Obstacle Movement Prediction considering Obstacle's Dynamics and a Priori Knowledge of its Goals

Ezequiel Bevilacqua, ETSEIB, UPC (Polytechnic Univ. of Catalunya),
Tetsuya Kimura, Nagaoka University of Technology*

* Dept. of Mech. Eng., Nagaoka Univ. of Tech., 1603-1 Kamitomioka,
Nagaoka, Niigata, 940-2188 JAPAN
e-mail: kimura@mech.nagaokaut.ac.jp , Tel.+81-258-47-9708 , Fax.+81-258-47-9770

Abstract : In this paper development of an advanced prediction method is proposed as a tool to enhance mobile robot navigation in dynamic environments. This research experiments the efficiency of this concept in the Small-size RoboCup environment in which the obstacles are the opponent soccer robots and, thus, not only move but act in an intelligent way to prevent our Multi-Agent System from achieving its goals. Understanding of the opponent's path-planning strategy and dynamical characteristics improves the accuracy of movement prediction. The focus is done in considering both the dynamics of the moving obstacles and a priori knowledge of their goals. Efficiency of this approach is tested for simple game configurations through simulations and experiments.
Keywords : intelligent prediction, motion control, moving obstacles, obstacle avoidance, dynamics based prediction.

1 Introduction

Given the fast evolution of the autonomous robot's field, it can already be foreseen that in a near future it will be common to encounter mobile Multi-Robot Systems (MRS) in industrial areas. The development of intelligent robots adds a certain capability for complex behaviour which our prediction systems should not ignore. We can state that problems of this type and several others can be better dealt with by predicting the future actions of all the robots in the system. Thus, advanced prediction methods become paramount.

To develop these advanced prediction methods we use as test bed the small-size RoboCup [1] environment for several reasons. Consisting of two opposed MRS pursuing conflicting goals in the same domain, RoboCup settings depend on efficient and accurate predictions in order to achieve a collision-free navigation. RoboCup robots use a vision system, and thus the control loop holds a huge delay which stresses the need for predictions. It is, then, an appropiate scenario to test our solution to the problem described previously.

The solutions offered so far to the obstacle prediction problem range from as simple as mere linear extrapolation of positional data [2] to complex conversions of the system into seemingly analogous expressions (such as force fields [3]). In many cases, the obstacle's trajectories are subject to several restraints which facilitate prediction but don't correspond to the actual nature of the system, such as its dynamic and kinematic characteristics.

In this paper it is demonstrated that better predictions can be made by considering the dynamic and kinematic characteristics of the obstacles together with an a priori knowledge of their behaviour; this is what will be called Intelligent Prediction in this paper. Future state of obstacles shall not be just mathematically extrapolated but deduced from their dynamic and kinematic models. In the same fashion, a priori knowledge of the behaviour/goals of the obstacles allow to predict an otherwise seemingly random evolution which would prevent accurate prediction only allowing for bounding of the possible future states.

2 Intelligent Prediction

The general concept for the obstacle intelligent prediction can be schematized into the diagram shown in Figure 1. It is based on the fact that what we usually call "obstacles" are actually systems whose evolution is tied to internal rules and, in case of having a complex behaviour, respond to certain conditions of the environment related to their activity. An example could be any intelligent robot, as its mechanics translate into kinematic and dynamic characteristics and its artificial intelligence pursues a goal following certain rules which we may estimate. In general, as can be deduced from the diagram, we focus on systems such that knowledge of its characteristics allows to estimate the corresponding control input from historic

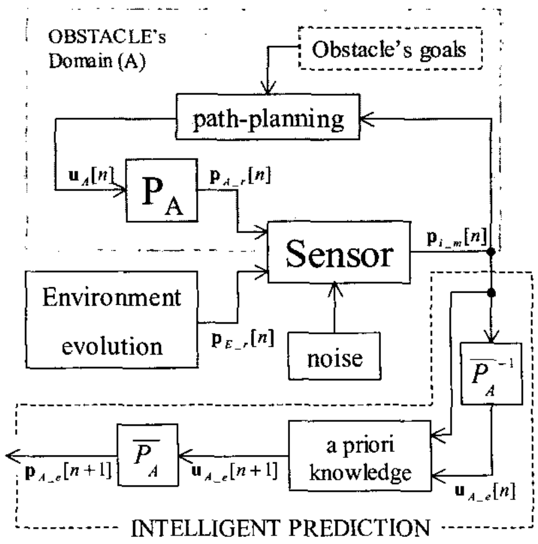

Figure 1: Discrete general diagram for the intelligent prediction system. P_A: obstacle's system. $\overline{P}_A$: model of obstacle's system.

data of the output variables.

Let $p[n]$ be the output variable and $u[n]$ the control input. If a record of the output is kept and a model of the obstacle ($\overline{P_A}$) is known, it is possible to accurately predict the future state of the obstacle if the future control signal can be estimated.

Dealing with obstacles whose path-planning logic is unknown, we can always count on the obstacle being programmed to carry out a certain task. That knowledge is the key to the accurate estimation of $u[n+1]$. Reached this point it is convenient to separate the prediction in two levels. The first is the one predicting the system from its control input to its output; we call this the *navigation prediction*. Then we have the prediction which associates the output of the system with the future control input, this we name *path-planning prediction*. Both will now be discussed in detail.

2.1 Navigation prediction

Once the control input is estimated by the path-planning prediction, the prediction of the future state is relatively easy based on equation of motion, if available. The following example with simplified moving robot (modeled as a single mass) will help to clarify this prediction's working.

$$M\ddot{p} + C\dot{p} = F \ , \ with \ p = \begin{pmatrix} x \\ y \end{pmatrix} \ , \ F = \begin{pmatrix} f_x \\ f_y \end{pmatrix} \quad (1)$$

where M =mass of the robot, C =damping coeficient of the robot's dynamics and F, the control input (force). When the sampling period Δt is sufficiently small, we have

$$\dot{p} \approx \frac{p(n\Delta t) - p((n-1)\Delta t)}{\Delta t} \ , \ \ddot{p} \approx \frac{\dot{p}(n\Delta t) - \dot{p}((n-1)\Delta t)}{\Delta t}$$

writing, for simplicity, $p(n\Delta t)$ as $p[n]$ and substituting in (1). Then, the $p[n+1]$ we want to predict is explicitly given by

$$p[n+1] = A\left(F[n+1] + \left(\frac{2M}{\Delta t^2} + \frac{C}{\Delta t}\right)p[n] - \frac{M}{\Delta t^2}p[n-1]\right)$$

where

$$A = \left(\frac{M}{\Delta t^2} + \frac{C}{\Delta t}\right)^{-1}$$

Thus, knowing $F[n+1]$, $p[n]$ and $p[n-1]$ we have $p[n+1]$. Here we can already see one of the advantages of the intelligent prediction as, being deduced from the robot's model, it will consider the physical restrictions of the system automatically. In this fashion, for example, we will never predict a motion in which the robot would have to follow a trajectory beyond its maneuverability capacity, nor will we predict an acceleration faster than the dynamics of the system allow. It is now only necessary to estimate the future control input.

2.2 Path-planning prediction

At this level is when the "intelligence" of our approach has full effect. This prediction has to determine what the future control signal will be for the obstacle based on a priori knowledge on the system. The feedback information for this algorithm is the output of the system but, by applying inverse dynamics (as shown in Fig. 1 as $\overline{P}_A^{-1}$), we also obtain the control signal $u[n]$ which corresponds to this feedback. This is done because some systems express their path-planning logic in a way directly related to the control signal and, thus, estimating it may improve the estimation of the future one.

In the determination of the future control signal $u[n+1]$ we have a chance to further improve the prediction results in several ways. For instance, sensor measurement errors can be detected and conveniently removed at this phase. We also have the opportunity of adapting to changes in the environment which will affect the system we try to predict. Some examples on how to apply the a priori knowledge to the path-planning prediction are given in Section 3.2.

3 Application to Particular Case

In order to obtain clear results of simple interpretation, it was decided to focus the experiments on a particular robot of a RoboCup team: the goalie. Our experimental apparatus for RoboCup experiments is depicted by Figure 2. The goalie robot characterizes for moving always parallel to the goal line and for having a very simple logic: it just tracks, *not* anticipate, the movement of the ball. As a result of these characteristics, the movement of the goalie robot is substantially simple when considered on the cartesian plane. The target of this research is to demonstrate that, even in this case, an intelligent prediction is more efficient than linear extrapolation. Finally, the robot chosen as goalie is two-wheeled as the one shown in Figure 3. The next two sections discuss the application of the intelligent prediction concept to this system in consideration of the described characteristics.

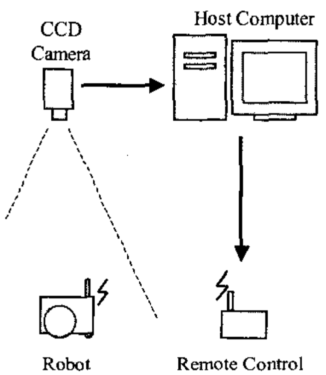

Figure 2: Experimental apparatus.

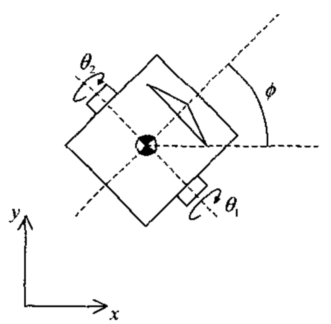

Figure 3: System's variables in global reference.

3.1 Navigation prediction

In our case, the output obtained consists of the (x, y) coordinates of the robot on the cartesian plane and the orientation ϕ, with respect to the x-axis, of the robot. The input signals to the system are the speed references $\dot{\theta}_{r1}$ and $\dot{\theta}_{r2}$ for the independent step-type motors which drive the wheels.

3.1.1 Hypotheses

The robot has non-holonomic nature of its mechanical system. Due to this, it is necessary to make some hypotheses about the navigation of the robot in order to obtain explicit input/output expressions with which the predictions can be made. Given that the feedback is obtained through a global vision system supplied by a CCD camera, it is common to have a sampling period of around 100ms. As this sampling period is relatively short compared to the dynamics of the robot, the following hypotheses are considered fitting.

Hypothesis 1: for each sampling interval the control signal is constant.

Hypothesis 2: given a sampling interval, the path of the robot can be fairly approximated by a circumference arc.

3.1.2 Kinematic considerations introduced by Hypothesis 2

In order to learn how the circumference arc restriction affects the independence of the rotation of the robot's wheels, it is necessary to state how the arc itself is defined. Once the arc's characteristics are known, the restrictions it introduces to the kinematics of the robot can be studied.

Determining the Arc Path from the Experimental Data. Here we assume the initial state of the robot is the origin with orientation $\phi = 0$ and the final coordinates are (x_f, y_f, ϕ_f). As a result of this transformation the trajectory is tangent to the x-axis at $x = 0$. Furthermore, knowing the trajectory is circular we can state that the centre of the arc's circumference is on the y-axis. From this arc we will need to know its radius R and its angle α_f.

The following expressions are directly deduced from Figure 4.

$$R = \frac{d}{2\sin\frac{\alpha_f}{2}},\ d = \sqrt{x_f^2 + y_f^2},\ \frac{\alpha_f}{2} = \arctan\frac{y_f}{x_f} \quad (2)$$

and, $\phi = \alpha$, $l = R\alpha$, where d=cord of arc, l=arc length.

Note that two cases, a rectilinear movement and a pure-spinning movement, would not give direct numeric values for the arc's parameters and it will be treated as a special case in experiment.

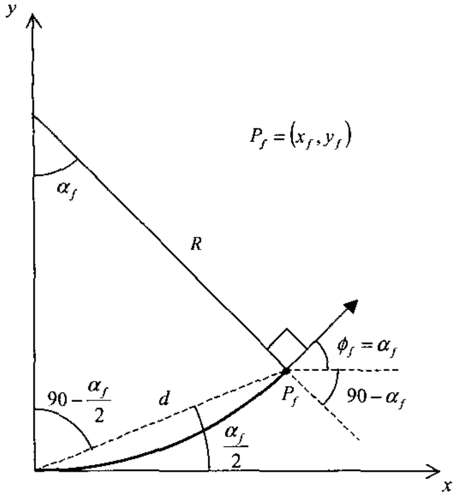

Figure 4: Arc description in local reference.

3.1.3 Matching Robot's Kinematics with Path's Characteristics

In first place, the kinematic model of the robot must be deduced. Figure 3 describes the system's variables in the global reference. Its equations are the following,

$$\dot{\phi} = \frac{r}{L}(\dot{\theta}_1 - \dot{\theta}_2),\ v = \frac{r}{2}(\dot{\theta}_1 + \dot{\theta}_2) \qquad (3)$$

where v=linear speed, $\dot{\theta}_i$=angular speed for wheel i, r=wheel radius, L=distance between wheels.

Now, the evolution of ϕ with respect to l can be

$$\frac{d\phi}{dl} = \frac{\alpha}{R\alpha} = \frac{1}{R} \qquad (4)$$

which is the condition that must be verified at every instant in order to remain on the arc. The next step consists in translating this restriction to our kinematic model. Once the dependency of $\ddot{\theta}_i$ and $\dot{\theta}_i$ is learnt, the problem of finding the required input signal to satisfy it can be faced.

$$\frac{d\phi}{dl} = \frac{\frac{d\phi}{dt}}{\frac{dl}{dt}} = \frac{\dot{\phi}}{v}$$

now, combining equations (3) and (4) we obtain

$$\dot{\theta}_1 = K_a\dot{\theta}_2 \qquad (5)$$

with

$$K_a = \frac{2R+L}{2R-L}$$

This expression describes the relation that the angular speeds of the wheels must verify at any time to track the circumference arc. Differentiating it we obtain

$$\ddot{\theta}_1 = K_a\ddot{\theta}_2 \qquad (6)$$

In addition, we have

$$\theta_{f_1} = \frac{\alpha(R+L/2)}{r}$$

3.1.4 Kinematic restrictions on Dynamics

Let's now clearly state the problem which must be tackled. For each wheel of the robot, it is now known which angular distance they must cover while holding to the kinematic restrictions introduced by Hypothesis 2. Moreover, in calculating the input signal which satisfies the characteristics of this motion, it must be made sure that Hypothesis 1 remains true. The first step consists in describing the dynamic model of the system.

Stepping Motor Dynamics. The input signal for a stepping motor is a reference value ($\dot{\theta}_r$) of the angular speed we want it to develop. As a result, it can be approximated by a first order system with unity gain and a time constant T_{step} as described in the following transfer function

$$\frac{\dot{\theta}}{\dot{\theta}_r} = \frac{1}{T_{step}s+1}$$

Transforming this expression to the time domain we obtain the following relation between the input signal $\dot{\theta}_r$ and the kinematic variables $\dot{\theta}$ and $\ddot{\theta}$,

$$\ddot{\theta} = \frac{\dot{\theta}_r - \dot{\theta}}{T_{step}} \qquad (7)$$

Obtaining $\dot{\theta}_r$. From now on, for notational simplification, the subscripts referring to the wheels shall be obviated as the previous section demonstrated it is irrelevant for which wheel the calculations are made. An explicit expression of the constant control signal which drives the robot along the pre-defined arc trajectory will be found. The available data to do this consists of: the angular distance covered by the wheel along the arc (θ_f), the initial angular speed of the wheel ($\dot{\theta}_0$) and the duration of the motion (T_c). The approach to this problem starts with the integragration of equation (7). Assuming the initial condition $\theta(0) = 0, \dot{\theta}(0) = \theta_0$, we have

$$\theta_f = \int_0^{T_c} \dot{\theta}\, dt == \dot{\theta}_r T_c + \frac{T_{step}}{\exp\left(\frac{C_1}{T_{step}}\right)}\left(\frac{1}{\exp\left(\frac{T_c}{T_{step}}\right)} - 1\right)$$

Through some algebraic transformations the desired explicit expression of $\dot{\theta}_r$ is finally obtained.

$$\dot{\theta}_r = \frac{\theta_f + A \cdot T_{step}\dot{\theta}_0}{T_c + A \cdot T_{step}} \tag{8}$$

with A being the following constant

$$A = \frac{1}{\exp\left(\frac{T_c}{T_{step}}\right)} - 1$$

This $\dot{\theta}_r$ is the estimated control input u to the robot.

3.2 Path-planning Prediction

In the path-planning level, the case at hand is very simple: it is assumed that the robot tries to remain parallel to the goal line while tracking the position of the ball. If this information was ignored, predictions would just assume the goalie tries to move towards the ball, incurring in error. Using a priori knowledge allows to rectify the tendency of the robot to leave the goal-line. Moreover, it can be anticipated when the robot will change directions as it is known it tracks the position of the ball. Experience showed that, due to deficient control, the goalie did not remain over the goal-line but deviated, sometimes taking a long time to return. Thus, another use of a priori knowledge applied in the experiments was the introduction of a parameter Ω to better represent the behaviour of the goalie. The Ω parameter was used as a weight function of this deviation so that it was not totally ignored as in the original path-planning prediction. Specifically, Ω is an absolute boundary of the robot's deviation from the ideal path which acts in the following way: while the deviation is inferior to this parameter, the predictions will aim for the robot tracking the ball ignoring the deviation. On the other hand, whenever the deviation is larger than Ω, the predictions will assume the robot is still tracking the ball while trying to return to the acceptable deviation area. For different kinds of goalies, different Ω's give the optimal prediction results.

4 Experimental Results

Two kinds of experiments were carried out. In both, the prediction algorithms were identical being, thus, the only difference, the source of the goalie's and ball's data. One consisted of simulations in which an ideal goalie tracked the random movement of a ball. The other was the experimental result of the goalie's and ball's motions by means of the RoboCup experimental apparatus. The robot was manually operated with a remote control. In this case, the experiments were done with a sampling period of 60ms. For both experiments the prediction error was calculated as the 2-norm of the difference vectors from the actual robot's position to the predicted one.

4.1 Simulation Result

The simulations, in which the behaviour of the robot is ideal and no feedback noise interferes with the predictions, show that, despite the highly linear behaviour of the robot, the intelligent prediction's average error is a 30% of the linear extrapolation's. An example simulation result is described by the plots in Figure 5. The top plot shows how the simulated robot perfectly tracks the ball without anticipating its moves. The bottom one plots the error for both prediction methods as a funtion of time so we may easily associate the error peaks to the times at which the robot changes direction. It can be seen that the intelligent prediction sports two main advantages. One is that it anticipates the changes in the system as it knows which changes in the environment will affect the system's evolution. This shows clearly at the initial impulse of the peaks where the intelligent prediction error is usually a 50% of the linear extrapolation's error. Moreover, as can be seen in the peak at $t = 10s$, sometimes this anticipation drastically reduces the error, being a 5% of the linear one. The other obvious advantage is the reaction time of the intelligent prediction to the change. Its error peaks rapidly decrease as a consequence of the algorithm learning the nature of the change. On the other hand, the linear extrapolation does not react until the system is restored to a linear behaviour.

4.2 Experiment Result

Several experiments were carried out with the overall average intelligent prediction error being a 130% of the linear extrapolation's. Exploration of the data allowed to determine the causes for this result; a common experiment result is displayed and discussed. Figure 6, shows the result of the experiment for the real robot and both predictions. The intelligent prediction gives a much smoother result which is a consequence of it considering the dynamics of the system. Moreover it can be observed that this method incurs in large error when the robot behaves in an unexpected way. Specifically, at the beginning of the experiment the ball is above the robot (at greater Y) but the manual control does not react immediately. While the linear prediction is unaffected by this fact, the intelligent one keeps assuming the robot will try to move towards the ball, thus producing the prediction error. A similar thing happens for $t \in [1.5, 2.4]$ where the ball is below but the robot unexpectedly interrupts its downward motion. These intelligent prediction errors are easily identified in the bottom plot of the figure. Finally, the short sampling period tends to support the linear extrapolation method as consecutive samples are so close that dynamic and kinematic characteristics are poorly represented.

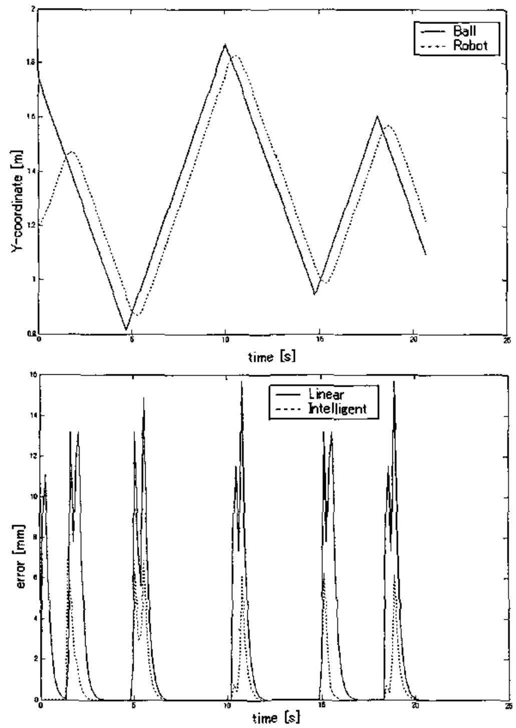

Figure 5: (top) Simulated robot tracking ball position. (bottom) Absolute error for both methods.

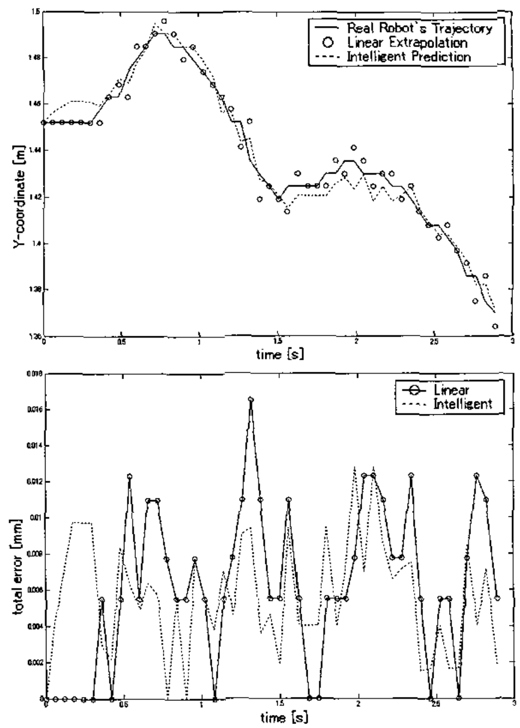

Figure 6: (top) Prediction results for both methods. (bottom) Prediction error time plot.

5 Conclusions

We have proposed an intelligent prediction method which takes into account the dynamic and kinematic characteristics of the system to be predicted together with a priori knowledge of the system's goals. By using this method we have the following results.

1. The intelligent prediction anticipates changes in the system in a fashion which allows to significantly reduce the prediction error associated to them.

2. Basing its predictions on the Dynamics model of the robot, the intelligent prediction rapidly learns the nature of changes and swiftly adapts to them. This way the error is reduced faster than for Dynamics-ignoring methods.

3. Smoother, system-comprehensive predictions are obtained from the intelligent approach.

4. An intelligent prediction will give improved results when the assumptions done on the path-planning and navigation of the robot prove to be true. On the other hand, wrong assumptions will cause the prediction to be inefficient.

References

[1] For detailed information on RoboCup the reader may address to http://www.robocup.org/

[2] Dilip Kumar Pratihar, Kalyanmoy Deb and Amitabha Ghosh: A Genetic-fuzzy Approach for Mobile Robot Navigation among Moving Obstacles. International Journal of Approximate Reasoning 20 (1999), pp. 145-172.

[3] Akira Inoue, Kenji Inoue and Yoshikuni Okawa: On-line Motion Planning of an Autonomous Mobile Robot to Avoid Multiple Moving Obstacles Based on the Prediction of their Future Trajectories. Journal of Robotics Society of Japan, Vol.15 No.2 (1997), pp. 249-260.