Journal of Robotics and Automation, RA-2 (1986).

[23] Weidemann, H.-J.: Dynamik und Regelung von sechsbeinigen Robotern und natürlichen Hexapoden, no. 362 in Fortschrittsberichte VDI, Reihe 8, VDI-Verlag, Düsseldorf, 1993.

[24] Weidemann, H.-J.; Eltze, J.; Pfeiffer, F.: Leg Design based on Biological Principles, Proc. of the 1993 IEEE Int. Conf. on Robotics and Automation, Atlanta, Georgia, May 1993, pp. 352-358.

Climbing Robots

Gurvinder S Virk
University of Portsmouth
Portsmouth, Hampshire, UK.
gsvirk@ee.port.ac.uk

Abstract: The paper presents an introduction to the main areas driving the development of climbing robots; the reasons for the climbers arise because many applications (including the nuclear and process industries, underwater operations, forestry work and the construction sector) require robotic intervention due to the hazardous environments encountered and because normal routes of access are not available. The status of climbing robots is presented covering the machines developed throughout the world with particular emphasis on the climbing aspects. In addition the future requirements for such mobile machines and how they can be achieved is described.

1. Introduction

Mobile robotics has received much attention in recent years with many innovative designs produced and demonstrated at exhibitions and scientific meetings. The driving forces for these machines (other than academic interest and general enthusiasm) are hazardous applications where it is either impossible (or too dangerous) to send humans to carry out particular operations of inspection, repair or a specific function, such as fire fighting or transporting material and equipment to inaccessible sites. There is a large variety of mobile robots and it is useful to classify them in some sensible way. One possible approach is to partition them by their locomotion technology as suggested in Virk [1]. Here the categories can be grouped into wheeled vehicles, tracked devices and articulated legged machines. Or indeed mobile machines can be classified into continuous or discontinuous locomotion with the discontinuous machines further split into walkers or climbers or machines which climb and walk. There are always some peculiar machines which cannot be put into the chosen categories, for example the Roobot machine developed by Dr Dissanayake at the University of Sydney has two legs and two wheels! There are other particular mechanisms which propel themselves by crawling and/or other submarine type swimming devices or special purpose designs for operation in particular environments such as in pipes or ducts (see the pipe climbing robot developed by Naubauer [2], [3] shown in Figure 1). However such examples should not stop us classifying mobile machines into some sensible grouping.

The intention of this paper is to concentrate on climbing robots so it is convenient to classify the machines into climbing or walking devices (as already mentioned, some can climb and walk!). This is especially relevant because the author has recently instigated the setting up of an EC Brite EuRam Thematic Network on Climbing and Walking Robots (CLAWAR). A six month study for this research contract has highlighted the following needs for CLAWAR type machines (see Virk [1]):

(i) **Nuclear industry:** here, there is a need for climbing and walking robots for carrying out remote operations such as non-destructive testing, surface preparation, hot spot localisation, and the retrieval of objects fallen in the reactor vessel. Clearly there is a specific need for radiation hardened components.

(ii) **Process industry:** here, climbing robots are required for the cleaning of reactors, testing the integrity of the containing vessels and for monitoring various processes. The chemical vessels may be part full, thus requiring that the machines be chemical resistant.

(iii) **Outdoor applications:** legged robots could be used for forest work (see the Forest Walking Machine shown in Figure 2), land mine clearing (see Cornelis et al [4], Nicoud [5]) and agricultural applications.

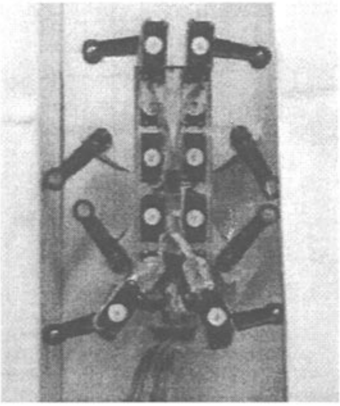


Figure 1: Pipe climbing robot

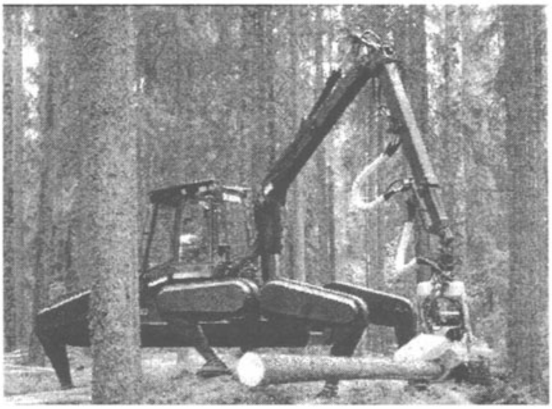


Figure 2: Plustech's Walking machine

(iv) **Construction:** climbing robots can be used for inspection, remote handling and window cleaning (see Seward [6]).

(v) **Ship cleaning:** climbing and walking machines can be used for cleaning the inside and outside of ship hulls when in dry dock (some machines can operate whilst the ship is still at sea).

All these present interesting and demanding challenges but we will be concentrating here on the climbing aspects. When we refer to a climbing robot we normally mean one which supports its weight off the ground. This means that wheeled machines which can climb stairs or negotiate rough terrain (such as the Hobo machine shown in Figure 3) is not classified as a climber. This seem sensible but some people have argued to the contrary and it is difficult to be totally prescriptive about such things. However the pipe climbing robot developed by Neubauer [2], [3], shown in Figure 1, (which can climb vertically in ducts and pipes) is easier to accommodate within the climbing group. In addition, the climbing and walking robot Robug IIs (shown in

Figure 4) developed at Portsmouth (see Luk et al [7]) clearly is a climbing machine; it can walk and transfer itself onto the wall as well as climb up tall structures.

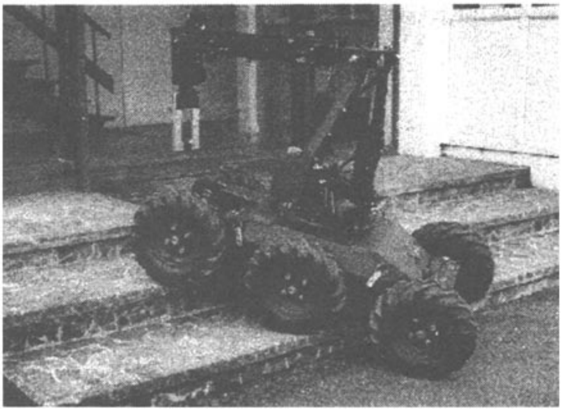


Figure 3: Kentree's Hobo machine

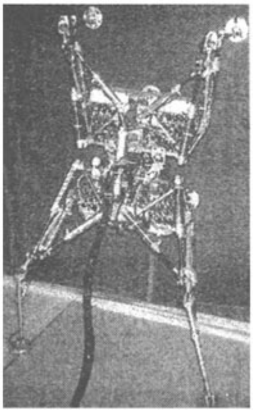


Figure 4: Robug IIs

We start our discussions by looking at the overall technologies available for designing and constructing climbing robots.

2. Technologies for Climbing Robots

The fundamental difference between a general ground-based mobile robot and a climbing machine is that the climber has to be able to sustain its weight in its operating environments. Many techniques have been developed to do this; these include:

(i) special end effectors to hang from scaffolding, girders or limbs to push against fixed structures (as for example the pipe climbing machine shown in Figure 1);

(ii) magnetic devices to attach to steel structures such as tanks and walls; and

(iii) vacuum suction pads to grip to a variety of surfaces ranging from concrete or brick walls, timber and other non-porous surfaces.

Each method has its good and not so good aspects, but high power-to-weight ratios are important to ensure that weights of the machines can be supported (see Collie [8]). There has been much work carried out in actuation technologies and how the various methods compare in terms of power/weight ratios, bandwidth limitations and individual characteristics (see Hollerbach et al [9], Colombi et al [10], Jezierski et al [11]). It is widely acknowledged that the most potent form of actuation is hydraulics, followed by pneumatics and then, lastly, electrical drives. However, hydraulic systems tend to be disliked as they are heavy, suited to larger applications and prone to leaking. Pneumatic actuators share many of the features of their hydraulic counterparts but specific design and operating differences result from

(i) the lower viscosity of air relative to hydraulic fluid (by a factor of 1000);

(ii) the higher compressibility of air (by a factor of 300); and

(iii) the poor lubrication properties of air relative to hydraulic fluid.

Practical consequences of these aspects has meant that tighter tolerances have been introduced to minimise leakage and fast-acting valves developed to counteract gas compressibility. Most pneumatic actuators involve a piston driven by pressurised gas much in the same way as in hydraulic actuators. However inflatable elastic tubes or bladders surrounded by a braided mesh that shortens with pressure have also become popular. These type of actuators are commonly referred to as pneumatic muscles and they offer the advantage of very high power-to-weight ratios (about 400:1) but their life is restricted to about 10,000-15,000 cycles before they fail (Greenhill [12]). There are other forms of actuation such as shape memory alloys and piezoelectric devices (see Hollerbach et al [9]).

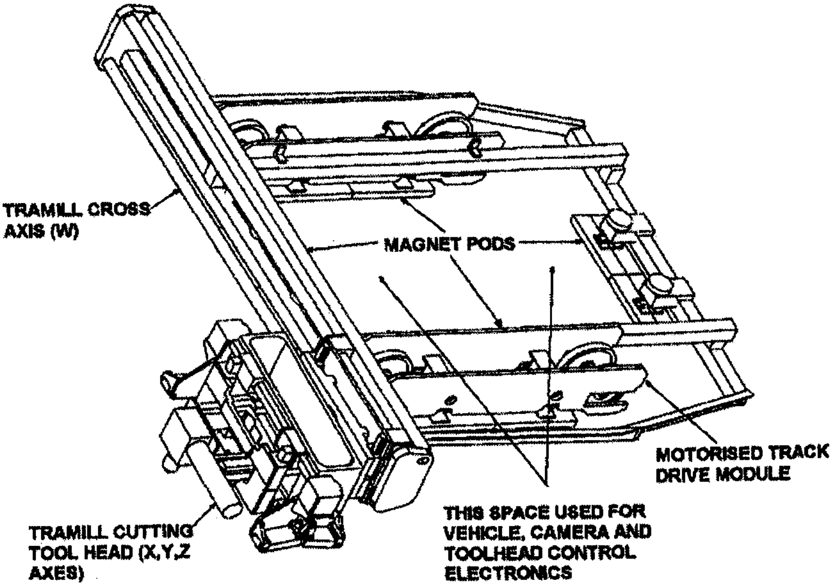

TRAMILL CROSS AXIS (W)
MAGNET PODS
MOTORISED TRACK DRIVE MODULE
TRAMILL CUTTING TOOL HEAD (X,Y,Z AXES)
THIS SPACE USED FOR VEHICLE, CAMERA AND TOOLHEAD CONTROL ELECTRONICS

Figure 5: Mavis 3 machine

The two most common forms of actuation used for climbing robots use vacuum suction and magnetic adhesion for the attachment method. Clearly the latter can only work in situations where the environment is magnetically suitable, such as nuclear installations or process plants for operation in metal tanks. Examples of such magnetic climbing machines include:

- Mavis (magnetically attached vessel inspection systems); this is a family of vehicles which has been developed by Nuclear Electric plc and Sonomatic Ltd (see Burrow and Yeomans [13]). Mavis 3 (shown in Figure 5) uses permanent magnets which do not touch the vessel to stick onto the outside of a Magnox reactor pressure vessel so as to provide work platforms to conduct a variety of tasks aimed at extending the plant life. These tasks include ultra sonic scanning, surface preparation via wire brushes and grinding and milling operations. Traction for the vehicle is effected by the use of two independent rubber belts driven by DC servos.

- Robinspec (see Fortuna et al [14]), shown in Figure 6 attaches itself to the walls of chemical reactors using three magnetic feet. This is a walking robot designed for the inspection of storage tanks in the petrochemical industry. Robinspec moves by means of its three legs (actuated by three independent DC motors), each connected to the surface with two electro-magnets that allow the robot to operate on vertical surfaces or upside down.

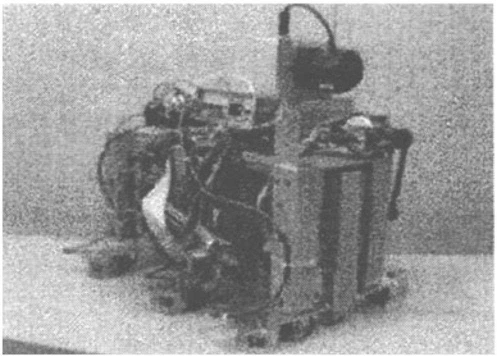


Figure 6: Robinspec magnetic robot

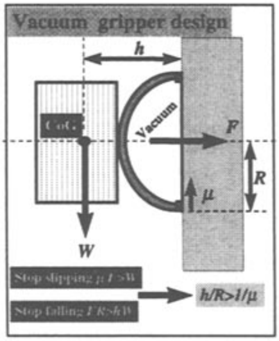


Figure 7: Vacuum pad design

The vacuum suction technique is the most potent form of gripping technique and one which has been applied by a number of researchers for various applications, including nuclear, construction and ship cleaning. The workings of these vacuum grippers is quite simple to understand and appreciate in that the negative pressure created under the foot holds it onto the wall as shown in Figure 7. The magnitude of this force (***F***) towards the wall is given by the product of the vacuum pressure (***P***) and the gripper area (***A***). The conditions to avoid slipping and falling are given by $\frac{\mu F}{W} > 1$, and $\frac{F}{W} > \frac{h}{R}$ respectively, where ***W*** is the dead weight of the robot, μ is the frictional coefficient, ***h*** is the distance from the wall surface to the centre of gravity and ***R*** is the distance from the centre of the gripper to the lower support point. If the robot is designed under the condition $\frac{h}{R} > \frac{1}{\mu}$, falling can be avoided. The method can work on fairly rough surfaces by using appropriate seals on the gripping pads so that good vacuums can be created. The two main methods of creating the suction are to create a higher negative pressure by using a vacuum pump or to produce a lower negative pressure by using a blowing device. With leakage problems in practical situations the design must ensure that sufficient negative pressure is maintained to support the robot as it moves even when rough surfaces are being negotiated. It is possible to improve these leakage problems by using more than one vacuum chamber so that the differential pressures and air losses are limited giving rise to improved performances.

Most climbing machines to date have tended to concentrate on the mechanical aspects so that the mechanical body and moving linkages are optimised, but the sensors and decision-making have been largely neglected in the industrial robots developed - in fact, most mobile robots fall in this category as well! The reason for this has been that most industrial mobile machines have a user in the loop who has been provided with sensor information such as CCD images and other range data and environmental conditions to ensure that the operator can pilot the machine in some sensible way. Various tele-operation systems and virtual environments have been designed but now the emphasis is moving on to giving a degree of autonomy to the machines. Such capabilities require significant investment and research in the area of AI and autonomous decision making and are covered elsewhere in these proceedings.

3. Applications

The main applications to date for climbing machines have been the construction, nuclear and process industries and ship cleaning. The process industry applications have concentrated upon using magnetic devices to attach to the reactor vessel and using the machine as a platform for carrying sensors an manipulator devices for cleaning operations. Tasks such as non-destructive testing and chemical analysis have been carried out. The nuclear machines have been numerous and address different scenarios encountered in the normal operation of a nuclear installation. This ranges from refuelling, retrieval of objects fallen in the reactor vessel and maintenance operations. In addition, the EC TELEMAN programme has specifically addressed different scenarios. One project funded under this initiative was to develop Robug III which was aimed at addressing a Chernobyl-type disaster and how a lightweight mobile machine could be used both for retrieving nuclear material as well as rescuing victims in such disaster scenarios. Robug III has been developed under the TELEMAN 44 project by the Portsmouth consortium led by Portech Ltd and involved several partners from Europe.

Clearly all the machines designed for nuclear applications need to be radiation-hardened and must be easy-to-clean by, say, being able to be hosed down after entering a radio-active environment. The machines need to be able to operate in unstructured environments so they need to be able to walk on rough terrains but when normal passages are destroyed they need to be able to climb vertical surfaces and make various plane transfers such as floor-to-wall, wall-to-roof, wall-to-ceiling, as well as internal and external wall-to-wall. Robug IIs and III machines developed by the Portsmouth group has been designed to address these issues.

The Ninja climbing machine described in section 4 has also been designed to perform some of these plane transfers which is felt to be an important capability for climbing machines generally. In fact there are many questions which need to be addressed when designing a climbing machine so that the required capabilities can be included in the formal specifications for the vehicle; the example machines discussed in section 4 will give an indication of these but the important points are stated below in itemised form.

(i) Are plane transitions needed to be carried out?

(ii) Is walking and climbing required?

(iii) Will sliding mechanisms suffice or are articulated legs needed?

(iv) Is speed of the essence or will be slow operational robot be adequate?

(v) Are special environments to be encountered?

(vi) What is the level of autonomy needed in the operation of the robot?

The list can go on but it is useful to turn next to actual machines which have been designed and built because this is the best way of illustrating the possibilities. By being aware of what has been achieved it is then easier to make modifications to allow different application specific climbing robots to be designed and constructed.

4. Machines Developed

The climbing machines developed to date have been numerous and only a few can be included in a paper of this kind. The main activity has been in Japan and Europe and some of the leading machines will be described here. The first of these is the Large Sucker robot developed by Nishi [15]. The robot has a large vacuum gripper, tracks for locomotion and is shown in Figure 8. The vacuum is created by using a fan to suck air from under the foot to create the negative pressure which holds the machine onto the wall when it is climbing. Similar machines to this include a simple suction device called Big Foot at the University of Portsmouth. These designs are basically inverted hovercrafts which suck rather than blow. By providing a locomotion facility such as wheels or tracks under the skirt, these machines can travel vertically on most construction materials and even on fairly roughly pointed brickwork. By using very compliant seals small ledges can be negotiated and such a machine is able to travel up vertical sheets of glass and window cleaning is one application where such devices could be used.

Another climbing machine developed by Nishi [16] is the biped machine shown in Figure 9. This is also based on vacuum suction grippers but has the ability to make plane transfers using its hinged ankles and legs in an optimised manner to minimise the moment on the fixed gripper. The biped has no specific operation in mind but it is reasonably straightforward to insert a manipulator and/or additional equipment on it for remote sensing and inspection purposes such as those required in the construction industry.

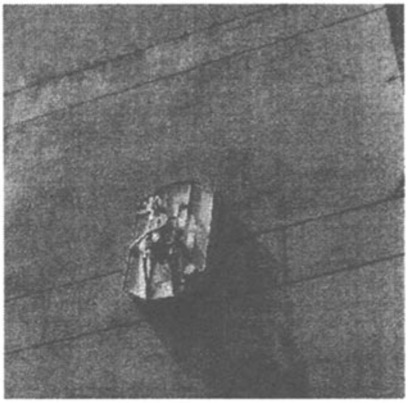


Figure 8: Big sucker robot

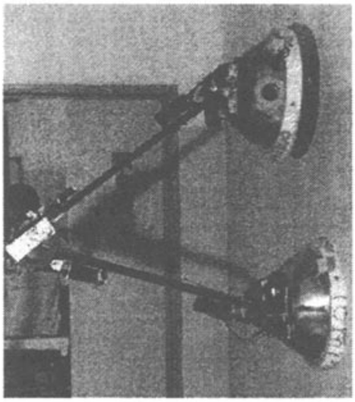


Figure 9: Biped walking robot

Nishi and Miyagi [17], [18] have also enhanced these designs to speedup the access times required to climb up buildings by producing a Wall Driving Robot as well as a Flight Robot which has been designed to fly over trees and other obstacles and then attach itself to a building and start climbing for emergency type applications such as fire fighting. Other machines have also been developed; these include the Ninja machine developed by Professor Hirose in Japan (see Figure 10), and the Robicen developed by Professor Serna at the University of Navarra, Spain. Information on these and other machines can be found in a recent report prepared by the author (see Virk [1]), as well as on the world wide web in the Walking and Climbing Machines catalogue set up by Dr Berns at Karlsruhe, Germany [19].

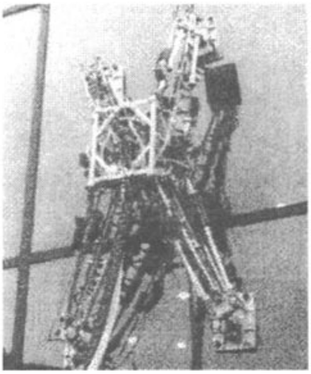


Figure 10: Ninja climbing robot

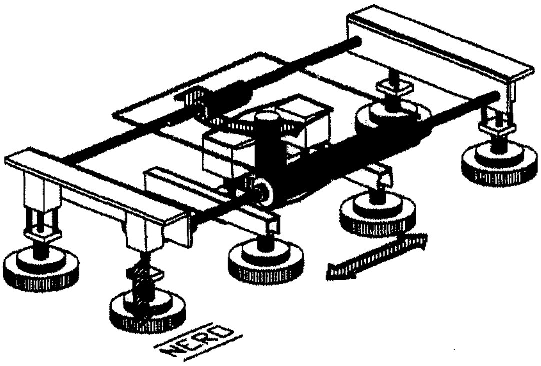


Figure 11: Nero's sliding chassis design

Several interesting machines have been developed under the leadership of Arthur Collie and John Billingsley at the University of Portsmouth in conjunction with industrial partners (Portech Ltd and Nuclear Electric plc). These include:

Toad (see Billingsley et al [20]): This is a simple mechanism designed to demonstrate walking on ceilings. It can be extended to include a spraying system so that difficult tasks such as painting of ceilings can be carried out by this device.

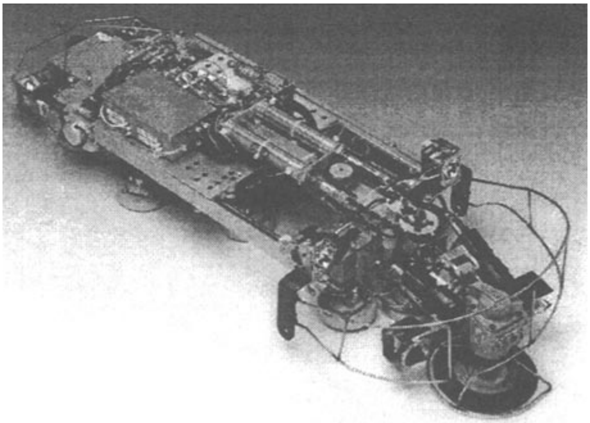


Figure 12: Nuclear Electric's Nero III climbing robot

The Nero series vehicles (Nuclear Electric Robot Operator): These are designed using a sliding chassis design shown in Figure 11, which can negotiate difficult climbing surfaces such as dusty reactor vessels in nuclear applications (see Luk et al [21], [22]). Nero I carries a tape feeder which has been used to install pulley systems for hoisting equipment onto the nuclear pressure vessel. Nero II has been designed to carry a rotary brush and vacuum system to clean and carry away the debris. Nero III (shown in Figure 12) has been designed to include an air-driven angle grinder which can cut through steel bolt heads.

Robug IIs (Luk et al [7]): The robot, shown in Figure 4, was designed because the Nero machines exposed the need for a self-launching capability. Robug IIs has an articulated body and four legs. Each leg has a vacuum gripper foot designed along the lines discussed in section 2 and the body has three further grippers all of which are used in the climbing operation.

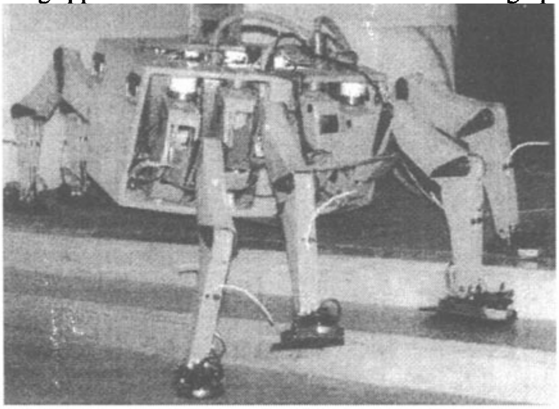


Figure 13: University of Portsmouth's Robug III

Robug III (Luk et al [23]): This is the latest machine designed by the group. The machine has been funded under the EC Teleman program and its specifications have been formulated by a user group comprising Nuclear Electric, Electricite de France, European Authority for Nuclear Research and the Italian Electricity Board. These specifications include the ability to perform plane transitions, drag a 100 Kg payload while climbing, carry a payload of 25 Kg, walk through narrow ducts and be able to operate in unstructured environments. The machine is shown in Figure 13.

5. Future

To date most robotic systems have been developed along a one-off prototyping basis. This has involved significant R&D costs for each individual project and there has been little opportunity for cross-fertilisation between the projects. Consequently there has been much re-invention of technology already developed and considerable wastage of energy. The current financial climate for R&D is getting difficult and as a result, such prototyping projects are becoming fewer and far between. Most research groups are therefore turning to largely simulation studies or producing small-scale robotic devices which have little industrial application. It is clear that mobile machines will continue to be developed along the dual path of academic research and application specific needs. However, the development cost in producing one-off machines is enormous and is only affordable by just one or two application areas where there is no other option.

However, having developed the core technology, it is felt that other "less well off areas" could also benefit from deploying these machines. It is inevitable that these new applications will require some revisions to the core design so that they can be used effectively. A modular approach is felt to be required so that it is possible to "mix and match" different modules to design application-specific machines in a relatively straightforward manner. When such a philosophy is well established, it would be much easier to identify missing elements or whether particular modules need to be redesigned to satisfy some specific constraint (of size or power). Several issues need to be considered in deciding on the most appropriate way of introducing the modularity; these involve the flexibility of the researchers with the uniformity and the different options offered by competitors. Communications protocol is a vital aspect since this determines whether the different components will connect to each other in a sensible manner. Consequently a thematic network for this technology area has been set up under the EC Brite EuRam programme. Here the intention is to provide a forum for the researchers to interact and maximise the future development of such industrial mobile robotic vehicles.

6. Conclusions

The papers has presented the state-of-the-art in the area of climbing robots. The main machines developed to date have been introduced and these show that the driving forces are hazardous applications where there is a clear need for robotic intervention. The most commonly used technique for climbing on surfaces is based on vacuum suction grippers and many of the machines discussed here utilise this method. The likely future development of mobile robots is also considered; it is expected that the traditional one-off prototyping approach to robot design cannot continue for much longer and there needs to be emphasis on re-using of the solutions already developed elsewhere. T support such transfer of technology from one application to another, greater thought has to be given to modularity and system integration issues so that the different components can be combined much more easily. This is the aim of the EC Brite Euram thematic network on climbing and walking robots being co-ordinated by the author in collaboration with partners across the European Community.

Researchers and scientists interested in mobile robotics vehicles are invited to play an active part in taking the technology to its next logical stage - whatever this may be! To do this simply contact the author so that you may be included in the activities of the CLAWAR Network.

7. References

[1] Virk GS; EC Brite Euram III Thematic Network on climbing and walking robots, Exploratory Phase CLAWAR Report, University of Portsmouth, January 1997.

[2] Neubauer W; Locomotion with articulated legs in pipes or ducts, Proceedings on Int Conference on Intelligent Autonomous Systems, pp 64-71, Pittsburgh, Feb 1993.

[3] Neubauer W; A spider-like robot that climbs vertically in ducts or pipes, Proceedings Int Conference on Intelligent Robots and Systems, Vol 2, pp 1178-1185, Munich, Sept 1994.

[4] Cornelis J, Sahli H, Acheroy M and Baudoin Y; Anti-personal mines, a worldwide problem: From political conscience towards humanitarian, research and industrial action, Proceedings of the 6th Int Symposium on Measurement and Control in Robotics (ISMCR 96), pp 1-5, Brussels 9-11 May 1996.

[5] Nicoud J-D; Mine Clearance: not only a problem for the military any more, Proceedings of the 6th Int Symposium on Measurement and Control in Robotics (ISMCR 96), pp 6-10, Brussels 9-11 May 1996.

[6] Seward D; Robots in construction, Industrial robots, Vol 19, No3, pp 25-29, 1992.

[7] Luk BL, Collie AA, Bevan N and Billingsley J; An articulated limb climbing vehicle with autonomous floor-to-wall transfer capability, Proceedings of 1st IFAC Int Workshop on Intelligent Autonomous Vehicles, pp 20-24, Southampton, April 1993.

[8] Collie AA; Unusual robots, Industrial robots, Vol 19, No4 pp 13-16, 1992.

[9] Hollerbach JM, Hunter IW and Ballantyne J; A comparative analysis of actuator technologies for robotics, The Robotics Review, Vol 2, pp 299-342, MIT Press, Cambridge, 1991.

[10] Colombi S, Raimondi T and Costi G; Improvements of actuators in tele-operators, Proceedings of the 4th Int Symposium on Offshore robotics and Artificial Intelligence, Marseille, pp 501-507, 11-12 December 1991.

[11] E W Jezierski, A Bartoszewicz and K Mianowski; Teleman 44: Manipulator arm development and end effector design, Final Report, University of Lodz, 1996.

[12] Greenhill R; Private communication, The Shadow Group Project, 251 Liverpool Road, London N1 1PX.

[13] Burrows MS and Yeomans D; Magnetically attached vehicles for a reactor pressure vessel inspection, Proceeding of conference on Remote techniques for nuclear plant, organised by the British Nuclear Energy Society, pp 152-156, Stratford-upon- Avon, May, 1993.

[14] Fortuna L, Gallo A, Giudice G and Muscato G; Robinspec: A mobile walking robot for the semi-autonomous inspection of industrial plants, WAC 96, Montpellier, May 1996.

[15] Nishi A; Development of wall-climbing robots, Computers Elect Engineering, Vol 22, No 2, pp 123-149, 1996.

[16] Nishi A; A biped walking robot capable of moving on a vertical wall, Mechatronics, Vol 2, No 6, pp 543-554, 1992.

[17] Nishi A and Miyagi H; Mechanism and control of propeller type wall-climbing robot, Intelligent robots and systems, edited by Volker Graefe, Elsevier, pp 669-677, 1995.

[18] Nishi A and Miyagi H; Simulation and test of a flight-type wall-climbing robot, Proceedings of the World Automation Congress (WAC 96), pp 569-576, Montpellier, May 1996.

[19] Berns K; Walking machines catalogue on the WWW, Internet address is at http://www:fzi.de/divisions/ipt/WMV/preface/preface.html.

[20] Billingsley J, Collie AA and Rook MJ; An improved wall-climbing robot with minimal action, Proceedings of the 1st IFAC Int Workshop on Intelligent Autonomous Vehicles, pp 14-17, Southampton, April 1993.

[21] Luk BL, Collie AA and White T; Nero: A teleoperated wall climbing vehicle are assisting inspection of a nuclear reactor pressure vessel, Proceedings 13th ASME Conference on Computers in Engineering, Vol 2, pp 167-170, 1993.

[22] Luk BL, Collie AA, Billingsley J, White T and Bevan N; Real time software control system for the Nero wall climbing robot, Proceedings of 4th Euromicro Workshop on real time systems, pp 74-78, Athens, June 1992.

[23] Luk BL, Collie AA, Piefort V and Virk GS; Robug III: A tele-operated climbing and walking robot, Proceeding of UKACC Int Conference Control 96, Vol 1, pp 347-352, Exeter, Sept 1996.

Lecture Notes in Control and Information Sciences

Edited by M. Thoma

1993–1998 Published Titles:

Vol. 186: Sreenath, N.
Systems Representation of Global Climate Change Models. Foundation for a Systems Science Approach.
288 pp. 1993 [3-540-19824-5]

Vol. 187: Morecki, A.; Bianchi, G.; Jaworeck, K. (Eds)
RoManSy 9: Proceedings of the Ninth CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators.
476 pp. 1993 [3-540-19834-2]

Vol. 188: Naidu, D. Subbaram
Aeroassisted Orbital Transfer: Guidance and Control Strategies
192 pp. 1993 [3-540-19819-9]

Vol. 189: Ilchmann, A.
Non-Identifier-Based High-Gain Adaptive Control
220 pp. 1993 [3-540-19845-8]

Vol. 190: Chatila, R.; Hirzinger, G. (Eds)
Experimental Robotics II: The 2nd International Symposium, Toulouse, France, June 25-27 1991
580 pp. 1993 [3-540-19851-2]

Vol. 191: Blondel, V.
Simultaneous Stabilization of Linear Systems
212 pp. 1993 [3-540-19862-8]

Vol. 192: Smith, R.S.; Dahleh, M. (Eds)
The Modeling of Uncertainty in Control Systems
412 pp. 1993 [3-540-19870-9]

Vol. 193: Zinober, A.S.I. (Ed.)
Variable Structure and Lyapunov Control
428 pp. 1993 [3-540-19869-5]

Vol. 194: Cao, Xi-Ren
Realization Probabilities: The Dynamics of Queuing Systems
336 pp. 1993 [3-540-19872-5]

Vol. 195: Liu, D.; Michel, A.N.
Dynamical Systems with Saturation Nonlinearities: Analysis and Design
212 pp. 1994 [3-540-19888-1]

Vol. 196: Battilotti, S.
Noninteracting Control with Stability for Nonlinear Systems
196 pp. 1994 [3-540-19891-1]

Vol. 197: Henry, J.; Yvon, J.P. (Eds)
System Modelling and Optimization
975 pp approx. 1994 [3-540-19893-8]

Vol. 198: Winter, H.; Nüßer, H.-G. (Eds)
Advanced Technologies for Air Traffic Flow Management
225 pp approx. 1994 [3-540-19895-4]

Vol. 199: Cohen, G.; Quadrat, J.-P. (Eds)
11th International Conference on Analysis and Optimization of Systems – Discrete Event Systems: Sophia-Antipolis, June 15–16–17, 1994
648 pp. 1994 [3-540-19896-2]

Vol. 200: Yoshikawa, T.; Miyazaki, F. (Eds)
Experimental Robotics III: The 3rd International Symposium, Kyoto, Japan, October 28-30, 1993
624 pp. 1994 [3-540-19905-5]

Vol. 201: Kogan, J.
Robust Stability and Convexity
192 pp. 1994 [3-540-19919-5]

Vol. 202: Francis, B.A.; Tannenbaum, A.R. (Eds)
Feedback Control, Nonlinear Systems, and Complexity
288 pp. 1995 [3-540-19943-8]

Vol. 203: Popkov, Y.S.
Macrosystems Theory and its Applications: Equilibrium Models
344 pp. 1995 [3-540-19955-1]

Vol. 204: Takahashi, S.; Takahara, Y.
Logical Approach to Systems Theory
192 pp. 1995 [3-540-19956-X]

Vol. 205: Kotta, U.
Inversion Method in the Discrete-time Nonlinear Control Systems Synthesis Problems
168 pp. 1995 [3-540-19966-7]

Vol. 206: Aganovic, Z.; Gajic, Z.
Linear Optimal Control of Bilinear Systems with Applications to Singular Perturbations and Weak Coupling
133 pp. 1995 [3-540-19976-4]

Vol. 207: Gabasov, R.; Kirillova, F.M.; Prischepova, S.V.
Optimal Feedback Control
224 pp. 1995 [3-540-19991-8]

Vol. 208: Khalil, H.K.; Chow, J.H.; Ioannou, P.A. (Eds)
Proceedings of Workshop on Advances inControl and its Applications
300 pp. 1995 [3-540-19993-4]

Vol. 209: Foias, C.; Özbay, H.; Tannenbaum, A.
Robust Control of Infinite Dimensional Systems: Frequency Domain Methods
230 pp. 1995 [3-540-19994-2]

Vol. 210: De Wilde, P.
Neural Network Models: An Analysis
164 pp. 1996 [3-540-19995-0]

Vol. 211: Gawronski, W.
Balanced Control of Flexible Structures
280 pp. 1996 [3-540-76017-2]

Vol. 212: Sanchez, A.
Formal Specification and Synthesis of Procedural Controllers for Process Systems
248 pp. 1996 [3-540-76021-0]

Vol. 213: Patra, A.; Rao, G.P.
General Hybrid Orthogonal Functions and their Applications in Systems and Control
144 pp. 1996 [3-540-76039-3]

Vol. 214: Yin, G.; Zhang, Q. (Eds)
Recent Advances in Control and Optimization of Manufacturing Systems
240 pp. 1996 [3-540-76055-5]

Vol. 215: Bonivento, C.; Marro, G.; Zanasi, R. (Eds)
Colloquium on Automatic Control
240 pp. 1996 [3-540-76060-1]

Vol. 216: Kulhavý, R.
Recursive Nonlinear Estimation: A Geometric Approach
244 pp. 1996 [3-540-76063-6]

Vol. 217: Garofalo, F.; Glielmo, L. (Eds)
Robust Control via Variable Structure and Lyapunov Techniques
336 pp. 1996 [3-540-76067-9]

Vol. 218: van der Schaft, A.
L_2 Gain and Passivity Techniques in Nonlinear Control
176 pp. 1996 [3-540-76074-1]

Vol. 219: Berger, M.-O.; Deriche, R.; Herlin, I.; Jaffré, J.; Morel, J.-M. (Eds)
ICAOS '96: 12th International Conference on Analysis and Optimization of Systems - Images, Wavelets and PDEs: Paris, June 26-28 1996
378 pp. 1996 [3-540-76076-8]

Vol. 220: Brogliato, B.
Nonsmooth Impact Mechanics: Models, Dynamics and Control
420 pp. 1996 [3-540-76079-2]

Vol. 221: Kelkar, A.; Joshi, S.
Control of Nonlinear Multibody Flexible Space Structures
160 pp. 1996 [3-540-76093-8]

Vol. 222: Morse, A.S.
Control Using Logic-Based Switching
288 pp. 1997 [3-540-76097-0]

Vol. 223: Khatib, O.; Salisbury, J.K.
Experimental Robotics IV: The 4th International Symposium, Stanford, California, June 30 - July 2, 1995
596 pp. 1997 [3-540-76133-0]

Vol. 224: Magni, J.-F.; Bennani, S.; Terlouw, J. (Eds)
Robust Flight Control: A Design Challenge
664 pp. 1997 [3-540-76151-9]

Vol. 225: Poznyak, A.S.; Najim, K.
Learning Automata and Stochastic Optimization
219 pp. 1997 [3-540-76154-3]

Vol. 226: Cooperman, G.; Michler, G.; Vinck, H. (Eds)
Workshop on High Performance Computing and Gigabit Local Area Networks
248 pp. 1997 [3-540-76169-1]

Vol. 227: Tarbouriech, S.; Garcia, G. (Eds)
Control of Uncertain Systems with Bounded Inputs
203 pp. 1997 [3-540-76183-7]

Vol. 228: Dugard, L.; Verriest, E.I. (Eds)
Stability and Control of Time-delay Systems
344 pp. 1998 [3-540-76193-4]

Vol. 229: Laumond, J.-P. (Ed.)
Robot Motion Planning and Control
360 pp. 1998 [3-540-76219-1]

Vol. 230: Siciliano, B.; Valavanis, K.P. (Eds)
Control Problems in Robotics and Automation
328 pp. 1998 [3-540-76220-5]

Vol. 231: Emel'yanov, S.V.; Burovoi, I.A.; Levada, F.Yu.
Control of Indefinite Nonlinear Dynamic Systems
196 pp. 1998 [3-540-76245-0]

Vol. 232: Casals, A.; de Almeida, A.T.
Experimental Robotics V: The Fifth International Symposium Barcelona, Catalonia, June 15-18, 1997
190 pp. 1998 [3-540-76218-3]

Vol. 233: Chiacchio, P.; Chiaverini, S.
Complex Robotic Systems
189 pp. 1998 [3-540-76265-5]

Vol. 234: Arena, P, Fortuna, L, Muscato, G and Xibilia, M.G.
Neural Networks in Multidimensional Domains: Fundamentals and New Trends in Modelling and Control
179 pp. 1998 [1-85233-006-6]

Vol. 235: Chen, B.M
H_∞ Control and Its Applications
361 pp. 1998 [1-85233-026-0]