BIOSYSTEMS ENGINEERING 146 (2016) 85–97

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/issn/15375110

Special Issue: Robotic Agriculture

Research Paper

Analysis of a motion planning problem for sweet-pepper harvesting in a dense obstacle environment

C. Wouter Bac [a,b,*], Tim Roorda [b], Roi Reshef [c], Sigal Berman [c], Jochen Hemming [a], Eldert J. van Henten [b]

[a] Wageningen UR Greenhouse Horticulture, Wageningen University and Research Centre, Droevendaalsesteeg 1, 6708 PB Wageningen, The Netherlands

[b] Farm Technology Group, Wageningen University and Research Centre, Droevendaalsesteeg 1, 6708 PB Wageningen, The Netherlands

[c] Department of Industrial Engineering and Management, Ben-Gurion University of the Negev, P.O. Box 653, 84105 Beer-Sheva, Israel

ARTICLE INFO

Article history:
Published online 17 July 2015

Keywords:
Motion planning
Agricultural robot
Sensitivity analysis
Rapidly exploring random trees
Grasp pose

To reach a fruit in an obstacle-dense crop environment, robotic fruit harvesting requires a collision-free motion of the manipulator and end-effector. A novel two-part analysis was conducted of a sweet-pepper harvesting robot based on data of fruit (N = 158) and stem locations collected from a greenhouse. The first part of the analysis compared two methods of selecting the azimuth angle of the end-effector. The new 'constrained-azimuth' method avoided risky paths and achieved a motion planning success similar to the 'full-azimuth' method. In the second part, a sensitivity analysis was conducted for five parameters specifying the crop (stem spacing and fruit location), the robot (end-effector dimensions and robot position) and the planning algorithm, to evaluate their effect on successfully finding a collision-free goal configuration and path. Reducing end-effector dimensions and widening stem spacing are promising research directions because they significantly improved goal configuration success, from 63% to 84%. However, the fruit location at the stem is the strongest influencing parameter and therefore provides an incentive to train or breed plants that develop more fruit at the front side of the plant stem. The two analyses may serve as useful tools to study motion planning problems in a dense obstacle environment.

* *Corresponding author*. Wageningen UR Greenhouse Horticulture, Wageningen University and Research Centre, Droevendaalsesteeg 1, 6708 PB Wageningen, The Netherlands. Tel.: +31 317 481880; fax: +31 317 418094.
E-mail address: wouter.bac@wur.nl (C.W. Bac).

http://dx.doi.org/10.1016/j.biosystemseng.2015.07.004
1537-5110/

1. Introduction

Robotic fruit harvesting is motivated by social, economic, environmental and food quality aspects (Lewis, Watts, & Nagpal, 1983). The need to reduce the production costs is a prominent motivator to automate harvesting in high-value crops such as tomato, cucumber or sweet pepper. Harvesting these crops is challenging given the complex working environment for the robot that comprises varying poses, sizes, shapes and colours of fruit and other objects (Bac, Van Henten, Hemming, & Edan, 2014). Furthermore, a target fruit can be surrounded by densely spaced obstacles, i.e. plant parts, support wires, and fruit clusters. The manipulator and end-effector need to avoid these obstacles to successfully approach a target fruit and prevent damages to the plant or nearby fruit. Successful planning and implementation of the manipulator motion determines the feasibility of robotic harvesting, and therefore it could be useful to understand the influence of various parameters affecting motion planning success.

Motion planning has received attention in literature on autonomous crop harvesting. Work reported ranges from simple manipulator control and kinematics to advanced motion planning (Edan, Rogozin, Flash, & Miles, 2000; Guo, Zhao, Ji, & Xia, 2010; Hannan & Burks, 2004; Kondo, Nishitsuji, Ling, & Ting, 1996; Kondo et al., 2007; Liang & Wang, 2010; Sakai, Iida, Osuka, & Umeda, 2008; Sivaraman & Burks, 2006; Van Henten, Hemming, Van Tuijl, Kornet, & Bontsema, 2003; Van Henten, Schenk, van Willigenburg, Meuleman, & Barreiro, 2010; Van Willigenburg, Hol, & Van Henten, 2004). Some papers conducted an in-depth analysis of the manipulator kinematics (Sivaraman & Burks, 2006; Van Henten, Van't Slot, Hol, & Van Willigenburg, 2009). The influence of the crop structure on the success rate of generating successful harvest grasp poses for the manipulator was also analysed (Van Henten et al., 2010). However, to date, analyses of motion planning success in relation to various parameters involved (design of end-effector, planning algorithm, crop environment) have not been reported.

The current work attempts to fill this gap for the field of fruit harvesting, by an analysis for sweet pepper. Additionally, the paper may contribute to the field of robotics at large. Sensitivity analysis is not a commonly used instrument for performance analysis of robotic systems. Some examples include sensitivity analysis focussing on improving trajectories (de Luca, Lanari, & Oriolo, 1991), comparing robot designs (Tannous, Caro, & Goldsztejn, 2014), or positioning errors of the end-effector (Zhang, Ba, & Mu, 2012). These studies mostly concentrated on the influence of robot parameters. The current research will extend the analysis beyond the robot design to consider parameters of the planning algorithm, and of the crop, i.e. the working environment of the harvesting robot.

To provide a realistic case study for the analysis of motion planning for autonomous sweet pepper harvesting, measurements were collected of the three-dimensional (3D) shape of a crop grown in a greenhouse. In the 3D working environment generated with these data, simulations were performed with a model of the harvesting robot developed for the crop. The analysis contained essentially two steps. Firstly, the analysis focussed on the effect of the angle of approach of the end-effector, to the fruit, on the motion planning success. Then, secondly, building on these results, a sensitivity analysis was performed in which the effect of end-effector size, the robot position, a parameter of the motion planning algorithm as well as parameters determining the crop environment (fruit location and stem spacing), were evaluated. The methodology and results may also provide insights for motion planning in other high-value crops.

2. Components of the motion planning problem

The robot used was developed as part of the research project CROPS 'Clever Robots for Crops' (CROPS, 2014). Section 2.1 details this robot, Section 2.2 describes the crop row used and Section 2.3 describes the algorithms used to address the motion planning problem.

2.1. Robot

The robot consisted of a platform, a manipulator and an end-effector (Fig. 1). The platform developed (Jentjens Machinetechniek B.V., The Netherlands) used the heating pipes of the greenhouse as a rail system to travel along the crop row. Hemming et al. (2014) describe more details about the robot.

The manipulator (Technical University Munich, Germany) comprised nine degrees-of-freedom (DOF). The first joint was prismatic, whereas the other joints were rotational. Transformations were derived between the links (Table 1), using the modified Denavit–Hartenberg convention introduced by Craig (2004), where a_{i-1} refers to the link length (m) of link i, α_{i-1} refers to the link twist (rad) of link i, d_i refers to the link offset (m) of link i, and θ_i refers to the joint angle (rad) at axis i (Table 1).

Manipulators used for fruit harvesting typically consist of two to seven DOF (Bac, Van Henten, et al., 2014). An advantage of using many DOFs is an improved target-reachability in cluttered environments. However, the disadvantage is that common motion planning algorithms have difficulty in finding a collision-free path within a reasonable time. Furthermore, additional DOFs decrease speed (the stiffness reduces, which imposes lower motor torques to avoid vibrations) and reliability, and increase the cost of the manipulator.

The end-effector grasps the fruit using a suction cup and subsequently cuts the peduncle (Fig. 2), which is the connecting stem between the fruit and main stem. The bounding box of the end-effector had dimensions of 170 mm (width), 260 mm (length), and 110 mm (height). More details about the end-effector are described in the patent (Van Tuijl & Wais, 2014).

2.2. Crop row

2.2.1. Measurements in the greenhouse

A total of 60 plant stems, and 165 attached fruit, were recorded from a crop row (sweet-pepper cultivar: Waltz) grown in a commercial greenhouse in the Netherlands. Only hard obstacles (fruit, stem, support wire, robot) that may cause

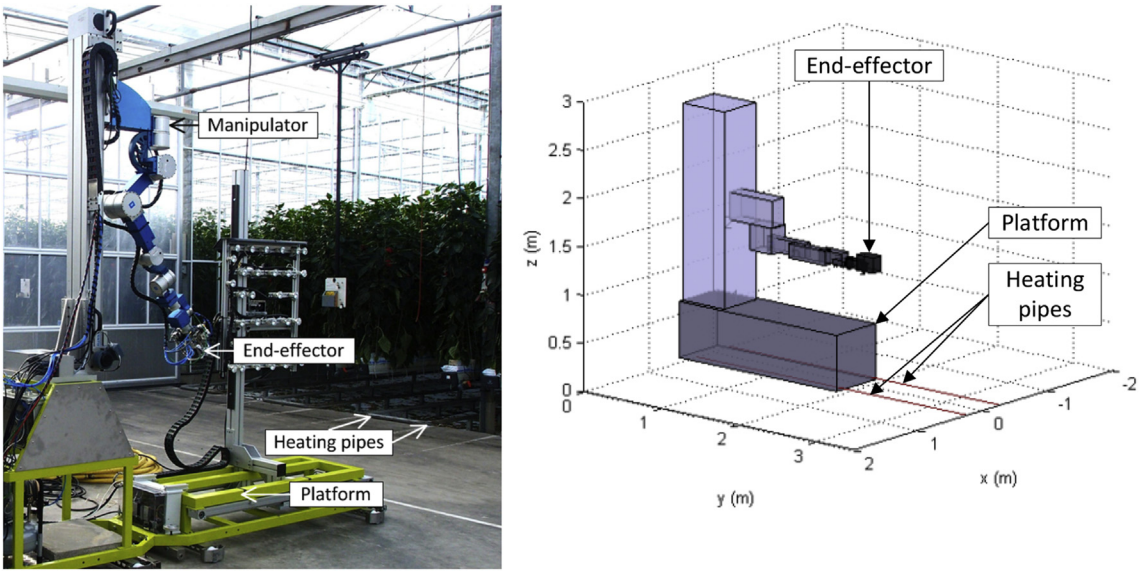

Fig. 1 – Photo of the platform, manipulator and end-effector (left), and a model representation of the manipulator based on bounding boxes (right). The robot platform travels in the y-direction (along the crop row) and uses the heating pipes as a rail system.

damage to the crop or robot were measured and mapped. Soft obstacles (leaves, petioles) can be pushed aside (Bac, Hemming, & Henten, 2013) and they were therefore disregarded. Stems and support wires were located using a stereo-vision method described in a previous article (Bac, Hemming, & van Henten, 2014). This previous article discusses an issue of inaccurate localisation caused by sequential image acquisition, which we resolved by using two cameras. As a result, the localisation accuracy was ±4 mm in the depth direction and ±0.5 mm in width and height.

Table 1 – Denavit–Hartenberg parameters describing transformations between links of the manipulator.

Link i	a_{i-1} (m)	α_{i-1} (rad)	d_i (m)	θ_i (rad)
1	0.115	0	0.796	$\pi/2 + 0.2352$
2	0.480	0	−0.200	−0.2352
3	0	$\pi/2$	0	0
4	0.350	π	0	0
5	0.350	π	0	0
6	0.180	$-\pi/2$	0	$-\pi/2$
7	0	$-\pi/2$	0.150	0
8	0	$\pi/2$	0	0
9	0	$-\pi/2$	0.065	0

a_{i-1} = link length; α_{i-1} = link twist; d_i = link offset; θ_i = joint angle.

The following fruit parameters were measured manually (measurement resolution is indicated in parenthesis):

- Height of the fruit centre relative to the robot platform (10 mm) (Fig. 3);
- Azimuth angle φ_{Fruit} with respect to the stem (10°), to determine if the fruit is located at the front, left or right side of the stem (Fig. 3);
- Fruit width (1 mm);
- Fruit height (1 mm);
- Offset, if present, between the fruit surface and stem (10 mm) (Fig. 3).

During preliminary testing in the greenhouse, it was found that the end-effector could successfully detach a fruit if a

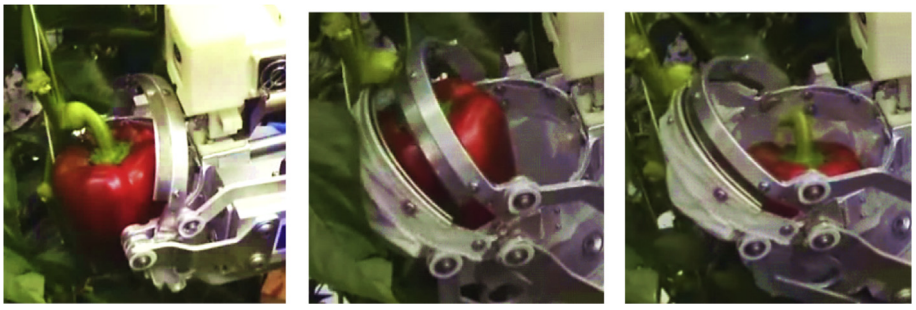

Fig. 2 – The end-effector sucks the fruit surface, (left), cuts the peduncle (middle) and disposes the sweet-pepper, in a tube (right).

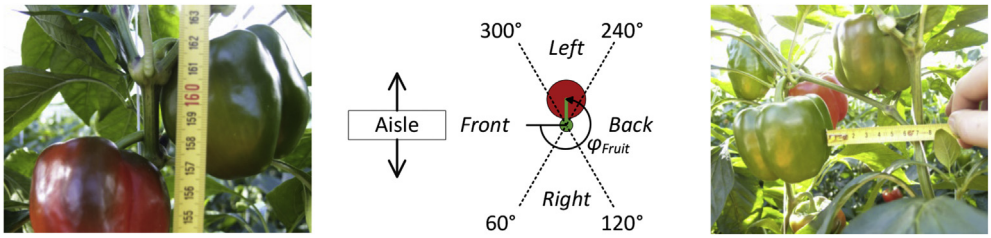

Fig. 3 – Manual measurements: height of the fruit centre relative to the robot platform (left), azimuth angle with respect to the stem (centre), and offset between fruit surface and stem (right).

collision-free azimuth angle was selected for the end-effector, whereas the skew and elevation angle of the fruit hardly influenced detachment success. Thus, the elevation and skew angle were not measured and fruit were assumed to be oriented vertically, i.e. orthogonal to the floor. Correspondingly, all end-effector poses were taken in the horizontal plane and only the azimuth angle of the end-effector was varied. In addition, the position and orientation of the peduncle was not measured because the end-effector does not require this information for successfully cutting the peduncle. These assumptions concerning the peduncle, and the skew and elevation angle may have to be reconsidered if a different type of end-effector is used in future work.

2.2.2. *3D mapping*

The measurements recorded in the greenhouse were used to generate a 3D virtual environment, using Matlab® 2012a (Mathworks Inc., Natick, MA, USA). The 3D environment consisted of the robot, plant stems, attached fruit, and a support wire that was twisted around each stem (Fig. 4).

The workspace of the manipulator was limited to a height range of about 1 m. Out of the total 165 fruit, seven were hanging below this height range (e.g. the lowest fruit in Fig. 4). Therefore, only the 158 accessible fruit were analysed.

2.3. *Algorithms*

A sequence of algorithms (Fig. 5) was implemented in Matlab®, consisting of three steps: calculation of the start and goal configuration (Section 2.3.2), path planning (Section 2.3.3) and path smoothing (Section 2.3.4). Each step uses a collision detection module (Section 2.3.1). The algorithms were run on a computer with an Intel Core i5 CPU 2.4 GHz processor with 4 GB of memory using Windows 7 as the operating system.

2.3.1. *Collision detection*

The collision detection algorithm was developed based on distance computation between shape primitives describing the manipulator and end-effector, wires, stem segments and fruit. For self-collision detection of the manipulator and end-effector, each link was modelled by a bounding box. To reduce the number of collision checks, and to generate a small safety zone around a box, octrees were implemented (Meagher, 1982) in which collisions are checked at the child level only if there is a collision at the parent level. Each

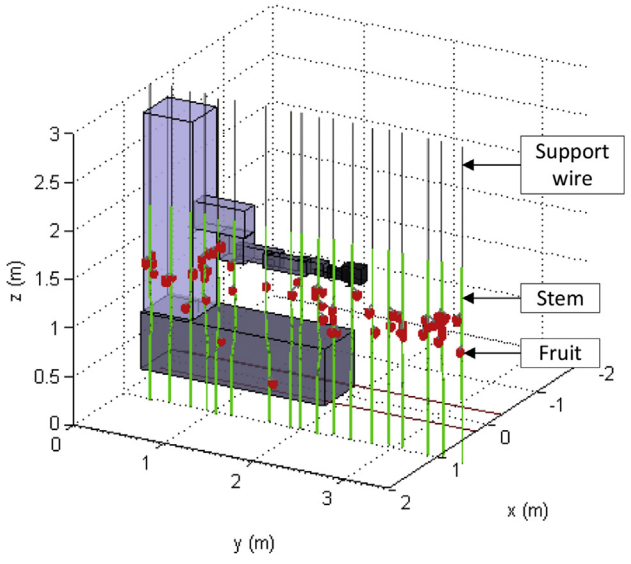

Fig. 4 – 3D virtual environment consisting of the manipulator, end-effector, plant stems, attached fruit, and a support wire that was twisted around each stem.

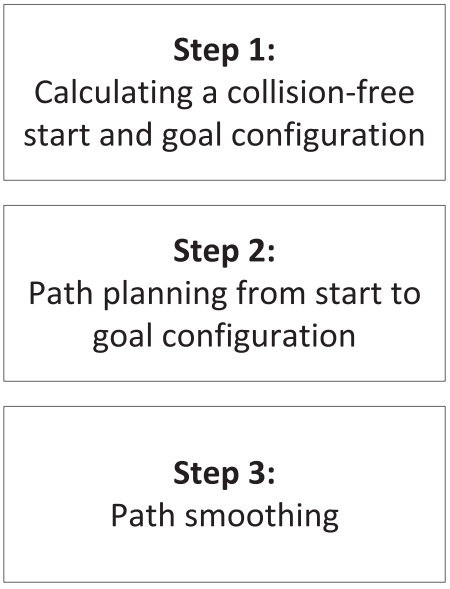

Fig. 5 – Sequence of algorithms for motion planning.

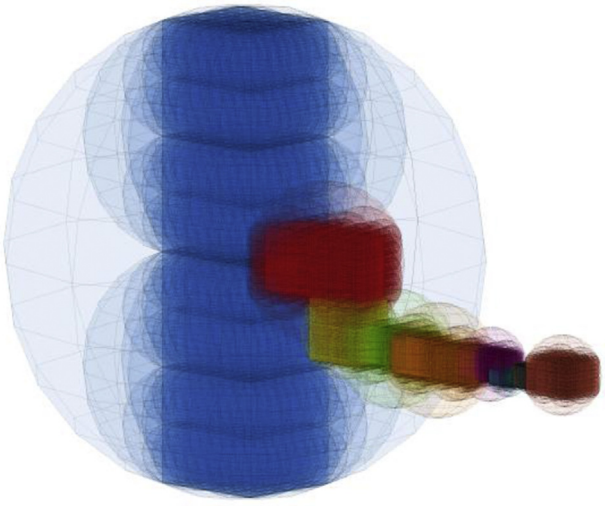

Fig. 6 – Bounding box representation of the manipulator with octrees of spheres used for collision detection. Colour coding is used to distinguish between the links.

bounding box was split, at the smallest edge, into two layers of four child boxes. This procedure was repeated at four levels of octrees, thereby generating 512 boxes per bounding box at the deepest level. Each box was modelled by one sphere positioned at the centre of the box, where its radius was equal to the diameter of the box (Fig. 6). This model yielded a safety zone of at most 4 mm for the end-effector. Link pairs 1–3, 5–9, 6–9, and 7–9 never collided and were therefore excluded from collision detection. Wires, stem segments and fruit were represented by a cylinder to resemble the shape of the object in reality. Therefore, a collision check with the manipulator or end-effector involved a sphere-cylinder check, which was more time consuming than the sphere–sphere checks used for self-collision detection (Kodam, Bharadwaj, Curtis, Hancock, & Wassgren, 2010).

2.3.2. *Step 1: calculating a start and goal configuration*

To calculate a goal configuration, an end-effector pose was selected with respect to the fruit and subsequently an inverse kinematics solution was calculated for this pose. Since end-effector poses were taken in the horizontal plane, only the azimuth angle of the end-effector φ_{EE} was varied (the pose is orthogonal to the aisle at $\varphi_{EE} = 0°$). The preferred angle of φ_{EE} is a pose where the end-effector is in line with the fruit and stem, as visualised in Fig. 7. However, due to presence of obstacles, the preferred pose was not always accessible. Therefore a φ_{EE} was selected that deviated from this preferred pose, at an angle φ_{Dev} (Eq. (1)). Preliminary field tests of the end-effector mechanism revealed that an increasing φ_{Dev} increases the probability of unsuccessful fruit detachment and of damaging the fruit or stem. Therefore, the pose with the smallest φ_{Dev} was selected, for path planning, from the poses resulting in a collision-free goal configuration.

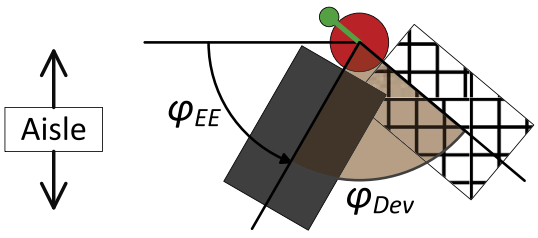

Fig. 7 – Azimuth angle φ_{EE} of the end-effector with respect to the fruit. Depending on φ_{EE}, the selected end-effector pose (■) may deviate from the preferred pose (□) by φ_{Dev}.

$$\varphi_{Dev} = |\varphi_{Fruit} - \varphi_{EE}| \quad (1)$$

Inverse kinematics were calculated using the Robotics, Vision & Control Toolbox (Corke, 2011). The inverse kinematics solution was furthermore checked for collisions. The goal configuration was calculated for each fruit, whereas a fixed start configuration (Fig. 4) was taken that was collision-free for all fruit.

2.3.3. *Step 2: path planning*

Most path planning algorithms were not suitable for the nine DOF manipulator because planning time increases exponentially with the number of DOFs (Choset et al., 2005). However, the sampling-based planner implemented, a balanced bi-directional rapidly exploring random tree (bi-RRT) (Kuffner & Lavalle, 2000; Lavalle, 2006), is less affected by the number of DOFs. This planner randomly samples the configuration space using a user-defined sampling resolution ε. If the sample is collision-free, it is added to a growing tree. After a number of iterations, the trees growing from the start and goal configuration may connect and a path is found. In preliminary analyses, paths were mostly found within 200 iterations and, in an exceptional case, 800 iterations were needed. The maximum number of iterations was therefore set to 1500, which corresponds to a planning time of about half an hour. Such a long time is acceptable for objectives of this research, but the cycle time of the real robot should not exceed 6 s to be economically feasible (Pekkeriet, 2011). Although reducing planning time is outside the scope of this research, a possible direction would be to implement the algorithm in C++ code and to use a rapid collision detection algorithm (Sucan & Chitta, 2014).

2.3.4. *Step 3: path smoothing*

Paths found by RRT-based planning algorithms tend to be tortuous due to the random nature of such sampling algorithms. Therefore, a path-smoothing algorithm was implemented following the path-planning phase. The implemented algorithm was an adaptation of a classical heuristic based on the divide and conquer concept, which consists of decomposing the original path and omitting some of its nodes, iteratively, in order to shorten it (Carpin & Pillonetto, 2005). Additionally a "memory matrix" was implemented where calculated sub-path options were retained for subsequent use. These modifications facilitated achieving shorter processing times. To ensure the smoothened path was indeed collision-free, consecutive vertices were linearly interpolated using ten configurations. A path was determined as collision-free if all interpolated configurations were collision-free.

3. Performance measures

Performance was measured in terms of success rate, planning time, and path quality. Success rates comprised of the following:

- Suc_{Goal} – Goal configuration success (%) was calculated by:

$$Suc_{Goal} = \frac{100 \cdot [no.\ of\ fruits\ accessed]}{[total\ no.\ of\ fruits\ on\ plants]} \quad (2)$$

This measure expresses the percentage of fruits for which a collision-free goal configuration was found.

- Suc_{Path} – Path planning success (%) was calculated by:

$$Suc_{Path} = \frac{100 \cdot [no.\ of\ fruits\ for\ which\ a\ collision-free\ path\ is\ planned]}{[no.\ of\ fruits\ accessed]} \quad (3)$$

A path was considered collision-free if it was collision-free before and after path smoothing.

- Suc_{Mot} – Motion planning success (%) was calculated by:

$$Suc_{Mot} = \frac{Suc_{Goal} \cdot Suc_{Path}}{100} \quad (4)$$

Planning time refers to the time elapsed to plan a path. Path quality was quantified by the joint angles index of curvature (*JAIC*) (Eq. (5)). *JAIC*, measured in the configuration space, is based on the city-block distance metric capturing the overall sum of the joint angle paths, i.e. *JAIC* measures how much the joints rotate. *JAIC* was normalised by the sum of joint angle paths for the "line-of-sight" path (the shortest path from start point to goal point), thus *JAIC* ranges from 1 (the shortest path) to zero. The normalisation facilitated path quality comparisons among groups of fruit.

$$JAIC = \frac{\sum_{j=1}^{J} |x_{1,j} - x_{V,j}|}{\sum_{i=2}^{V} \sum_{j=1}^{J} |x_{i-1,j} - x_{i,j}|} \quad (5)$$

where: $x_{i,j}$ (°) is the ith vertex's position in the jth dimension of the configuration space (the jth joint's angle value); V the total number of vertices in the path; and J the total number of joints, which was nine in the current implementation.

4. Parameters

The influence of five parameters, on success or planning time, was investigated in this research (Table 2), namely: average stem spacing (S_{Stem}), fruit location at a side of the stem (S_{Fruit}), dimensions of the end-effector (D_{EE}), robot position (P_{Robot}) and sampling resolution (ε). Few values were selected per parameter to simplify analyses in Section 5.

Average stem spacing (S_{Stem}) was expected to influence Suc_{Goal} and Suc_{Path} because it modifies the density of the obstacle map. The parameter value depends on the cultivation system used for sweet-pepper production. Plants were planted precisely 0.4 m apart and pruned such that two stems developed, in an approximately upward direction. Stem spacing therefore strongly varied from 0.1 to 0.3 m, among harvest cases, whereas average stem spacing was precisely known (0.2 m). We tested an increase of 50%, to 0.3 m, to realise a strong change in density of the obstacle map.

Fruit location at a side of the stem (S_{Fruit}) was expected to influence Suc_{Goal} and Suc_{Path} because, for instance, back side fruit are further away from the robot and obstacles are more likely to obstruct a path, than for front-side fruit. Fruit were grouped in four sides based on the measured azimuth angle at the fruit φ_{Fruit} (Fig. 3): front ($300° < \varphi_{Fruit} \leq 60°$), right ($60° < \varphi_{Fruit} \leq 120°$), back ($120° < \varphi_{Fruit} \leq 240°$), or left side ($240° < \varphi_{Fruit} \leq 300°$). These ranges were chosen to produce equal ranges of 120° for front, back and the sides.

Changing the dimensions of the end-effector (D_{EE}) was expected to influence goal configuration success (Suc_{Goal}) and path planning success (Suc_{Path}) because D_{EE} influences the size of the collision-free workspace of the manipulator. Dimensions were precisely known and did not vary among harvest cases. A 25% decrease of the bounding box of the end-effector was investigated. Given the empty spaces observed, between end-effector parts, it was considered that such a decrease was feasible after another design iteration.

Robot position (P_{Robot}) influences dexterity of the manipulator (Klein & Blaho, 1987), i.e. the range of motion toward a fruit. In addition, variation in position and orientation of the fruit influence the robot position required to reach a fruit. Therefore, P_{Robot} was expected to influence Suc_{Goal} and Suc_{Path}. For each fruit, eight robot positions were tested. The first position was always taken at an offset of 1.5 m between robot and target fruit. Subsequently, steps of 0.1 m were made until the last position (Fig. 8). Investigating this range of robot positions assured that fruit were within the workspace, whereas for a wider range of robot positions, fruit were outside the workspace. To choose a robot position for path planning, the

Table 2 – Parameters analysed in this research.

Parameter	Description	Values tested
S_{Stem}	Change (%) in average stem spacing (m)	[0 (narrow), 50 (wide)]
S_{Fruit}	Fruit location at a side of the stem	[Front, left, right, back]
D_{EE}	Change (%) in dimensions (width, length, height) of the end-effector	[−25 (small), 0 (big)]
P_{Robot}	Position of robot (m) along the rail, relative to the fruit	[0, 0.1, …, 0.7]
ε	Sampling resolution of RRT algorithm (–)	[0.03, 0.05, 0.07]

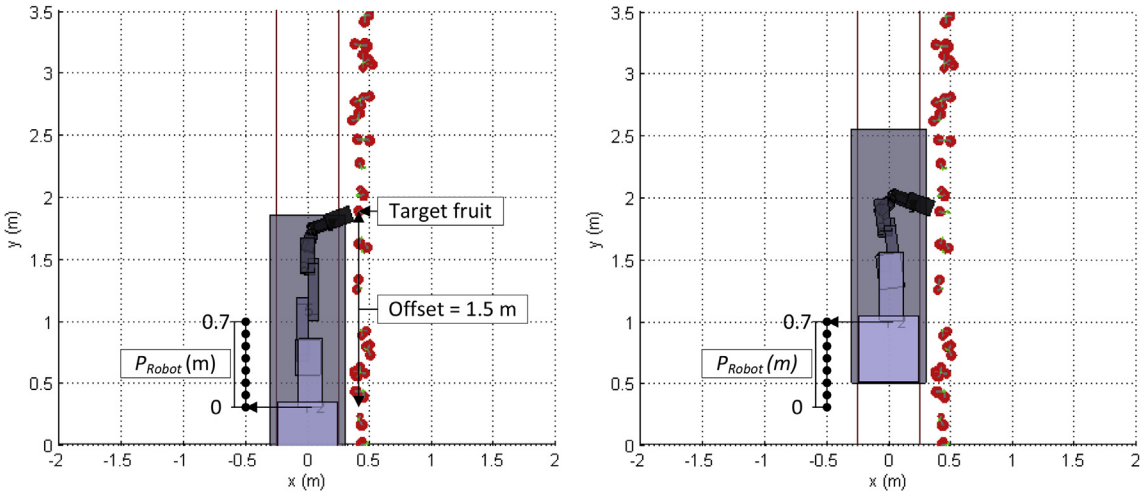

Fig. 8 – Visualisation of robot position (P_{Robot}), with values ranging from of 0 m (left) to 0.7 m (right) in steps of 0.1 m. Initial robot position ($P_{Robot} = 0$) was put at a fixed offset of 1.5 m, between the robot and target fruit.

position with most collision-free azimuth angles of the end-effector (φ_{EE}) was selected. If several robot positions resulted in the same number of angles, the one closest to the middle position ($P_{Robot} = 0.3$ or 0.4) was selected because dexterity was greatest at these positions.

Sampling resolution (ε) is a parameter in the bi-RRT algorithm that determines the probability of finding a path. A smaller resolution increases this probability at the cost of additional iterations. A small resolution is typically required for fruit surrounded by many obstacles. At each fruit, the resolution was initially set to 0.07. If a path was not found, or if the path contained a collision after path smoothing, a resolution of 0.05 and 0.03 were subsequently attempted. Smaller resolutions were not tested because the probability of existence of a solution would become very low and the maximum number of iterations would easily be exceeded.

5. Analyses

Two analyses were performed using simulation. In the first analysis, all three steps of the algorithm sequence were performed and goal configuration success, path planning success and motion planning success were compared between methods of selecting the azimuth angle of the end-effector: the constrained-azimuth method and the full-azimuth method (Section 5.1). The constrained-azimuth method was the best and was therefore used for the second analysis that also involved a sensitivity analysis. The effect of stem spacing, fruit location and end-effector dimensions were tested on goal configuration success (Section 5.2). Subsequently, the goal configurations obtained were used to plan a path, and smooth the path, where the sensitivity of stem spacing, end-effector dimensions and the sampling resolution (ε) were tested on path planning success (Section 5.3).

5.1. *Constrained-azimuth method vs. full-azimuth method*

This analysis focused on properly selecting the azimuth angle of the end-effector with respect to the target fruit. Preferably, azimuth angle should be selected such that the end-effector is positioned in front of the fruit, with the stem located behind the fruit (Fig. 9), because preliminary field tests of the end-effector mechanism revealed that such a pose increased the probability of successful fruit detachment while reducing the probability of damaging the fruit or stem (Bac et al., 2015). However, accessing this preferred pose is sometimes problematic for fruit located on the left, right, or back side of the stem, for two reasons. Firstly, in the few cases a path exists to the preferred pose, there is a high probability of damaging a plant part while executing the path. Secondly, planning a path to the preferred pose may result in a long planning time. To avoid these issues and to quickly find a successful path, the azimuth angle was selected to approach the fruit from a constrained range of 120° (taken 60° to the left and 60° to the right of the axis perpendicular to the crop row), which was defined as the 'constrained-azimuth' method. The constrained-azimuth method may offer advantages in terms of planning time and avoiding a motion that risks collisions. A possible disadvantage is that some fruit may not be reached. To validate these expectations, this method was compared with selection of the azimuth angle from the full range of 360°, which is defined as the 'full-azimuth' method, and determined the level of motion planning success.

Goal configurations were evaluated in the simulation and subsequently path planning was performed. For the constrained-azimuth method, 13 azimuth angles φ_{EE} (−60, −50, …, 60°) were simulated, whereas all 36 azimuth angles φ_{EE} (0, 10, …, 350°) were simulated for the full-azimuth method. The configuration with the smallest φ_{Dev} was selected as goal configuration for path planning.

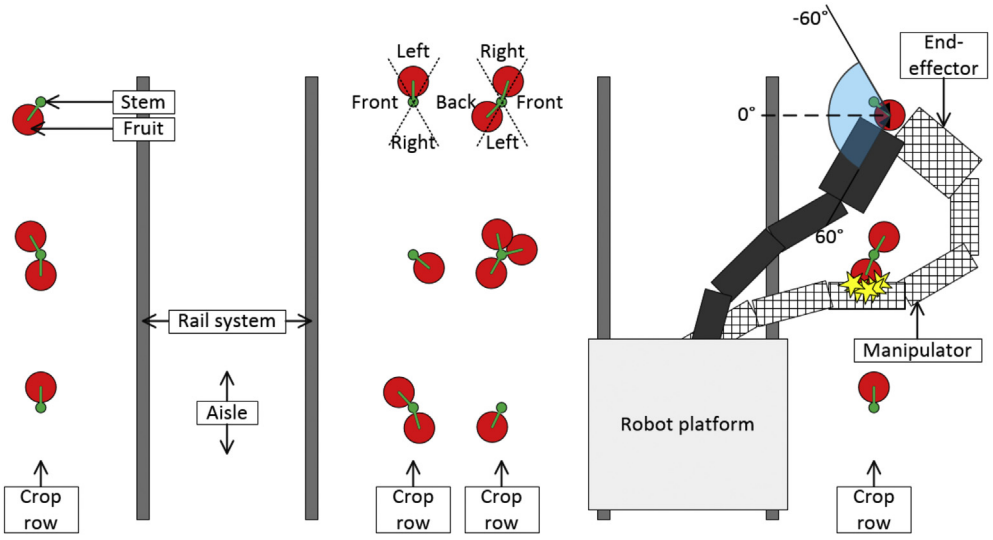

Fig. 9 – Top view of four crop rows, along two aisles, in the greenhouse. Two end-effector poses at a back side fruit are shown. In the preferred pose (□), the end-effector is positioned in front of a fruit and the stem behind the fruit. Yet, the manipulator collides (indicated by stars) with a nearby fruit. The alternative collision-free pose (■) is selected from a constrained range (120°) of the azimuth angle, taken 60° to the left and 60° to the right of the axis perpendicular to the crop row, instead of from the full range of 360°.

The performance measures (Section 3) were evaluated for both methods. In addition, fruit were counted that were either reachable by only the full-azimuth method, or reachable by both methods, but where the full-azimuth method resulted in a smaller φ_{Dev}.

5.2. *Sensitivity analysis of finding a goal configuration*

This sensitivity analysis tested effects on finding a successful goal configuration (Suc_{Goal}) for a fruit, regardless whether a path existed toward that goal configuration. In Section 5.2.1, the effect of stem spacing, fruit location and dimensions of the end-effector on Suc_{Goal} are assessed. Section 5.2.2 describes the effect of distinct robot positions on Suc_{Goal}.

5.2.1. *Stem spacing, fruit location and dimensions of the end-effector*

Assessing S_{Stem} and D_{EE} each involved two levels and therefore four combinations of these two parameters were tested for goal configuration success (Suc_{Goal}). Subsequently, results were grouped by S_{Fruit}: all sides (N = 158), the front (N = 46), left (N = 31) and right (N = 32) combined, and the back side of the stem (N = 48).

To determine if S_{Stem}, S_{Fruit}, and D_{EE} had a statistically significant effect on Suc_{Goal}, a repeated measures logistic regression was performed (Field, 2009). To focus the regression on these effects, the effect of robot position was disregarded and was separately investigated in Section 5.2.2. Hence, a fruit was successfully accessed, if at least one of the eight robot positions resulted in a successful goal configuration. Logistic regression is a multiple regression with a categorical output variable and predictor variables that are continuous or categorical. The values of the predictor variables determine probability of membership to a category: success or failure. Predictor variables were pair-wise compared, to validate the following hypotheses:

- Suc_{Goal} is significantly greater for fruit at the front side than any of the other three sides, i.e. three pair-wise comparisons;
- Dimensions of the end-effector (D_{EE}) have a significant effect on Suc_{Goal}, i.e. a single pair-wise comparison;
- Stem spacing (S_{Stem}) has a significant effect on Suc_{Goal}, i.e. a single pair-wise comparison;
- There are significant parameter interaction effects between D_{EE}, S_{Stem} and side of fruit location.

The output was presented in a table, with an odds ratio and a probability value to determine significance.

The side predictor involved multiple pair-wise comparisons on a single set of data. To reduce the chances of obtaining false-positive results in such a statistical model, a post-hoc analysis adjusted using Bonferroni correction was performed (Field, 2009). Furthermore, the Quasi likelihood Independence Criterion (QIC) was used to test the contribution of each predictor to the statistical model (Field, 2009).

5.2.2. *Distinct robot positions*

In previous analyses all robot positions were tested, whereas in this analysis the effect of distinct robot positions (P_{Robot}) on goal configuration success (Suc_{Goal}) was tested, for combinations of end-effector dimensions and stem spacing: S_{Stem} = narrow & D_{EE} = big, S_{Stem} = narrow & D_{EE} = small, S_{Stem} = wide & D_{EE} = big and S_{Stem} = wide & D_{EE} = small. This analysis considered if goal configuration success was sensitive to distinct robot positions.

The requirement to optimise the robot position in order to access the fruit was investigated by comparing Suc_{Goal} between a fixed robot position (P_{Robot} = 0.3) with any of the eight

positions. This test was conducted for one parameter combination: S_{Stem} = narrow and D_{EE} = big.

To evaluate if robot position influences the expected success of detachment, φ_{Dev} was measured at distinct robot positions. At each position, a mean φ_{Dev} was taken over the fruit belonging to one of the twelve possible parameter combinations of S_{Stem}, D_{EE} and S_{Fruit}.

5.3. Sensitivity analysis of path planning and path smoothing

Goal configurations obtained in Section 5.2 were used in this analysis, to determine if a collision-free and smooth path can be planned from the start configuration to a goal configuration. The influence of D_{EE} and S_{Stem} (same combinations as in Section 5.2.2) on path planning success Suc_{Path} and path quality measure *JAIC* was tested. Furthermore, only one planning attempt was performed for each fruit due the probabilistic completeness property of the bi-RRT planner. Probabilistic completeness means that, if a solution exists, the probability that the planner solves the problem becomes 1 as the planning time extends to infinity (Berenson & Srinivasa, 2010). Probabilistic completeness is closely approximated, but not fully met, because a maximum number of iterations (1500) was set. This maximum was about double the maximum number of iterations observed to find a path (800). Hence, it was considered that the probability of finding a solution beyond 1500 iterations was extremely low.

6. Results

6.1. Constrained-azimuth method vs. full-azimuth method

Goal configuration success (Suc_{Goal}) of the full-azimuth method (66%) was slightly greater than for the constrained-azimuth method (63%) (Table 3). A goal configuration was found by only the full-azimuth method for five fruit, but for only one of these fruit a path was found. A goal configuration was found by both methods, where the full-azimuth method resulted in a smaller φ_{Dev}, for eight fruit. Yet, for only one of these fruit a path was found. As a result, motion planning success (Suc_{Mot}) of the constrained-azimuth method (63%) was similar to the full-azimuth method (64%).

6.2. Sensitivity analysis of finding a goal configuration

6.2.1. Stem spacing and dimensions of the end-effector

Figure 10 shows the effect of stem spacing S_{Stem} and D_{EE} on goal configuration success (Suc_{Goal}), grouped by fruit location at the stem. Fruit located at the front side (Suc_{Goal} = 93%) of the stem were better accessible than left + right side (Suc_{Goal} = 59%) and back side (Suc_{Goal} = 41%). In addition, smaller end-effector dimensions (D_{EE}) and wide stem spacing (S_{Stem}) both improved goal configuration success (Suc_{Goal}).

In the repeated measures logistic regression of Suc_{Goal}, interactions were found to have no significant influence on Suc_{Goal}, and therefore were removed from the model. The main effects were all found significant and results are presented in Table 4. All three predictors have a significant effect ($p \leq 0.01$) on goal configuration success. The odds of accessing a fruit, using smaller end-effecter dimensions (−25%), are about twice (1/0.52) as high than using the big end-effector dimensions. In addition, the odds of accessing a fruit, when stems are additionally spaced apart (+50%), is about twice as high (1/0.52) than with narrow stem spacing. However, the strongest influencing predictor was the side where the fruit was located. The odds of accessing fruit at the front side were significantly higher than any other side (right = 1/0.099, left = 1/0.112, back = 1/0.046).

The Bonferroni corrected post-hoc analysis of side revealed that only front was significantly different from all other sides (right and left: $p < 0.01$, back: $p < 0.001$). Although the influence of the side predictor S_{Fruit} was much higher than that of D_{EE} or S_{Stem}, including D_{EE} and S_{Stem} predictors improved the model according the Quasi likelihood Independence model Criterion (QIC) (610 for all three predictors, 629 for just S_{Fruit}). Hence, this finding indicates that all three predictors should be considered in the statistical model.

6.2.2. Distinct robot positions

Figure 11 displays the effect of distinct robot positions on goal configuration success (Suc_{Goal}). The difference between minimum and maximum goal configuration success was greater for S_{Stem} = narrow & D_{EE} = big (14%) than for S_{Stem} = wide & D_{EE} = small (7%). This finding indicates that robot position became more critical for decreasing stem spacing and increasing end-effector dimensions. Moreover, for the best position (P_{Robot} = 0.3 m), goal configuration success was 61% (Fig. 11). When testing all positions, Suc_{Goal} further increased to 63% (not shown in Fig. 11). The three fruit (2%) that were not accessible from P_{Robot} = 0.3 m, were accessible from P_{Robot} = 0.4 or 0.5 m. Therefore, one can conclude that fruit were always accessible from robot positions where the manipulator had a high dexterity (P_{Robot} = 0.2–0.5 m).

The effect of distinct robot positions on φ_{Dev} is presented in Fig. 12. Parameter combinations of S_{Stem} and D_{EE} did not strongly influence φ_{Dev}, whereas fruit location had a strong influence.

Fruit locations at the front-side of the stem could be accessed with a small mean φ_{Dev} of <30° that is desirable for successful detachment. At the left and right sides, the deviation angle φ_{Dev} deteriorated with values ranging between 55° and 75°. For back side fruit, mean φ_{Dev} drastically deteriorated with values greater than 110°. Moreover, for front, left and right side fruit, mean φ_{Dev} was smaller for robot positions ranging between 0.2 and 0.5 m, than for "extreme" robot positions. This finding supports earlier results as shown in Fig. 11.

Table 3 – Comparison of success rates between the constrained-azimuth method and the full-azimuth method.

	Constrained-azimuth	Full-azimuth
Suc_{Goal} (%)	63	66
Suc_{Path} (%)	100	96
Suc_{Mot} (%)	63	64

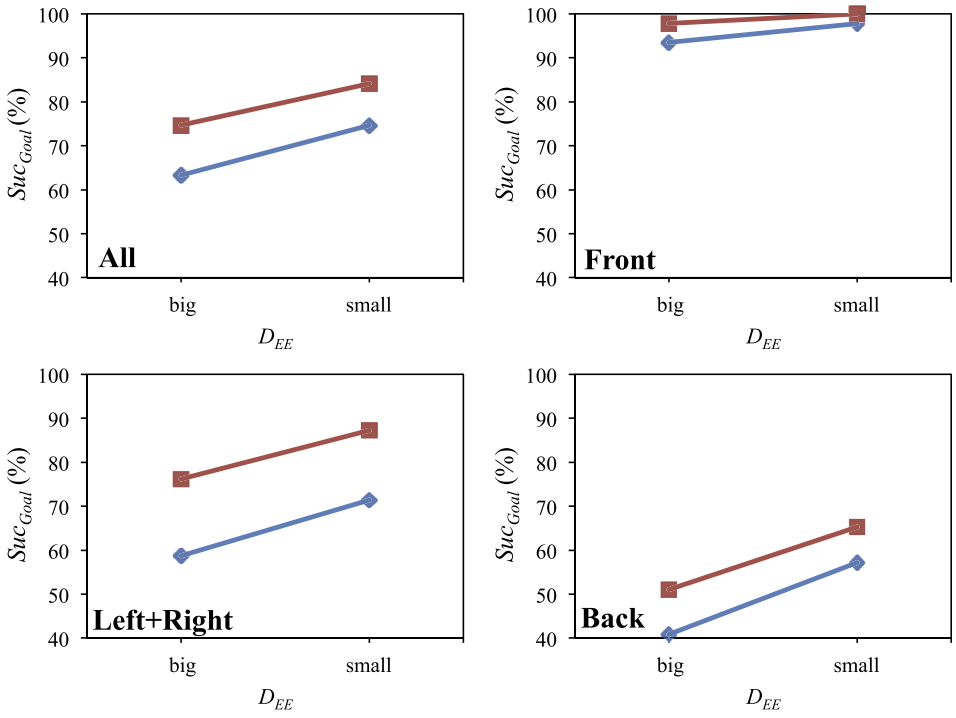

Fig. 10 – Effect of end-effector dimensions (D_{EE}) and stem spacing (◆ S_{Stem} = narrow; ■ S_{Stem} = wide) on goal configuration success (Suc_{Goal}). Graphs are grouped by fruit location: all 158 fruit (top-left), 46 front-side fruit (top-right), 63 left + right-side fruit (bottom-left), and 48 back-side fruit (bottom-right).

Table 4 – Results of repeated measures logistic regression for goal configuration success. Predictors are fruit location (three pair-wise comparisons of the side), stem spacing (S_{Stem}), and dimensions of the end-effector (D_{EE}).

	Beta (std. error)	95% Confidence interval for odds ratio		
		Lower	Odds ratio	Upper
Constant	3.91 (0.63)**	14.62	49.82	169.77
S_{Fruit} = right vs. front	−2.31 (0.71)*	0.025	0.099	0.394
S_{Fruit} = left vs. front	−2.19 (0.71)*	0.028	0.112	0.450
S_{Fruit} = back vs. front	−3.09 (0.67)**	0.012	0.046	0.169
S_{Stem} = narrow vs. wide	−0.65 (0.11)**	0.416	0.52	0.650
D_{EE} = big vs. small	−0.65 (0.11)**	0.420	0.52	0.645

Note: **p < 0.001, *p < 0.01.

6.3. *Sensitivity analysis of path planning and path smoothing*

Results of path planning and smoothing revealed no parameter influences on Suc_{Path} (Table 5) and path planning was generally successful (>97%). No notable difference in path quality was observed, in all four combinations.

Sampling resolution (ε) required was 0.07 for at least 85% of the paths. For the remaining paths a finer resolution was required. Planning time, for combined parameter combinations, was on average (range) 59 (3–1494) s for $\varepsilon = 0.07$ with N = 423, 137 (23–607) s for $\varepsilon = 0.05$ with N = 27, and 401 (42–901) s for $\varepsilon = 0.03$ with N = 8. Although planning time was longer at a smaller sampling resolution, one should consider that these planning problems were typically harder to solve. Furthermore, the planning time varies significantly.

7. Discussion

The new constrained-azimuth method resulted in successfully planned motions for 100 fruit (63%), and for one additional fruit in case of the full-azimuth method. The constrained-azimuth method avoids challenging planning problems by selecting a different goal configuration. Furthermore, the constrained-azimuth method may result in less damages to the crop because, during robotic harvesting tests trails (Bac et al., 2015), it was noticed that the manipulator sometimes collided with unseen plant parts when it approached a fruit from the side or back side of the stem. Other crops with a similar plant architecture may benefit from this method as well and validating this hypothesis is therefore a task for future work. Lastly, the value of 120°, to constrain

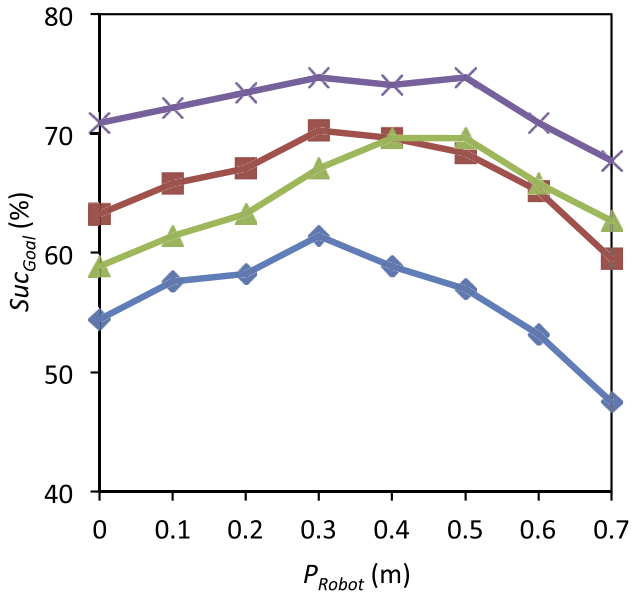

Fig. 11 – Effect of robot position (P_{Robot}) on goal configuration success (Suc_{Goal}), grouped by four combinations of S_{Stem} and D_{EE}: S_{Stem} = *narrow* and D_{EE} = *big* (blue, diamond), S_{Stem} = *narrow* and D_{EE} = *small* (red, square), S_{Stem} = *wide* and D_{EE} = *big* (green, triangle), S_{Stem} = *wide* and D_{EE} = *small* (purple, cross). All 158 fruit were evaluated to determine each value. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the azimuth angle, was empirically determined and not optimised in this study. Optimising this value is therefore a task for future work.

Fruit location strongly determined goal configuration success. Especially back side fruit were hard to reach and, if reached, the end-effector pose resulted in a large deviation angle ($\varphi_{Dev} > 110°$) that will probably lead to unsuccessful fruit detachment. How strongly detachment success will decrease, at such an increasing deviation angle, is currently unknown and should be determined in future field tests. In related work, fruit location also played a dominant role. For strawberry harvesting, harvest success was only 8% for fruit located behind other fruit, yet 72% for fully accessible fruit (Hayashi et al., 2010). To address the issue of back side fruit or complex harvest cases, three directions can be pursued. Firstly, the end-effector design can be adapted to widen the allowable range of φ_{Dev}. Secondly, adaptations in plant genetics, the cultivation system, or cultivation practices can be implemented to generate more fruit at the front side of the plant stem. Adaptations made in other crops were reviewed in previous work (Bac, Van Henten, et al., 2014). Thirdly, human–robot co-working can be applied, where the robot only harvests easy harvest cases, whereas humans harvest the remaining more difficult cases (Bac, Van Henten, et al., 2014). The economic feasibility of this scenario is yet to be determined.

Although the end-effector interacts more with the plant than the manipulator, the dimensions of manipulator links and the kinematic structure may also influence motion planning success. Different lengths and dimensions of the

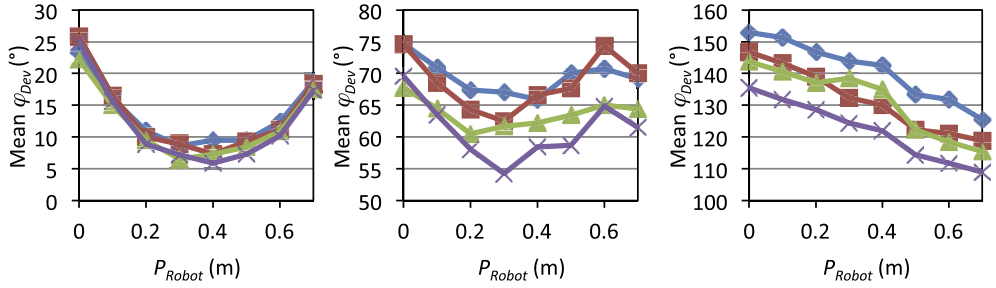

Fig. 12 – Effect of robot position (P_{Robot}) on mean φ_{Dev}, for fruit locations at the front-side (left), left + right-side (centre) and back side (right) of the stem. Lines in each graph indicate parameter combinations of S_{Stem} and D_{EE}: S_{Stem} = *narrow* and D_{EE} = *big* (blue, diamond), S_{Stem} = *narrow* and D_{EE} = *small* (red, square), S_{Stem} = *wide* and D_{EE} = *big* (green, triangle), S_{Stem} = *wide* and D_{EE} = *small* (purple, cross). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 5 – Results of path planning and path smoothing.

Combination	Path planning				Path quality	
	Suc_{Path} in (%)	Number of fruit evaluated (#)	Proportion of paths found per ε: 0.07, 0.05, 0.03	Mean planning time in s (SD)	Mean JAIC before smoothing	Mean JAIC after smoothing
S_{Stem} = narrow & D_{EE} = big	100	100	90%, 7%, 3%	56 (96)	0.73	0.91
S_{Stem} = narrow & D_{EE} = small	99	118	85%, 12%, 3%	63 (136)	0.74	0.93
S_{Stem} = wide & D_{EE} = big	97	118	95%, 5%, 0%	86 (161)	0.74	0.91
S_{Stem} = wide & D_{EE} = small	98	133	95%, 4%, 1%	97 (227)	0.74	0.94

links may influence the reachable workspace of the manipulator and optimising the robot position may therefore become more relevant. Optimising the robot position increased goal configuration success by only 2% for the manipulator used in this research. Furthermore, the number of DOF influences the manoeuvrability around obstacles. In this application, planned motions were relatively simple and fewer DOFs may not have a strong influence on the result. However, in other applications, DOFs could play an important role. When reusing the methodology in future work, these manipulator aspects should be considered.

In the current research, it was assumed that fixed obstacles were not touched by the robot. In practice, however, workers commonly touch or move fruit or stems aside to better access fruit. Relaxing the hard obstacle constraint and implementing danger zones (Sent & Overmars, 2001) may improve motion planning success and is therefore suggested for future work.

Another assumption was that soft obstacles, i.e. leaves and petioles, could be pushed aside and were therefore disregarded in the analyses. If a soft obstacle was pushed aside, the fruit position might change, leading to unsuccessful detachment. Furthermore, leaves can get stuck in the end-effector, which may lead to contamination and improper functioning of the end-effector. Hence, the influence of soft obstacles on motion planning success requires further study in future work.

A collision-free path was not found for 37% of the sweet-pepper fruit, whereas for cucumbers the motion planner was responsible for 4.9% of the total number of failed attempts during a field test (Van Henten, Van Tuijl, et al., 2003). Although robot designs differed, one may conclude that sweet pepper is more challenging to harvest than cucumber, probably due to the dense obstacle spacing one finds in a sweet-pepper crop.

The success rates reported may differ from success rates of a future field test with the actual robot because additional sources of error may lower the motion planning success. These errors include hardware related errors, imprecise positioning of the end-effector, inaccurate localisation of objects, and path obstructions by objects not included in the obstacle map (Van Henten, Van Tuijl, et al., 2003). Such errors were not investigated in this research, but they are relevant upon implementation of the robot in a greenhouse. A comparison of success rates between this study and a field test is a task for future work. Nevertheless, this study provides more insight in which parameters cause the suboptimal motion planning success of 63%, similar to a previous analysis for robotic cucumber harvesting (Van Henten et al., 2010). The current paper may also provide tools to analyse sensitivity of complex robotic applications where multiple parameters influence successful automation of the task.

8. Conclusion

Motion planning success was 63% (N = 158) for the constrained-azimuth method and 64% for the full-azimuth method. Approaching fruit from only the aisle side of the crop is successful because this method avoids long planning times and avoids executing paths with a potential risk for colliding and damaging the plant, whereas motion planning success hardly decreases. The 'constrained-azimuth' method may serve as a useful generic method to approach fruit in crops with a similar plant architecture, such as tomato, cucumber, or eggplant.

The sensitivity analysis revealed that reducing end-effector dimensions and widening stem spacing significantly improve the goal configuration success, from 63% to 84%. Fruit located at the front of the stem are significantly more accessible than any other side. Yet, most importantly, the location of fruit proved to be the strongest influencing factor on goal configuration success. Hence, this new finding provides an incentive to train or breed plants in such a way that more fruit develop at the front side of the plant. Furthermore, goal configuration success was 61% for a fixed robot position and 63% for an optimised robot position. Robot position therefore hardly affected goal configuration success in this study. However, robot position influences deviation in azimuth angle between end-effector and fruit. The effect of this deviation on success of fruit detachment by the end-effector is needs to be tested in future field tests.

Path planning success was in the range of 98–100% for the four parameter combinations of end-effector dimensions and stem spacing. As such it can be concluded that motion planning success was mostly limited by the ability to find a collision-free goal configuration in this obstacle-dense crop environment, whereas path planning towards that goal configuration was not critical. Therefore, paths were almost flawlessly found, probably because challenging planning problems were avoided by applying the constrained-azimuth method to determine a goal configuration. Furthermore, more than 85% of the paths were found by a sampling resolution of 0.07.

Acknowledgements

This research was funded by the European Commission in the 7th Framework Programme (CROPS GA no. 246252) and by the Dutch horticultural product board (PT no. 14555). We are grateful to growers Cees and Rolf Vijverberg who allowed us to record measurements in their greenhouse.

REFERENCES

Bac, C. W., Hemming, J., & Van Henten, E. J. (2013). Robust pixel-based classification of obstacles for robotic harvesting of sweet-pepper. *Computers and Electronics in Agriculture, 96*, 148–162.

Bac, C. W., Hemming, J., & van Henten, E. J. (2014). Stem localization of sweet-pepper plants using the support wire as a visual cue. *Computers and Electronics in Agriculture, 105*, 111–120.

Bac, C. W., Hemming, J., Van Tuijl, B. A. J., Barth, R., Wais, E., & Van Henten, E. J. (2015). Performance evaluation of a harvesting robot for sweet-pepper (submitted for publication).

Bac, C. W., Van Henten, E. J., Hemming, J., & Edan, Y. (2014). Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. *Journal of Field Robotics, 31*(6), 888–911.
Berenson, D., & Srinivasa, S. S. (2010). Probabilistically complete planning with end-effector pose constraints. In *Proceedings – IEEE international conference on robotics and automation* (pp. 2724–2730).
Carpin, S., & Pillonetto, G. (2005). Motion planning using adaptive random walks. *IEEE Transactions on Robotics, 21*(1), 129–136.
Choset, H., Lynch, K. M., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L. E., et al. (2005). *Principles of robot motion*. Cambridge, MA: MIT Press.
Corke, P. I. (2011). *Robotics, vision & control*. Germany, Berlin: Springer.
Craig, J. J. (2004). *Introduction to robotics* (3rd ed.). Upper Saddle River, NJ: Pearson Education Inc.
CROPS. (2014). CROPS: clever robots for crops. http://www.crops-robots.eu/ Accessed 31.03.14.
Edan, Y., Rogozin, D., Flash, T., & Miles, G. E. (2000). Robotic melon harvesting. *IEEE Transactions on Robotics and Automation, 16*(6), 831–835.
Field, A. (2009). Logistic regression (Ch. 8). *Discovering statistics using SPSS* (3rd ed.). London: Sage Publication Ltd..
Guo, J., Zhao, D. A., Ji, W., & Xia, W. (2010). (Motion planning). Design and control of the open apple-picking-robot manipulator. In *Proceedings – 2010 3rd IEEE international conference on computer science and information technology, ICCSIT 2010* (Vol. 2, pp. 5–8).
Hannan, M. W., & Burks, T. F. (2004). Current developments in automated citrus harvesting. In *ASAE annual international meeting 2004* (pp. 4187–4196).
Hayashi, S., Shigematsu, K., Yamamoto, S., Kobayashi, K., Kohno, Y., Kamata, J., et al. (2010). Evaluation of a strawberry-harvesting robot in a field test. *Biosystems Engineering, 105*(2), 160–171.
Hemming, J., Bac, C. W., Van Tuijl, B., Barth, R., Bontsema, J., Pekkeriet, E., et al. (2014). *A robot for harvesting sweet-pepper in greenhouses*. Paper presented at the International Conference of Agricultural Engineering, 6–10 July 2014, Zürich, Switzerland. URL http://www.geyseco.es/geystiona/adjs/comunicaciones/304/C01140001.pdf.
Klein, C. A., & Blaho, B. E. (1987). Dexterity measures for the design and control of kinematically redundant manipulators. *International Journal of Robotics Research, 6*(2), 72–83.
Kodam, M., Bharadwaj, R., Curtis, J., Hancock, B., & Wassgren, C. (2010). Cylindrical object contact detection for use in discrete element method simulations. Part I – contact detection algorithms. *Chemical Engineering Science, 65*(22), 5852–5862.
Kondo, N., Nishitsuji, Y., Ling, P. P., & Ting, K. C. (1996). Visual feedback guided robotic cherry tomato harvesting. *Transactions of the ASAE, 39*(6), 2331–2338.
Kondo, N., Taniwaki, S., Tanihara, K., Yata, K., Monta, M., Kurita, M., et al. (2007). End-effector and manipulator control for tomato cluster harvesting robot.
Kuffner, J. J., & Lavalle, S. M. (2000). (Motion planning). RRT-connect: an efficient approach to single-query path planning. In *Proceedings – IEEE international conference on robotics and automation* (Vol. 2, pp. 995–1001).
Lavalle, S. M. (2006). *Planning algorithms*. Cambridge, UK: Cambridge University Press.
Lewis, A., Watts, P. L., & Nagpal, B. K. (1983). *Investment analysis for robotic applications*. Technical paper. Society of Manufacturing Engineers. MS.
Liang, X., & Wang, Y. (2010). Analysis on the workspace of a tomato harvesting manipulator with 7 DOF. *Applied Mechanics and Materials, 37–38*, 155–161.
de Luca, A., Lanari, L., & Oriolo, G. (1991). A sensitivity approach to optimal spline robot trajectories. *Automatica, 27*(3), 535–539.
Meagher, D. (1982). Geometric modeling using octree encoding. *Computer Graphics and Image Processing, 19*(2), 129–147.
Pekkeriet, E. J. (2011). *CROPS project deliverable 12.1: Economic viability for each application*. Wageningen, The Netherlands: Wageningen UR Greenhouse Horticulture.
Sakai, S., Iida, M., Osuka, K., & Umeda, M. (2008). Design and control of a heavy material handling manipulator for agricultural robots. *Autonomous Robots, 25*(3), 189–204.
Sent, D., & Overmars, M. H. (2001). (Motion planning). Motion planning in environments with dangerzones. In *Proceedings – IEEE international conference on robotics and automation* (Vol. 2, pp. 1488–1493).
Sivaraman, B., & Burks, T. F. (2006). Geometric performance indices for analysis and synthesis of manipulators for robotic harvesting. *Transactions of the ASABE, 49*(5), 1589–1597.
Sucan, I. A., & Chitta, S. (2014). MoveIt. http://moveit.ros.org Accessed 03.03.14.
Tannous, M., Caro, S., & Goldsztejn, A. (2014). Sensitivity analysis of parallel manipulators using an interval linearization method. *Mechanism and Machine Theory, 71*, 93–114.
Van Henten, E. J., Hemming, J., Van Tuijl, B. A. J., Kornet, J. G., & Bontsema, J. (2003). Collision-free motion planning for a cucumber picking robot. *Biosystems Engineering, 86*(2), 135–144.
Van Henten, E. J., Schenk, E. J., van Willigenburg, L. G., Meuleman, J., & Barreiro, P. (2010). Collision-free inverse kinematics of the redundant seven-link manipulator used in a cucumber picking robot. *Biosystems Engineering, 106*(2), 112–124.
Van Henten, E. J., Van Tuijl, B. A. J., Hemming, J., Kornet, J. G., Bontsema, J., & Van Os, E. A. (2003). Field test of an autonomous cucumber picking robot. *Biosystems Engineering, 86*(3), 305–313.
Van Henten, E. J., Van't Slot, D. A., Hol, C. W. J., & Van Willigenburg, L. G. (2009). Optimal manipulator design for a cucumber harvesting robot. *Computers and Electronics in Agriculture, 65*, 247–257.
Van Tuijl, B. A. J., & Wais, E. (2014). *Harvesting device*. Patent no.: P6051697NL. Netherlands: Dutch Patent Office.
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*(3), 223–237.
Zhang, X., Ba, P., & Mu, L. (2012). Position error sensitivity analysis for polishing robot. *Advanced Materials Research, 500*, 326–330.

Nomenclature

a_{i-1}: link length of link i (m)
α_{i-1}: link twist of link i (rad)
d_i: link offset of link i (m)
θ_i: joint angle at axis i (rad)
Suc_{Goal}: goal configuration success (%)
Suc_{Path}: path planning success (%)
Suc_{Mot}: motion planning success (%)
JAIC: joint angles index of curvature (–)
$x_{i,j}$: joint position at vertex i and joint j (rad or m)
D_{EE}: change in dimensions (width, length, height) of the end-effector (%)
S_{Stem}: additional average stem spacing (m)
S_{Fruit}: fruit location at a side of the stem
P_{Robot}: position of robot (m) along the rail, relative to the fruit
ε: sampling resolution of the RRT planner (–)
φ_{Fruit}: azimuth angle of the fruit with respect to the stem (rad)
φ_{EE}: azimuth angle of the end-effector with respect to the fruit (rad)
φ_{Dev}: deviation between the selected azimuth angle and the desired azimuth angle of the end-effector (rad)