COMPUTERS AND ELECTRONICS IN AGRICULTURE 65 (2009) 247–257

available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/compag

# Optimal manipulator design for a cucumber harvesting robot

E.J. Van Henten[a,b,*], D.A. Van't Slot[c], C.W.J. Hol[d], L.G. Van Willigenburg[e]

[a] Wageningen UR Greenhouse Horticulture, P.O. Box 16, NL-6700 AA Wageningen, The Netherlands
[b] Farm Technology Group, Wageningen University, P.O. Box 17, NL-6700 AA Wageningen, The Netherlands
[c] DWA installatie- & energieadvies, Duitslandweg 4, 2411 NT Bodegraven, The Netherlands
[d] Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, NL-2628 CD Delft, The Netherlands
[e] Systems and Control Group, Wageningen University, Bornsesteeg 59, NL-6708 PD Wageningen, The Netherlands

ARTICLE INFO

*Article history:*
Received 13 April 2006
Received in revised form
13 October 2008
Accepted 4 November 2008

*Keywords:*
Greenhouse
Cucumber
Harvest
Robot
Optimisation

ABSTRACT

This paper presents a procedure and the results of an optimal design of the kinematic structure of a manipulator to be used for autonomous cucumber harvesting in greenhouses. The design objective included the time needed to perform a collision-free motion from an initial position to the target position as well as a dexterity measure to allow for motion corrections in the neighborhood of the fruits. The optimisation problem was solved using the DIRECT algorithm implemented in the Tomlab package. A four link PPRR type manipulator was found to be most suitable. For cucumber harvesting four degrees-of-freedom, *i.e.* three translations and one rotation around the vertical axis, are sufficient. The PPRR manipulator described in this paper meets this requirement. Although computationally expensive, the methodology used in this research was found to be powerful and offered an objective way to evaluate and optimise the kinematic structure of a robot to be used for cucumber harvesting.
© 2008 Elsevier B.V. All rights reserved.

## 1. Introduction

In 2001 a 5 years research project on autonomous cucumber harvesting was concluded with a successful field test of the harvesting robot in a research greenhouse (Van Henten et al., 2003b). Various aspects of the development of this agro-robotic system such as the adoption of a new cultivation system, economics, logistics and, last but not least, the robot technology have been reported in Van Henten et al. (2002, 2003a,b), Van Willigenburg et al. (2004) and the references therein.

This paper highlights an aspect of this project not reported so far, namely task specific manipulator design and optimisation. At the beginning of the project the choice of the manipulator for the harvesting robot was based on an empirical analysis of the working environment and the task to be performed. This process is illustrated in Fig. 1. Main design criteria were the ability of the robot to reach cucumbers within a predefined greenhouse volume, its size because of the limited workspace and the possibility of collisions with plants and greenhouse structure, its weight carrying ability, speed and the ability of the machine to operate in adverse climate conditions with high temperature and humidity levels. Two-dimensional and three-dimensional models were used to evaluate various manipulator structures with respect to the harvesting task it should perform. As a result of this empirical qualitative analysis, a manipulator was chosen consisting of 7 links: a linear slide (*i.e.* a prismatic link) on top of which a Mitsubishi RV-E2 manipulator with 6 rotating links was mounted. This platform

* *Corresponding author at*: Farm Technology Group, Wageningen University, P.O. Box 17, NL-6708 AA Wageningen, The Netherlands. Tel.: +31 317 476491; fax: +31 317 483328.
E-mail address: eldert.vanhenten@wur.nl (E.J. Van Henten).
0168-1699/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.compag.2008.11.004

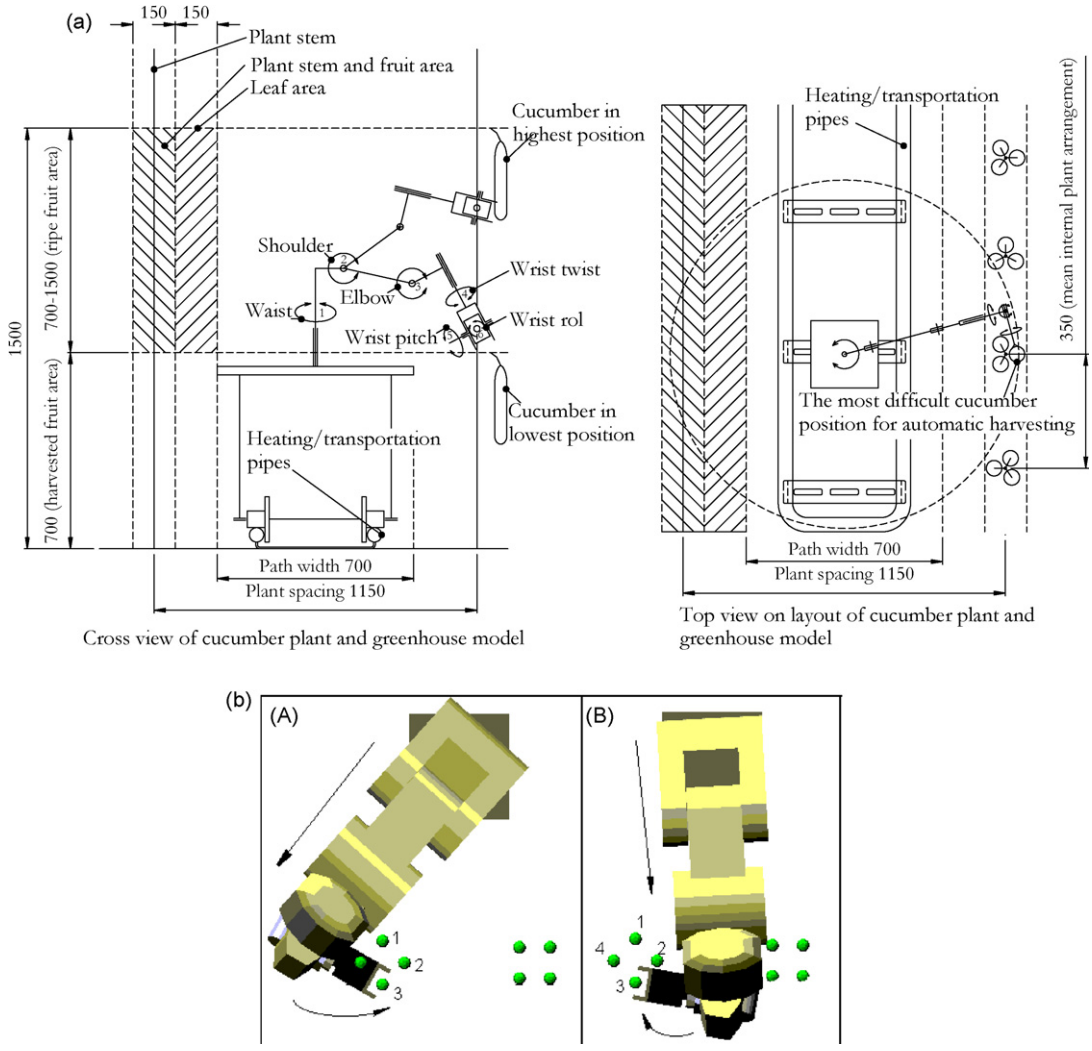


**Fig. 1 – Manipulator choice based on two-dimensional (a) and quasi-three-dimensional (b) models of the working environment of the harvesting robot.**

offered abundant flexibility for research and development during the initial phases of this project. Once the functional model of the harvesting robot had demonstrated the feasibility of fully autonomous harvesting of cucumber fruits in a greenhouse (Van Henten et al., 2003b), ways were investigated to improve the speed and success rate and to reduce the complexity of this robotic system. During the field tests of the harvesting robot it was found that four degrees of freedom, *i.e.* three translations and one rotation, were sufficient to harvest a cucumber. For this task, the 7-link manipulator used was overly complex. Using quantitative measures and nonlinear optimisation techniques, in a simulation study, optimal redesign of the manipulator was pursued. The methodology and results of that study are presented in this paper.

Optimal robot design has received considerable attention in the engineering literature. See for instance Paredis and Khosla (1995), Yang and Chen (2000), Ceccarelli and Lanni (2004), Bergamaschi et al. (2008) and Carbone et al. (2007), to mention a few. This research field is still evolving and has not yet produced clear cut answers to design questions evolving from application fields like the agricultural domain. To the best of our knowledge no ready to use software packages are available to tackle these design issues either. On the other hand, in the agricultural engineering domain, manipulator design has hardly received attention yet. Kondo et al. (1993) devoted special attention to the selection of the manipulator of their tomato-picking robot. They did not optimise the robot structure, but evaluated quantitatively four robot configurations based on a manipulability measure, normalised workspace volume and the so-called redundant space, *i.e.* the space that the middle of the manipulator can reach while the endpoint remains in the same position and orientation. They found that a seven link robot consisting of two prismatic and five rotational joints was best suited for the specific task. Recently, Song et al. (2007) reported the optimisation of a manipulator to be used for harvesting egg plants. Based on a literature review, Song et al. (2007) confirmed the observation that most agro-robotics research focuses on recognition and manipulator control and that manipulator design itself is hardly being addressed. Though ground breaking in their approach, Song et al. (2007) limited their work to the optimisation of two design parameters, namely the lengths of the

upper arm and the fore arm of a predefined manipulator with four rotational links (*i.e.* 4R) and the design objective was to cover a predefined workspace with the most compact mechanical structure. Design parameters such as the link type were not included in their research. And the design objective did not consider motion time or dexterity. It is the objective of this paper to expand this field of research into that direction. This paper presents the methodology and results of a case study focussed on the design of a manipulator for a cucumber harvesting robot.

The outline of the paper is as follows. In Section 2 the methodology will be presented. Section 3 contains the results of three optimisation case studies concerned with the design of a manipulator of a cucumber harvesting robot. Finally, in Section 4, the methodology and results will be discussed in view of application in horticultural practice and conclusions will be drawn.

## 2. Methodology

In the methodology of optimal robot design essentially three steps can be identified. First, the design specifications are defined. Secondly, these specifications are translated into a suitable quantitative performance criterion. This criterion is used to evaluate and compare various robot designs. Finally, heuristic, analytical or numerical techniques are used to find a design that optimises the performance criterion.

### 2.1. Definition of the optimisation problem

In line with the definition of Carbone et al. (2007), the problem studied in this research is to determine an optimal manipulator design represented by the set of optimised design parameters $\mathbf{p}^*$, such that

$$\mathbf{p}^* = \inf_{\mathbf{p}} P(\mathbf{p}) \tag{1a}$$

subject to constraints of the form

$$\mathbf{p}_{\min} \leq \mathbf{p} \leq \mathbf{p}_{\max}. \tag{1b}$$

Here $P$ is a suitably defined performance criterion. Basically, the set of design parameters $\mathbf{p}$ may include all kinematic design parameters of the robot such as the number of links, the type of links (prismatic (P) or rotational (R)) and, dependent on the type of the link, a subset of the Denavit–Hartenberg parameters $\theta$, $\alpha$, $a$ and $d$ associated with each link. See Craig (1989) for a definition of these Denavit–Hartenberg parameters. If a link includes a rotational joint, the joint angle $\theta$ is a control variable used to steer the joint and is not explicitly optimised. Then link twist $\alpha$, the link length $a$ and the link offset $d$ can be varied continuously to search for the best performance of the kinematic design. If a link is prismatic, $d$ is a control variable used to steer the joint. In that case, $\theta$, $\alpha$ and $a$ can be varied continuously during an optimisation. If the number and/or type of links is varied, the optimisation problem combines both discrete values (the number and/or type of links) and continuous values (the link parameters). Clearly, all variables can be restricted to lie within bounded regions to represent, for instance physical design limits, expressed by the inequality constraint (1b).

### 2.2. Design specification and quantitative evaluation

For autonomous cucumber harvesting the design requirements were defined as follows. First of all, a cucumber harvesting robot has to operate in a tight working environment as illustrated in Fig. 1 and the manipulator structure should be such that, during the manipulator motions, collisions of the manipulator with the canopy, the greenhouse structure and other parts of the robot are prevented. Secondly, to be economically feasible, the cycle time of a single harvest operation should not exceed 10 s (Bontsema et al., 1999). This includes fruit detection, ripeness assessment, 3D localisation of the fruits, motion planning, the motion of the end-effector to the fruit, gripping and cutting of the fruit and the return motion to the crate. To satisfy this performance requirement, the motion trajectory of the manipulator should be as short as possible. Finally, the manipulator should have maximum dexterity once the tool-centre-point (TCP) has arrived at the picking position. Dexterity is a measure quantifying the ability of a manipulator to move and rotate the TCP in all directions. On the harvest robot, a camera mounted on top of the end-effector may be used for final alignment of the end-effector with the fruit stalk in case of inaccurate positioning information from the main camera system of the robot. Therefore, considerable dexterity is needed to be able to make the required corrections to the position and orientation of the end-effector.

The above mentioned design specifications were formalised by two quantitative performance criteria, one evaluating the length of a collision-free path to the target, $P_{\text{path}}(\mathbf{p})$, and one evaluating the manipulator dexterity at the target, $P_{\text{dexterity}}(\mathbf{p})$.

#### 2.2.1. A performance criterion measuring the length of collision-free manipulator motions

For a manipulator with $n$ links, the overall path length was calculated as

$$P_{\text{path}}(\mathbf{p}) = \sum_{k=2}^{N} \sqrt{(\boldsymbol{\gamma}_k - \boldsymbol{\gamma}_{k-1}) A (\boldsymbol{\gamma}_k - \boldsymbol{\gamma}_{k-1})^T}, \tag{2}$$

in which $A$ is a $n \times n$ weighting matrix to scale the joint translations and rotations represented by $\boldsymbol{\gamma}_k = \begin{bmatrix} \gamma_k^1 & \gamma_k^2 & \dots & \gamma_k^{n-1} & \gamma_k^n \end{bmatrix}^T$, $\gamma_k^i \in \{d_k^i, \theta_k^i\}$ is the $k$th point along the trajectory through the $n$-dimensional configuration space and $N$ is the total number of points of the motion path through the configuration space.

Manipulator motions have hardly been used in manipulator optimisation. For instance Agrawal and Veeraklaew (1997) explicitly considered manipulator motions while optimising a robot that executed a cyclic, repeated motion. But the motion planning algorithm they used is not suitable for our purpose, because collision detection was not taken into account.

In this research a heuristic A*-search algorithm was used to calculate collision-free manipulator motions represented by $\boldsymbol{\gamma}_k$, $k = 1, \dots, N$. The A*-search algorithm minimizes a measure

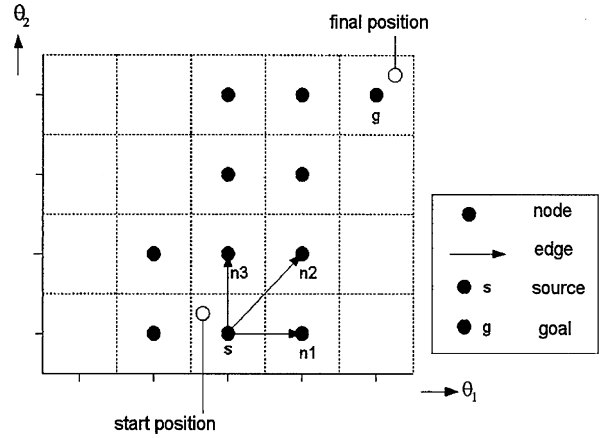


**Fig. 2 – Orthogonal and diagonal node expansion in the A*-search algorithm.**

of the length of the motion trajectory through the configuration space (Pearl, 1984; Kondo, 1991; Van Henten et al., 2003a). Assuming that the length of the motion path is equivalent to the motion time, this allowed to compare different robot designs in terms of motion time. It may appear that a max norm in Eq. (2) might have been a more appropriate measure. However, this choice does not affect the methodology and is not expected to influence the presented results significantly either.

Another interesting feature of the A*-algorithm is that it only returns a solution if one exists. This means that if a goal configuration is not reachable without collisions caused by, for instance, too large links, this design can be heavily penalised. For the robot design optimisation, the A*-search algorithm was modified to allow for both orthogonal and diagonal node expansion, as illustrated in Fig. 2, resulting in smoother motion trajectories than with orthogonal node expansion only.

#### 2.2.2. A *performance criterion measuring the dexterity of the manipulator*

Manipulator dexterity has been commonly used as a criterion for manipulator design evaluation and optimisation. Almost all dexterity measures found in the literature involve the so-called 'Jacobian' of the manipulator $\mathbf{J}$ which maps joint velocities into Cartesian velocities of, for instance, the TCP as:

$$\dot{\mathbf{x}} = \mathbf{J}(\boldsymbol{\gamma})\dot{\boldsymbol{\gamma}}. \tag{3}$$

$\mathbf{J}(\boldsymbol{\gamma})$ a $m \times n$ matrix has the form:

$$\mathbf{J}(\boldsymbol{\gamma}) = \begin{bmatrix} \frac{\partial x_1}{\partial \gamma^1} & \cdots & \frac{\partial x_1}{\partial \gamma^n} \\ \vdots & & \vdots \\ \frac{\partial x_m}{\partial \gamma^1} & \cdots & \frac{\partial x_m}{\partial \gamma^n} \end{bmatrix}, \tag{4}$$

in which $m$ is the number of degrees-of-freedom (DOF) in the Cartesian workspace and $n$ is the number of joints. The vector $x$ may contain a stack of linear and angular velocities of the TCP.

Different Jacobian based criteria have been reported in the literature and one may question which of these is most suitable for the problem under investigation. For instance Kirćanski (1994), Lee et al. (1997), Stocco et al. (1998a,b) and Yang and Chen (2000) used the ratio of singular values of the Jacobian. But a condition number defined as the ratio between the maximum and minimum singular value only compares the best motion direction with the most worse motion direction. If $m = n$, a more balanced measure of dexterity is the determinant of the Jacobian matrix:

$$\det(\mathbf{J}(\boldsymbol{\gamma})), \tag{5}$$

which was for instance used by Faraz and Payendeh (1998). Since the absolute value of the determinant of a square matrix equals the product of its singular values, the advantage of using the determinant of $\mathbf{J}$ instead of the condition number is that the determinant takes into account all singular values and the condition number just the largest and smallest one. Nakamura (1991), Kondo et al. (1993) and Chen and Burdick (1995) used a modified version of the previous dexterity measure:

$$\sqrt{\det(\mathbf{J}(\boldsymbol{\gamma})\mathbf{J}(\boldsymbol{\gamma})^T)}. \tag{6}$$

This modification allows for situations in which the number of degrees of freedom $m$ does not equal the number of links $n$ and $\mathbf{J}$ is a non-square matrix. Instead of $\det(\mathbf{J})$, $\det(\mathbf{J}^T\mathbf{J})$ and $\det(\mathbf{J}\mathbf{J}^T)$ can be calculated in case of $m > n$ and $m < n$, respectively. When the square root of these values is taken, the measure is comparable to $\det(\mathbf{J})$. Still, all the measures mentioned above depend on the number of links of the manipulator investigated: the unit of the determinant of a square $n$-dimensional Jacobian equals the unit of a single element of the Jacobian to the $n$th power. To make the unit of the measure of dexterity independent of the size of $\mathbf{J}$, the measure $\det(\mathbf{J})$ can be replaced by:

$$\sqrt[n]{\det(\mathbf{J}(\boldsymbol{\gamma}))}, \tag{7}$$

which allows for evaluating designs with different number of links $n$. There is however one disadvantage of using the determinant as a measure of dexterity; it is dependent on the physical size of the robot links. A robot with larger links will often have a higher rate of dexterity when evaluated using the determinant. This can be prevented by dividing the determinant measure by a measure of the link lengths as was done by Kim and Khosla (1993) and Han et al. (1997).

In this research, a mixture of the previously described criteria was used for evaluating the dexterity of the fruit-picking robot:

$$P_{\text{dexterity}}(\mathbf{p}) = \sqrt[2m]{\det(\mathbf{J}(\boldsymbol{\gamma}_N)\mathbf{J}(\boldsymbol{\gamma}_N)^T)}. \tag{8}$$

This measure does not explicitly account for the total link length because too long link lengths will cause collisions during motion planning and will be heavily penalised during the optimisation. The measure of Eq. (8) evaluates the contribution of all degrees of freedom. Scaling the matrix to the number of DOF's $m$ is important because the number of DOF's

may vary during the optimisation. Furthermore, the robot may be redundant so the Jacobian is in general not a square matrix. The dexterity measure will only be evaluated at the endpoint of the motion path represented by $\boldsymbol{\gamma}_N$ and not along the whole path, because, in this case, the manipulator only needs a large amount of dexterity at the endpoint to perform the picking task. The unit of this measure is the same as the unit of a singular value of matrix **J**. Only in case of combinations of both prismatic and rotational joint, the unit becomes a mixture of both physical units, because the unit of the joint angles and link offsets are not equal. In those cases, the joint angles and link offsets have to be scaled.

#### 2.2.3. *A performance criterion with two weighed objectives*

The performance criterion used in this research combines both the measure of path length $P_{\text{path}}(\mathbf{p})$ and the dexterity measure $P_{\text{dexterity}}(\mathbf{p})$ resulting in the relation:

$$P_{\text{path}}(\mathbf{p}) = \sum_{k=2}^{N} \sqrt{(\boldsymbol{\gamma}_k - \boldsymbol{\gamma}_{k-1}) A (\boldsymbol{\gamma}_k - \boldsymbol{\gamma}_{k-1})^T} + \frac{1000}{\sqrt[2m]{\det(\mathbf{J}(\boldsymbol{\gamma}_N)\mathbf{J}(\boldsymbol{\gamma}_N)^T)}}. \quad (9)$$

An optimal design combines small values of $P_{\text{path}}$ and large values of $P_{\text{dexterity}}$. Therefore, the reciprocal value of $P_{\text{dexterity}}$ was used in the combined performance criterion. The weighting factor of 1000 balances the contribution of the two different objectives to the final performance criterion. This value was determined empirically.

### 2.3. *Solution of the optimisation problem*

Once a performance criterion is defined, there are various ways to solve the optimisation problem. Because the number of free variables to be optimised rapidly grows for an increasing number of links, the number of possible kinematic designs suffers from combinatorial explosion and it will not be feasible to check all possible designs. So computing power, efficient optimisation algorithms and possibly also a reduction of the dimension of the optimisation problem is required for a solution to be obtained within a reasonable amount of time. Because we are dealing with a highly non-linear optimisation problem, the solution may contain several local minima (Paredis and Khosla, 1995). Therefore, global optimisation techniques are preferred for the solution.

In the literature several techniques have been described. Kirćanski (1994) used evaluation criteria that were amenable to analytic solution. But in many practical applications analytic solution will hardly ever be possible. Paredis and Khosla (1991,1995) used simulated annealing to find the optimal design. Genetic algorithms were used by Kim and Khosla (1993), Chen and Burdick (1995), Han et al. (1997) and Yang and Chen (2000). Stocco et al. (1998a,b, 1999) used the so-called 'culling algorithm'. This is an efficient way to solve discrete minimax problems that may occur in robot optimisation. Bergamaschi et al. (2008) compared Sequential Quadratic Programming (SQP), Genetic Algorithms (GA), Differential Evolution (DE) and Particle Swarm Optimisation (PSO). They concluded that local optima were detrimental to the performance of the SQP algorithm and that Differential Evolution was most capable in dealing with the multi-modality of the optimisation problem.

Because of the expected multi-modality, in this research the DIRECT algorithm implemented in the TOMlab package was used. This algorithm is a deterministic approach to global optimisation and includes a Lipschitzian optimisation procedure. The algorithm was described by Jones et al. (1993).

## 3. Results

In this section, three cases of the kinematic design optimisation of a cucumber-picking robot will be presented. The

**Fig. 3 – Cucumbers grown in a high-wire cultivation system.**

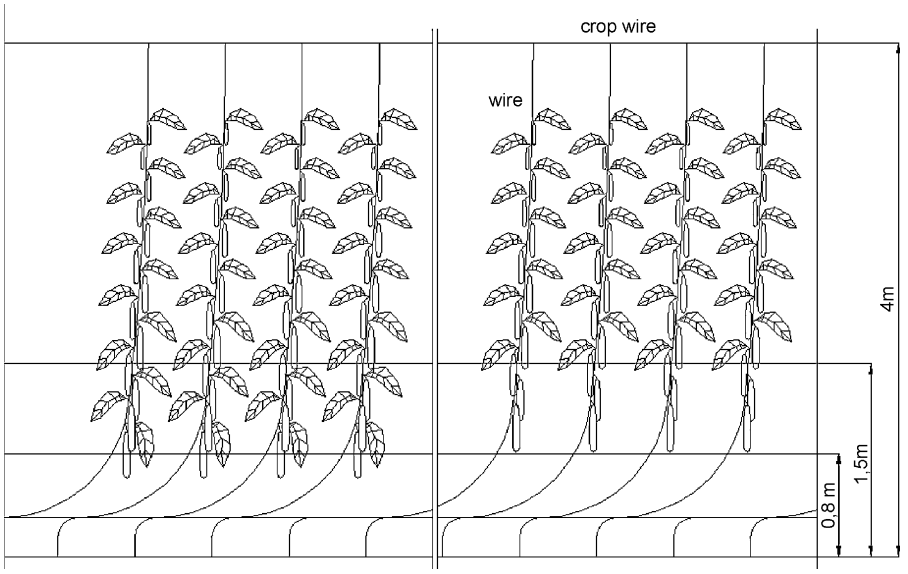


**Fig. 4 – The high-wire cultivation system before (left) and after (right) removing the leaves around the ripe cucumbers.**

harvesting robot is designed to be used for picking cucumbers in a high-wire cultivation system. Refer to Van Henten et al. (2002) for details. Fig. 3 shows pictures of this cultivation system and Fig. 4 shows a simplified two-dimensional model representation. To facilitate automatic harvesting, leaves around the harvestable cucumbers are removed. This is illustrated in Fig. 4 as well. This particular canopy structure allowed a further simplification of the working environment of the robot during the optimal design process; the working environment was simplified to a horizontal slice of the three-dimensional work space of which a top view is shown in Fig. 5.

### 3.1. *Case 1—optimisation of the link type of a three link robot*

#### 3.1.1. *The robot work cell and task description*

Fig. 5 shows the top view of an artificial greenhouse environment in which the squares represent the stems of cucumber plants, the × indicates a cucumber hanging behind the stem and I indicates the base position of a three link manipulator that is considered to be fixed exactly in front of the position of the cucumber fruit. The distance between the cucumber stems is 350 mm, which is an average value of the distances encountered in horticultural practice. The manipulator should be able to move from a straight configuration in the path (parallel to the plants) to the cucumber behind the plant without hitting the cucumber stems. This artificial scene represents one of the most difficult cases encountered in cucumber harvesting in practice, *i.e.* reaching for a cucumber hanging behind the stem of the plant as seen from the base position of the manipulator.

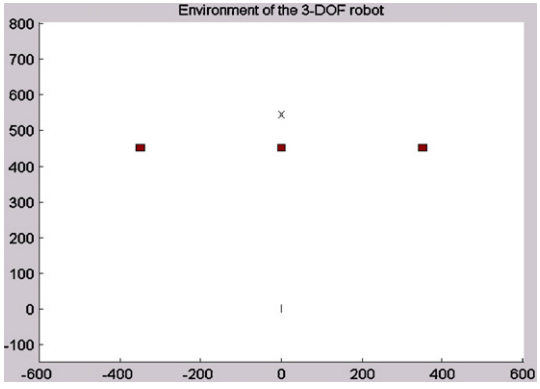


**Fig. 5 – Top view of an abstraction of the greenhouse working environment; squares: stems of cucumber plants, ×: a cucumber and I: fixed base position of manipulator in path.**

#### 3.1.2. *Definition of the optimisation problem*

To reduce complexity, in this case study the amount of links was fixed at three. Only motions in the horizontal plane were considered. For vertical motions a prismatic joint was considered to be used which was not subject to optimisation. To avoid redundancy, the picking angle was enforced to be 135° with respect to the x-axis. The task was to move the tool-centre-point of the robot from the stretched position to the goal position, without collisions.

Seven different configurations of three-link manipulators, *i.e.* RRR, RRP, RPR, RPP, PRR, PRP, PPR, were individually optimised. Here 'R' refers to a rotational joint and 'P' refers to a prismatic joint. The PPP-configuration was not optimised because it did not allow for a smooth approach of the cucumber fruit along a 135° angle. The search space of the optimisation consisted of the link parameters of the three-link robot. In case of a prismatic joint, a telescopic one was used. In case of a revolutionary joint, the link length $d$ was subjected to optimisation. In case of a prismatic joint, the twist angle $\alpha$ was optimised. Because the optimisation took place in a two-dimensional workspace, the link offset $a$ was fixed at a value of zero and not optimised. The rotation limits of a rotational joint were 360° (positive and negative) and the translation limits on a prismatic link were 50 and 600 mm. The

**Table 1 – Link parameters and corresponding costs of the optimal three-link robot designs.**

| Robot type | Link parameter | Link parameter | Link parameter | Performance P |
|---|---|---|---|---|
| RRR | $d_1 = 380$ mm | $d_2 = 251$ mm | $d_3 = 95$ mm | 133 |
| RRP | $d_1 = 411$ mm | $d_2 = 103$ mm | $d_3 = 50$ mm | 86 |
| RPR | $d_1 = 367$ mm | $\alpha_2 = 60°$ | $d_3 = 325$ mm | 135 |
| RPP | $d_1 = 399$ mm | $\alpha_2 = -3°$ | $\alpha_3 = 52°$ | 153 |
| PRR | $\alpha_1 = 80°$ | $d_2 = 350$ mm | $d_3 = 331$ mm | 135 |
| PRP | $\alpha_1 = 89°$ | $d_2 = 400$ mm | $\alpha_3 = 51°$ | 152 |
| PPR | $\alpha_1 = 80°$ | $\alpha_2 = -40°$ | $d_3 = 368$ mm | 433 |

twist angle of a prismatic joint was limited to plus or minus 90°.

### 3.1.3. *Results*

Results of the optimisation are presented in Table 1. The design with the lowest value of the performance criterion $P$ is the RRP robot. The motion of this manipulator in the workspace is shown in Fig. 6. It is striking that the prismatic joint is not translated at all. This is caused by the fact that a change in position of the TCP caused by the rotation by one unit of a long revolute joint is much larger than that of a translation of a prismatic joint by one unit. Therefore, the optimal design contains long revolute joints and the prismatic joint is hardly used. This suggests an appropriate scaling when different joint types are simultaneously used. The results also show that the manipulator tends to skim along the surface of obstacles. In the optimisation, the only definite requirement is that the motion should be collision-free. This requirement was satisfied. But in practice it may be better to incorporate some additional safety measures to deal with sensor inaccuracies. In the design optimisation this can be achieved by either extending the size of the objects or including a measure for the distance between the manipulator and the obstacles.

The number of function evaluations used by the optimisation algorithm varied between 201 for the RRR-design to 325 for the RRP-design with an average of 256. The average computation time of an optimisation was about 4 h on a Pentium III machine.

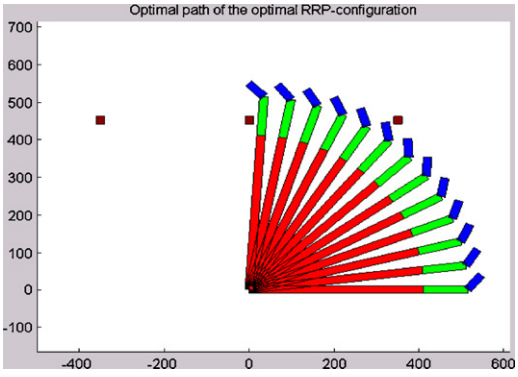


**Fig. 6 – Motion of the three link RRP manipulator from a location in the path to the cucumber hanging behind the stem.**

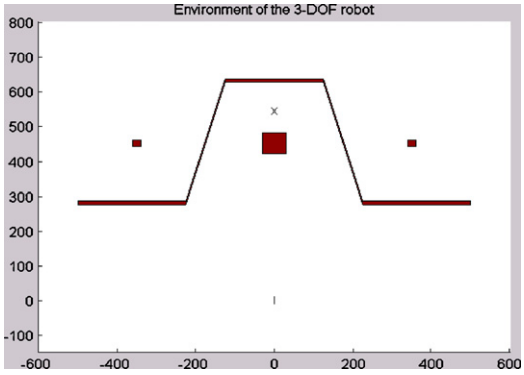


**Fig. 7 – Top view of an abstraction of the greenhouse working environment; squares: stems of cucumber plants, ×: a cucumber and I: fixed base position of manipulator in path.**

## 3.2. *Case 2—optimisation of three-link PPR, PRR and PRP manipulators for four different tasks*

Building on the results of Case 1, in Case 2 some aspects of the optimisation problem were modified.

### 3.2.1. *The robot work cell and task description*

The working environment of the robot was adjusted as shown in Fig. 7. The adaptations had to do with the observation in Case 1 that the manipulator skimmed along the surface of obstacles because the distance to obstacles was not penalised. To incorporate more safety, the cucumber stem carrying the fruit to be picked was modelled larger than its real size. Additionally, virtual walls were placed around this stem to force the manipulator away from the other obstacles. Also the task definition was modified. In this case the robot was optimised for picking cucumbers not just hanging behind the stem, but hanging behind, in front, to the left and to the right of the stem. This is considered to be a more realistic representation. The four positions of cucumbers were (−100; 450), (100; 450), (0; 350) and (0; 550), denoted as 'left', 'right' and 'in front' and 'behind', respectively. In this case picking angles were not predefined because in reality no picking angle will be enforced.

### 3.2.2. *Definition of the optimisation problem*

In this case, the search space was reduced by setting the first joint to be a prismatic one. This problem reduction was motivated by the fact that in practice a robot will move over a path parallel to the plants. This motion is effectively

**Table 2 – Dimensions of the Eshed MK2 robot.**

| Link nr. | Length | Width | Height |
|---|---|---|---|
| 1 | 227 mm | 172 mm | 320 mm |
| 2 | 402 mm | 107 mm | 118 mm |
| 3 | 337 mm | 75 mm | 93 mm |
| 4 | 143 mm | 67 mm | 69 mm |
| 5 | 165 mm (incl. gripper) | 65 mm | 65 mm |

*Note*: The width and height are measured at the middle of the link.

represented by a prismatic first link. The search space was enlarged by including the position of the robot base relative to the cucumber plant as an independent variable in the optimisation. In Case 1 this point was fixed exactly in front of the cucumber plant. Also, in this case, realistic values were used for the width or thickness of each link. Case 1 showed that manipulator motions tend to graze obstacles, therefore it is important that the values of the width of the robot links are realistic. To get an idea of realistic values of a robot link, the dimensions of the links of the MK2-robot (Eshed Robotec, 1994) were measured. The length of a link is the size in the direction of the z-axis when the axes are assigned according to the Denavit–Hartenberg convention (see McKerrow, 1998). The height and width are the size in the direction of the *x*- and *y*-axis, respectively. The given values are the physical dimensions of the links; not the Denavit–Hartenberg parameters. The height in Table 2 was most comparable to the width of the link s in the optimisation. Especially links 2 till 4 were realistic, because the first and last link rotate around their axis (*i.e.* the link length in the Denavit–Hartenberg parameters is zero). Based on these values, the first link width was set at 110 mm, the second at 90 mm and the third at 70 mm. The width was assumed to be independent of the link length.

To achieve a more balanced trade-off between prismatic links and rotational links, the costs corresponding to a one-degree-rotation were made unequal to a one-millimetre-translation. The ratio between them was based on the maximum movements of both link types. In general, the maximum required rotation of one link is about 90°; the maximum required translation of a prismatic link is estimated to be about 400 mm. Assuming the costs of a full movement, *i.e.* the time needed to perform the task, to be the same, the costs of an one mm translation were fixed at one fifth of the costs of an one degree of a rotation.

With these preliminaries, for each picking position, the three robot designs PPR, PRR and PRP were individually optimised, resulting in optimal values of the position of the base, the parameters of these links (link length or twist angle) and the picking angle.

**Table 3 – Performance *P* corresponding to the optimal design at different goal locations.**

| Robot type | 'Behind' | 'Left' | 'Right' | 'In front' |
|---|---|---|---|---|
| 'PPR' | 204 | 168 | 158 | 130 |
| 'PRR' | 183 | 84 | 62 | 40 |
| 'PRP' | 225 | 106 | 80 | 67 |

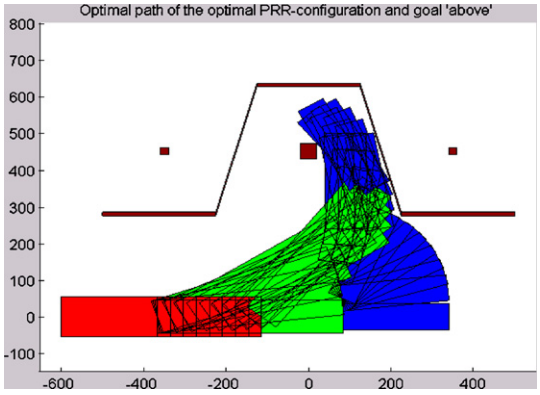


**Fig. 8 – Manipulator motion of the optimal PRR robot to the 'behind' goal.**

#### 3.2.3. *Results*

The results of the robot optimisations are summarised in Table 3. These results show that the PRR robot achieved the best performance, regardless of the location of the goal. The parameters of the optimal PRR designs differed for the different goal positions as is illustrated in Table 4. The negative values of the robots base indicate that the robot's initial position was located left from the picking position. The manipulator motion of the optimal PRR-robot design to the 'behind' goal is plotted in Fig. 8. The computation of each optimal robot type for one goal position took about 9 h.

### 3.3. *Case 3—optimisation of a single PRR manipulator design for four different tasks*

Because in reality one robot with a fixed structure has to pick cucumbers at all locations, a different design for each goal position is not feasible. Therefore in this case one robot was optimised for all four tasks.

#### 3.3.1. *The robot work cell and task description*

The work cell and task description were equal to Case 2.

#### 3.3.2. *Definition of the optimisation problem*

To save computation time only the PRR robot was optimised, because this robot appeared to be the best configuration for

**Table 4 – Overview of the optimal PRR design parameters for the different goal positions.**

| Goal position | Base | Length link 2 | Length link 3 | Picking angle |
|---|---|---|---|---|
| 'Behind' | −367 mm | 450 mm | 258 mm | 130° |
| 'Left' | −200 mm | 317 mm | 175 mm | 70° |
| 'Right' | −529 mm | 492 mm | 204 mm | 46° |
| 'In front' | −599 mm | 396 mm | 300 mm | 34° |

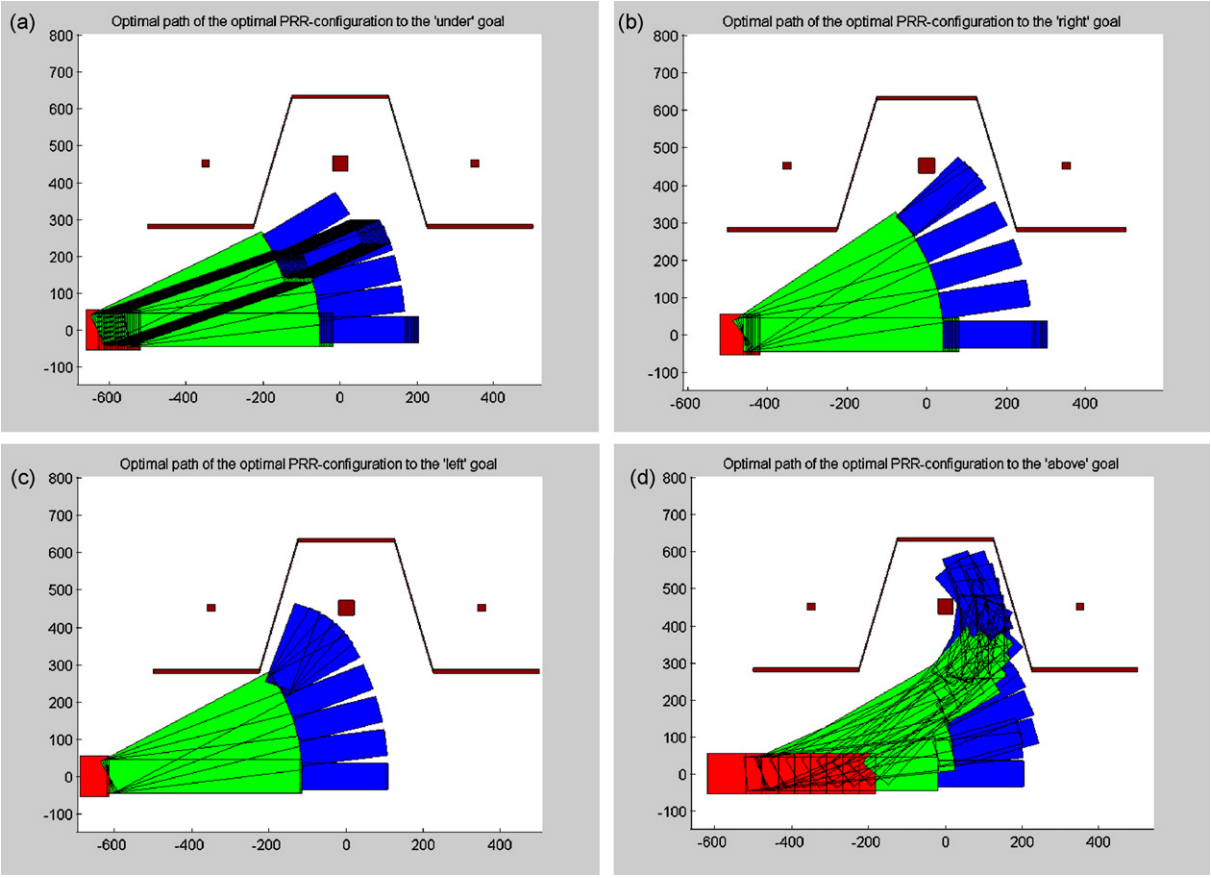


**Fig. 9 – Manipulator motions of the optimal PRR robot to four picking positions in front of the cucumber stem (a), to the right of the stem (b), to the left of the stem (c) and behind the cucumber stem (d).**

each separate goal position. Therefore, it was assumed that this robot would also be the best one for all goal positions. In this case study, the design performance criterion was set equal to the sum of the performances of the PRR-robot moving to all of the four goals. The computation time was limited by setting the picking angle to the optimal picking angle in the one-goal optimisation as shown in Table 3. So, in this optimisation, the search space consisted of the position of the base and the length of the two links.

#### 3.3.3. *Results*

For this three link PRR-robot the best position of the base was found to be 519 mm left of the position of the cucumber stem. Optimal lengths of the second and third link were found to be 499 and 223 mm, respectively. This optimisation took 3 days computation time on a Pentium III machine.

In Fig. 9, manipulator motions of the optimal PRR robot to all four picking positions are shown.

## 4. Final discussion and concluding remarks

### 4.1. *The manipulator design*

It seems a big step from the previously described optimisation in a two-dimensional world to the real three-dimensional world. But as shown in Figs. 3 and 4, the high wire cultivation system mainly contains vertical elements in the area where cucumbers are picked. The stems of the cucumber plants and the cucumber fruits are all more or less vertically oriented. The main exceptions to this rule are the leaves. But these are removed in the harvest area. So, the three-link PRR manipulator seems to be suited for the task of cucumber picking in this environment if cucumbers are all hanging at the same height, which is not the case in practice. Therefore, extension of the three-link manipulator with a fourth prismatic link is needed and suffices for picking cucumbers at various heights in the harvest region. This will yield for example a 4 link PPRR manipulator in which the cart that moves between the rows is used as a first prismatic link on top of which a vertical prismatic link and two rotational links in the horizontal plane are mounted. Picking vertically oriented cucumbers only requires 4 degrees-of-freedom, *i.e.* three translations and one rotation around the vertical axis. The optimised 4 link PPRR robot meets this requirement. The 7 link P6R manipulator originally used in the prototype cucumber harvester however had six degrees-of-freedom of which two frame rotations were never used. It is in that direction where a considerable complexity and cost reduction of the kinematic design can be achieved, when application in horticultural practice is considered.

Computational complexity did not allow for a fully autonomous search over the full parameter range in this optimisation

process, including variations of number of links, link types, link offset, link twist, rotational or prismatic link limits and link volumes. Using pragmatic arguments, the search space was limited and the question is whether or not this has resulted in trivial results. Albeit this potential limitation of this research, the following three observations strongly support the approach and the results presented in this paper. First of all, Cartesian manipulators are considered very simple and effective manipulator structures in many agricultural and non-agricultural applications. As can be seen in both Tables 1 and 3, the associated PPR manipulator exhibits a very poor performance in view of the design objectives used in this research and a PPPR manipulator built on that idea would be poor choice. So, the results in this paper contradict a commonly used engineering approach to the design of kinematic structures. Secondly, in an independent research line, Song et al. (2007) optimised a 4R manipulator for harvesting egg plants, a task quite similar as cucumber harvesting. Since this manipulator has to be mounted on top of a cart to move along side the plants, the overall kinematic chain would be of the P4R type. Interestingly enough Hayashi et al. (2001) used a P5R kinematic chain to harvest egg-plants. Though other objectives might have played a role in the manipulator choice of Song et al. (2007) and Hayashi et al. (2001), apparently the 2P2R manipulator proposed in this research is a non-trivial minimal solution not earlier reported in the literature on fruit harvesting. Thirdly, there is no way in which manual calculations or simple simulations based on CAD drawings might have lead to similar design results whilst accounting for design objectives like motion time and dexterity.

Clearly, the greenhouse scenes used in this research are abstractions of reality. As shown in Figs. 3 and 4, for this particular canopy structure, this abstraction was considered justified. Further research is required to exploit the potential of this design methodology under more realistic greenhouse conditions and for alternative tasks and canopy structures. This might lead to an understanding of what kind of manipulator structure is most suitable for particular tasks in particular canopy structures. These insights might guide design engineers in their work on agro-robotics.

### 4.2. *The methodology*

Although computationally expensive, the methodology used in this research was found to be powerful and offered an objective way to evaluate and optimise the kinematic structure of a robot to be used for cucumber harvesting. Expansion of this methodology is required to achieve a fully autonomous search over the full parameter range during this optimisation process, including variations of number of links, link types, link offset, link twist, rotational or prismatic link limits and link volumes.

## Acknowledgements

This project was partially funded by the Dutch Ministry of Agriculture, Nature and Food Quality. The authors gratefully acknowledge the critical review of two anonymous referees.

## REFERENCES

Agrawal, S.K., Veeraklaew, T., 1997. Designing robots for optimal performance during repetitive motion. In: Proceedings of the 1997 IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, USA, pp. 2178–2183.

Bergamaschi, P.R., Saramago, S.F.P., Coelho, L.S., 2008. Comparative study of SQP and metaheuristics for robotic manipulator design. Applied Numerical Mathematics 58, 1396–1412.

Bontsema, J., Van Kollenburg-Crisan, L.M., Van Henten, E.J., 1999. Automatic harvesting of vegetable fruits. In: Proceedings of the BRAIN-IAM International Symposium, 24 November 1999, Omiya, Japan.

Carbone, G., Ottaviano, E., Ceccarelli, M., 2007. An optimum design procedure for both serial and parallel manipulators. Journal of Mechanical Engineering Science 221, 829–843.

Ceccarelli, M., Lanni, C., 2004. A multi-objective optimum design of general 3R manipulators for prescribed workspace limits. Mechanism and Machine Theory 39, 119–132.

Chen, I.-M., Burdick, J.W., 1995. Determining Task Optimal Modular Robot Assembly Configurations. Internal report. Department of Mechanical Engineering, Caltech, Pasadena, USA.

Craig, J.J., 1989. Introduction to Robotics. Addison-Wesley, Massachusetts, USA.

Eshed Robotec, 1994. Performer MK2, User's Manual. Eshed Robotec Ltd., Duiven, The Netherlands.

Faraz, A., Payendeh, S., 1998. A robotic case study: optimal design for laparoscopic position stands. The International Journal of Robotics Research 17, 986–995.

Han, J., Chung, W.K., Youm, Y., Kim, S.H., 1997. Task based design of modular robot manipulator using efficient genetic algorithm. In: Proceedings of the 1997 IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, USA, pp. 507–512.

Hayashi, S., Ganno, K., Ishii, Y., Tanaka, I., 2001. Basic harvesting experiment of eggplant using a robotic system. In: Zude, M., Herold, B., Geyer, M. (Eds.), Proceedings of the 6th International Symposium on Fruit, Nut and Vegetable Production Engineering. Institut für Agrartechnik Bornim e.V. (ATB), Potsdam, Land Brandenburg, Federal Republic of Germany, pp. 221–226.

Jones, D.R., Perttunen, C.D., Stuckman, B.E., 1993. Lipschitzian optimisation without the Lipschitz constant. Journal of Optimization Theory and Applications 79, 157–181.

Kim, J.O., Khosla, P.K., 1993. A formulation for task based design of robot manipulators. In: Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems, Yokohama, Japan, pp. 2310–2317.

Kirćanski, M.V., 1994. Robotic isotropy and optimal robot design of planar manipulators. In: Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 1100–1105.

Kondo, K., 1991. Motion planning with six degrees of freedom by multistrategic bidirectional heuristic free-space enumeration. IEEE Transactions on Robotics and Automation 7, 267–277.

Kondo, N., Monta, M., Shibano, Y., Mohri, K., 1993. Basic mechanism of robot adapted to physical properties of tomato plant. In: Proceedings of the International Conference for Agricultural Machinery & Process Engineering, Seoul, Korea, pp. 840–849.

Lee, S.H., Yi, B.J., Kwak, Y.K., 1997. Optimal kinematic design of an anthropomorphic robot module with redundant actuators. Mechatronics 7, 443–464.

McKerrow, P.J., 1998. Introduction to Robotics. Addison-Wesley, Massachusetts, USA.
Nakamura, Y., 1991. Advanced Robotics: Redundancy and Optimization. Addison-Wesley, Massachusetts, USA.
Paredis, C.J.J., Khosla, P.K., 1991. An approach for mapping kinematic task specifications into a manipulator design. In: Proceedings of the 1991 IEEE International Conference on Robotics and Automation, pp. 556–561.
Paredis, C.J.J., Khosla, P.K., 1995. Design of modular fault tolerant manipulators. In: Goldberg, K., Halperin, D., Latombe, J.-C., Wilson, R. (Eds.), Algorithmic Foundations of Robotics. Peters Ltd., Wellesley, Massachusetts, USA, pp. 371–383.
Pearl, J., 1984. Heuristics. Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley, Reading, Massachusetts, USA.
Song, J., Sun, X., Zhang, T., Zhang, B., Xu, L., 2007. Design optimisation and simulation of structure parameters of an eggplant picking robot. New Zealand Journal of Agricultural Research 50, 959–964.
Stocco, L., Slacudean, S.E., Sassani, F., 1998a. Matrix normalization for optimal robot design. In: IEEE International Conference on Robotics and Automation, Leuven, Belgium, pp. 1–6.
Stocco, L., Salcudean, S.E., Sassani, F., 1998b. Fast constrained global minimax optimization of robot parameters. Robotica 16, 595–605.
Stocco, L., Slacudean, S.E., Sassani, F., 1999. On the use of scaling matrices for task-specific robot design. IEEE Transactions on Robotics and Automation 15, 958–965.
Van Henten, E.J., Hemming, J., Van Tuijl, B.A.J., Kornet, J.G., Meuleman, J., Bontsema, J., Van Os, E.A., 2002. An autonomous robot for harvesting cucumbers in greenhouses. Autonomous Robots 13, 241–258.
Van Henten, E.J., Hemming, J., Van Tuijl, B.A.J., Kornet, J.G., Bontsema, J., 2003a. Collision-free motion planning for a cucumber picking robot. Biosystems Engineering 86, 135–144.
Van Henten, E.J., Van Tuijl, B.A.J., Hemming, J., Kornet, J.G., Bontsema, J., Van Os, E.A., 2003b. Field test of an autonomous cucumber picking robot. Biosystems Engineering 86, 305–313.
Van Willigenburg, L.G., Hol, C.W.J., Van Henten, E.J., 2004. On-line near minimum-time path planning and control of an industrial robot for picking fruits. Computers and Electronics in Agriculture 44, 223–237.
Yang, G., Chen, I.M., 2000. Task-based optimization of modular robot configurations: minimized degree-of-freedom approach. Mechanism and Machine Theory 35, 517–540.