Computers and Electronics in Agriculture 116 (2015) 88–100

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

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

Optimization methodology to fruit grove mapping in precision agriculture

Javier Gimenez *, Daniel Herrera, Santiago Tosetti, Ricardo Carelli

Instituto de Automática, Universidad Nacional de San Juan-CONICET, Argentina

ARTICLE INFO

Article history:
Received 30 March 2015
Received in revised form 27 May 2015
Accepted 14 June 2015

Keywords:
Mapping
Agricultural environments
Optimization
Data filtering

ABSTRACT

The mapping of partially structured agricultural environments is a valuable resource for precision agriculture. In this paper, a technique for the mapping of a fruit grove by a mobile robot is proposed, which uses only front laser information of the environment and the exact position of the grove corners. This method is based on solving an optimization problem with nonlinear constraints, which reduces errors inherent to the measurement process, ensuring an efficient and precise map construction. The resulting algorithm was tested in a real orchard environment. For this, it is also developed a data filtering method capable to comply efficiently the observation-feature matching. The maximum average error obtained by the methodology in simulations was about 13 cm, and in real experimentation was about 36 cm.

1. Introduction

Robotics, a discipline considered within the automation area, is currently focusing their applications in partially structured or time varying scenarios, generating a significant and growing impact on the productive and service sectors. These applications require a high degree of operational autonomy, which should be designed for a specific task and take into account the environment conditions. An application scope of current interest with great potential impact in robotics is precision agriculture (Srinivasan, 2006). Within this, special robots, which have been called as service units (Auat Cheein and Carelli, 2013), are endowed with high degree of skill, autonomy and intelligence, allowing its application in the particular agricultural environment with variable weather conditions in terrains with irregular characteristics (Auat Cheein and Carelli, 2013).

Regarding map generation, there are many works (Ouellette and Hirasawa, 2008; Xiaogang and Xuetao, 2009; Lee et al., 2008) for indoor environments which give satisfactory results. However, there are still few proposals concerning the mapping in agricultural environments. When the map is available in this context, navigation errors decrease (Zhang et al., 2014). Moreover, this allows the vehicle to return to specific locations and perform tasks such as spraying in a suitable and precise manner, thus saving valuable resources (Libby and Kantor, 2011). From a cost perspective, the map should be built without expensive surveying equipment, and preferably using the same sensors used for vehicle guidance (Zhang et al., 2014). In Jin and Tang (2009), for example, a mapping strategy based on maize plants detection using a stereo camera is presented. In Zhang et al. (2014), Libby and Kantor (2011) it is reported navigation and mapping techniques on a commercial apple orchard, for which odometry and 3D LiDAR information is heavily used. Additionally, specific landmarks on the extremal trees were included to increase the probability of finding the next row and successfully enter in this, which also establish a loop closure indicator that facilitates odometry error corrections. In both articles, Zhang et al. (2014), Libby and Kantor (2011), the authors use landmarks in order to detect the extremal trees. Besides, these methods are based on a probalistic mapping approach. With exception of the filtering process, the mapping algorithm proposed in the present article is GPS and landmarks independent. It is also a non explicit probabilistic approach (such as Kalman Filter based mappings).

In recent years, many works have been made to solve the problem of simultaneous localization and mapping (SLAM) (Rovira-Mas, 2009; Bryson and Sukkarieh, 2008; Auat Cheein et al., 2011), in which a recursive process is generated in order to simultaneously minimize errors in vehicle location and environment mapping (Chatila and Laumond, 1985; Ayache and Faugeras, 1989). A concise introduction to the SLAM algorithm is offered in Durrant-Whyte and Bailey (2006,). Although SLAM is based on specific and precise methods such as the Kalman filter (Thrun et al., 2005), it involves high computational costs producing

* Corresponding author.
E-mail addresses: jgimenez@inaut.unsj.edu.ar (J. Gimenez), dherrera@inaut.unsj.edu.ar (D. Herrera), stosetti@inaut.unsj.edu.ar (S. Tosetti), rcarelli@inaut.unsj.edu.ar (R. Carelli).

http://dx.doi.org/10.1016/j.compag.2015.06.013
0168-1699/

List of symbols and acronyms

Symbols

$A_i^{\theta,\rho}$	homogeneous transformation matrix with a translation ρ_i and a rotation θ_i
$A_i^{-,\theta,\rho}$	matrix that results from eliminating the last row of $A_i^{\theta,\rho}$ when $i > 1$
$A_{k:m}^{\theta,\rho}$	product of the matrices $A_i^{\theta,\rho}$ with i from k to m
$a_n^{\gamma,\ell}$	$:= [-c_n^{\gamma}\widehat{\ell}_n, -s_n^{\gamma}\widehat{\ell}_n, 1]^T$
$a_1^{\varsigma,\delta,\gamma}$	$:= \widehat{\delta}_1[c_1^{\varsigma}, s_1^{\varsigma}]^T$
$a_n^{\varsigma,\delta,\gamma}$	$:= [-c_n^{\varsigma,\gamma}\widehat{\delta}_n, -s_n^{\varsigma,\gamma}\widehat{\delta}_n, 1]^T$ if $n > 1$
c_i^{θ}	$:= \cos(\widehat{\theta}_i)$
$c_n^{\varsigma,\gamma}$	$:= \cos(\varsigma_n + \gamma_{n-1})$
D_d	distance between the robot and the nearest right row of trees
D_i	distance between the robot and the nearest left row of trees
$d_1 = [d_{1,i}]$	vector with the average measurements of the left sides of the rectangles
$d_2 = [d_{2,i}]$	vector with the average measurements of the right sides of the rectangles
f'_{θ_i}	partial derivative of f with respect to $\Delta\theta_i$
f_n	function used to design the nth framing constraint
$\mathcal{F}$	$:= \{f, g, h, \phi\}$
g	function used to design the alignment constraint
h_n	function used to design the nth basic constraint
I_θ	set of all appropriate subscripts of θ
$\ell = [\ell_i]$	vector with the average measurements of the bases of the rectangles
M	maximum laser distance considered in the filtering process
$P_{i,1}$	exact position of the trees in the lower left corner of the plantation
$P_{i,2}$	exact position of the trees in the lower right corner of the plantation
$P_{f,1}$	exact position of the trees in the upper left corner of the plantation
$P_{f,2}$	exact position of the trees in the upper right corner of the plantation
p_n	exact position of the trees in the nearest left row
$\widehat{p}_n$	estimated position of the trees in the nearest left row
q_n	exact position of the trees in the nearest right row
$\widehat{q}_n$	estimated position of the trees in the nearest right row
R_i^{θ}	matrix that produces a rotation of θ_i grades
$R_{k:m}^{\theta,\rho}$	product of the matrices $R_i^{\theta,\rho}$ with i from k to m
s_i^{θ}	$:= \sin(\widehat{\theta}_i)$
$s_n^{\varsigma,\gamma}$	$:= \sin(\varsigma_n + \gamma_{n-1})$
v	control action of the linear velocity of the unicycle robot
$V[\theta_i]$	variance in the measurements corresponding to θ_i
w	control action of the angular velocity of the unicycle robot
$\tilde{x}$	position error of the vehicle in the corridor
X_i	points resulting from projecting the observations on a line with the estimated orientation of the tree rows
$\alpha = [\alpha_i]$	vector with average measures of the angles that define the locations of the nearest left row of trees
$\beta = [\beta_i]$	vector with average measures of the angles that define the locations of the nearest right row of trees
$\gamma = [\gamma_i]$	vector with average measures of the upper left angles of the rectangles
Γ	subset of $\Theta \cup \Lambda$ of independent variables, as small as possible, that allows writing the other variables as a function of them
$\delta = [\delta_i]$	vector with the average measurements of the diagonals of the rectangles
$\Delta^2(\Theta)$	objective function to be minimized for optimal corrections
$\Delta\theta_i$	correction to be made about θ_i in search of a compatible configuration
$\theta = [\theta_i]$	generic vector with some average measures of the rectangles
$\widehat{\theta}_i$	final estimation of θ_i
Θ	set of all vectors of averages measurements obtained
$\vartheta = [\vartheta_i]$	vector with average measures of the upper right angles of the rectangles
Λ	$:= \{\lambda_f, \lambda_g, \lambda_h, \lambda_\phi\}$ set of vectors containing the Lagrange multipliers arising in the optimization problem
μ	mode parameter of the density used in the filtering process
σ	distance between modes of the density used in the filtering process
$\varsigma = [\varsigma_i]$	vector with average measurements of the angles between the base and diagonal of the rectangles
Φ	Lagrangian function resulting in the optimization problem
ϕ_n	function used to design the nth rectification constraint
φ	orientation error of the vehicle in the corridor
φ_d	difference between the orientations of the robot and the nearest right row of trees
φ_i	difference between the orientations of the robot and the nearest left row of trees

Acronyms

LiDAR	Light Detection and Ranging
GPS	Global Positioning System
SLAM	Simultaneous Localization and Mapping
INTA	National Agricultural Technology Institute
ML	Maximum Likelihood

slow motions of the robot while mapping online. The matching problem between the observations and the map elements represents a weakness of the SLAM method (Adams et al., 2014). This drawback is drastically reduced if a draft map is previously obtained by exploiting the semi-structured nature of the working environment.

In this paper, a mapping algorithm for a grove of fruit trees is proposed. Thereby an autonomous robot may estimate the trees position faster without using self-localization devices as GPS, odometry or inertial measurement units, and without using-localization devices as GPS (which can fail due to occlusion by the orchard), among others, as is detailed in Section 2. The control system used for autonomous navigation of the robot is presented in Section 3. During navigation, the robot performs angular and linear measurements of the fruit grove as described in Section 4. Measurements include distance errors of the laser and angular errors caused by the discretization of the directions in which it measures. These errors lead to inconsistencies between the estimations, such as: sum of interior angles of the quadrilaterals other than 360° or improperly closed polygons. In Section 5, a measure adjustment procedure is developed in order to achieve the best consistent setting of the overdetermined measuring set. In Section 6, some simulations and an experience in a real olive grove show the proposed method benefits. Conclusions and future

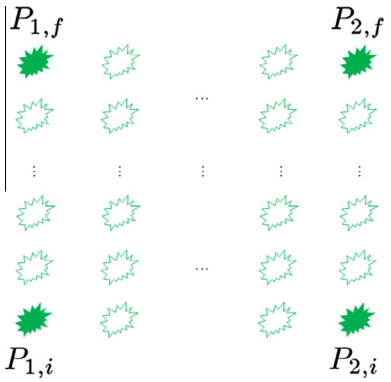


Fig. 1. Corner trees in the grove.

goals are given in Section 8. Finally, in the Appendix (see Section A.2), it is shown the ad hoc filtering procedure of the laser observations used in the experiments with real data.

2. Setting of the problem

Consider a robot navigating between rows of a plantation of vines or fruits, such as an olive grove, with the aim of mapping the orchard. Suppose that the grove has multiple parallel corridors and that it is regular, i.e., it is composed of rows with the same amount of trees located such that each path can be divided into quadrilaterals, as shown in Figs. 1 and 3. If an expected tree is lacking in some position, it is fictitiously created to achieve such regularity, and subsequently it is removed from the resulting map. Additionally, consider that the robot is equipped with a laser rangefinder sensor (LiDAR), which measures distances in a front rank of 180°. Finally, suppose that it is known the exact position of the trees in the plantation corners: $P_{i,1}, P_{f,1}, P_{i,2}$ and $P_{f,2}$ (see Fig. 1).

3. Navigation

In order to build the map, it is desired that the robot can navigate autonomously, based in laser information only. It is clear that the sensory data are not simple to process, however it is possible to generate navigation strategies on this type of agricultural environments, which is an adaptation of the navigation technique in interior corridors developed by the authors (Carelli and Oliveira Freire, 2003).

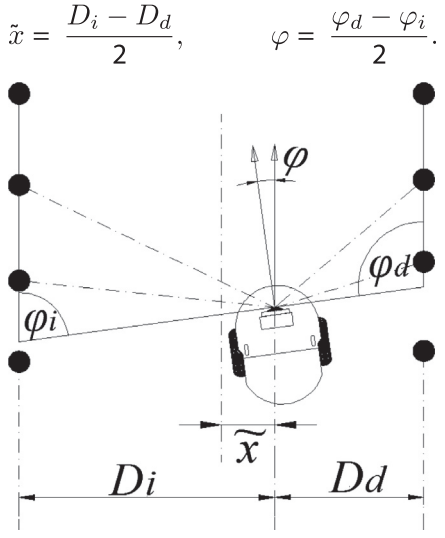


Fig. 2. Diagram of the system states.

3.1. Robot kinematic model

The kinematic equations for the robot in the corridor are (Carelli and Oliveira Freire, 2003):

$$\begin{cases} \dot{\tilde{x}} = u \sin\varphi \\ \dot{\varphi} = w, \end{cases}$$

where u and w are the control actions of an unicycle robot, and $[\tilde{x}\ \varphi]$ are the system errors, which identify the position and orientation errors of the vehicle in the corridor, as shown in Fig. 2. The path control objective is defined as:

$$[\tilde{x}(t), \varphi(t)] \to \mathbf{0}.$$

3.2. State equations using the laser sensor

The algorithm of the corridor following is focused on the right selection of the laser beams, where the two nearest trees, one in each side, and the next two between these are defined as references to determine the states of the system. Later, this information is used to calculate the orientation and position errors through the following equations (see Fig. 2):

$$\tilde{x} = \frac{D_i - D_d}{2}, \quad \varphi = \frac{\varphi_d - \varphi_i}{2}.$$

Based on these data, in Carelli and Oliveira Freire (2003) it is proposed the following control actions for navigation along the corridor,

$$v = (v_{\max} - k|\tilde{x}|)\cos\varphi,$$

$$w = -k_\varphi \varphi - k_x \tilde{x} v \sin\varphi/\varphi,$$

where, $v_{\max}$ is the maximum velocity of the unicycle mobile robot, and,

$$k_\varphi = \frac{k_1}{a_1 + |\varphi|} \quad \text{y} \quad k_x = \frac{k_2}{a_2 + |\tilde{x}|},$$

are defined by the design constants k, k_1, k_2, a_1 and a_2. These constants are selected to avoid saturation of control actions and to keep an adequate controller gain for small control errors.

At the end of the corridor, it is necessary to define a turning routine to continue with the plantation excursion. In this case, the nearest tree in one of the sides is used as pivot, and another three trees are taken empirically to define a virtual circular corridor around the pivot tree. Consequently, the same control can be used without switching the controller. The above described autonomous navigation algorithm is proved to successfully acquire in an autonomous way the measurements in a fruit grove for mapping, both in a simulation and a real framework, as it will be shown in next sections.

4. Measurement

During the navigation, the robot laser measures the distance and the direction to the nearest 4 trees of the corridor in which it is moving. In Section A.2, it is given details about the filtering process used to detect these trees. Based on this information, through classical trigonometric rules, it is possible to estimate iteratively the distances and the angles as drawn in Fig. 3.

In the time period in which the robot travels by a corridor, it collects a large number of measurements for each aforementioned segments and angles. With these observations the true measures

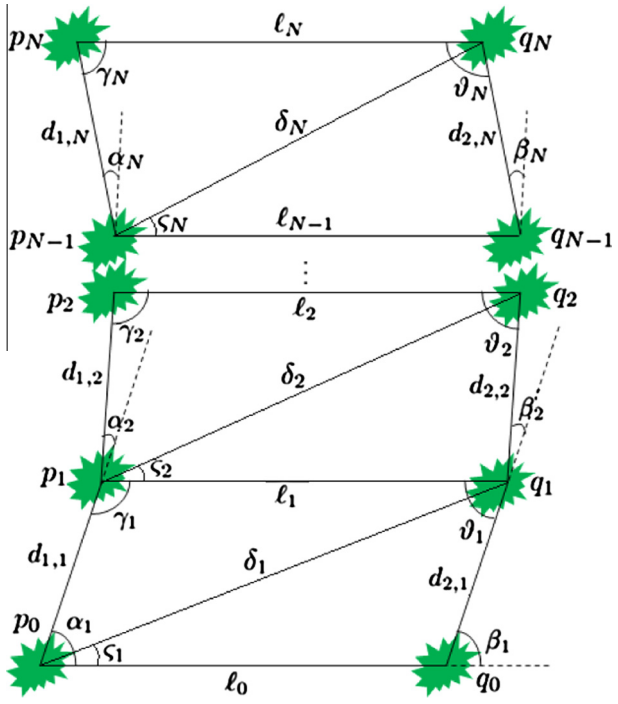


Fig. 3. Variables used in the fitting procedure.

are iteratively estimated with its mean and variance, by using the unbiased S^2 estimator (Ross, 1997).

5. Measures adjustment

Despite laser measurements are highly accurate, these have errors on account of the semi-structured environment and the rough terrain where the robot navigates. Furthermore, discretization of the laser directions generates additional errors. These measurement errors produce distorted and inconsistent maps. Nevertheless, errors arising in practice are small. This suggests that although the estimations produce an inconsistent measure set, there exist a compatible measure set close to the estimated one. In order to find such configuration, for each estimated measure θ_i, it is defined a new variable $\Delta\theta_i$, which represents the correction to be made about θ_i in search of a compatible configuration. In addition, the notation $\widehat{\theta}_i = \theta_i + \Delta\theta_i$ is used.

The proposed adjustment process is performed after the navigation and measurement steps are finished. At that point, it is available an estimation of each measure for each quadrilateral that constitutes the whole plantation. To subdivide the adjustment step (and thereby reduce the complexity of the problem), the correction is proposed in two stages: horizontal and vertical adjustments, as diagrammed in Fig. 4. In the first stage, the measurements of each quadrilateral that conform the pairs of extremal tree rows (above and below), highlighted on the left side of Fig. 4(a), are adjusted, and a location is assigned to each extremal tree highlighted on the right side of Fig. 4(a). Then, in the second stage, the vertical rows of trees are successively considered in pairs (see left side of Fig. 4(b)), where the corresponding measurements are adjusted and a location is assigned to each tree. After this process, all those trees with more than one assigned location, are relocated in the average position therebetween. Thereby, the problem of measure adjustment is reduced to perform multiple measure adjustments of two parallel and successive tree rows (vertical or horizontal). According to the ordering of adjustments raised (which is diagrammed in Fig. 4), at the beginning of each horizontal adjustment the locations p_0 and p_N (or symmetrically q_0 and q_N), diagrammed in Fig. 3, are firstly estimated. In contrast, when making the vertical adjustments, the four locations p_0, q_0, p_N and q_N, estimated at the horizontal adjustment stage, are assumed as known. The problems are solved using similar methods, which are described in Sections 5.1 and 5.2 respectively, where only the solution of the first one is explained in detail.

5.1. Horizontal adjustment

As described in the previous section, it should be first adjusted the measurements of the rectangles at the top two rows highlighted in the left part of Fig. 4(a). Turning 90° such rows and by performing simple operations, the estimations of the measurements diagrammed in Fig. 3 can be obtained. Moreover, it is known the exact locations of p_0 and p_N. The measurements α_i's, for example, are grouped in a vector α; the measurements β_i's in a vector β, and so on. Then, with these vectors, it is arranged the measurement set $\Theta = \{\alpha, \beta, \gamma, \vartheta, \varsigma, d_1, d_2, \ell, \delta\}$, where each $\theta \in \Theta$ is a vector of measurements θ_i of the same type, with $i \in I_\theta$, where I_θ is the set containing the integers beetwen 0 and N, or 1 and N, as appropriate.

The goal is to find measurement corrections $\Delta\theta_i$ which minimize the following sum

$$\Delta^2(\Theta) := \sum_{\theta\in\Theta}\sum_{i\in I_\theta} \frac{\Delta\theta_i^2}{V[\theta_i]}, \tag{1}$$

In this expression, it is considered the variances $V[\theta_i]$ in order to scale linear and angular measurements, which are estimated in the measuring step. Furthermore, a number of constraints should be satisfied to ensure compatibility of the measurement set. This represents a typical constrained optimization problem, which is solved by applying the Lagrange multipliers method (Brezhneva et al., 2012).

5.1.1. Notations

Before specifying the constraints, for simplicity the following notation is fixed. Given an angular vector $\theta \in \Theta$, for each $1 \leqslant i \leqslant N$, it is defined $c_i^\theta := \cos\widehat{\theta}_i, s_i^\theta := \sin\widehat{\theta}_i$, and the rotation matrices

$$R_i^\theta := \begin{bmatrix} c_i^\theta & -s_i^\theta \\ s_i^\theta & c_i^\theta \end{bmatrix}.$$

Moreover, given a distance vector $\rho \in \Theta$, for each $2 \leqslant i \leqslant N$, the following homogeneous transformation matrices are considered:

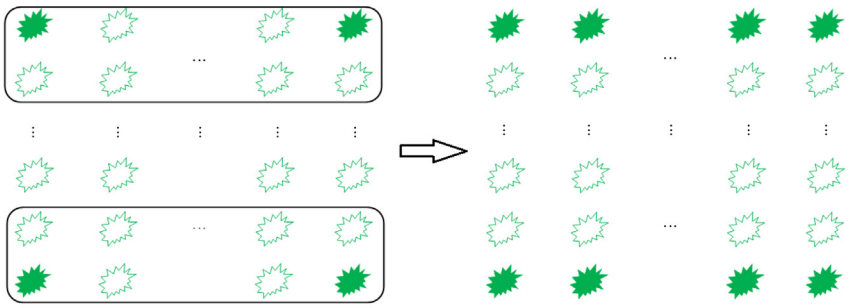
$$A_i^{\theta,\rho} := \begin{bmatrix} c_i^\theta & -s_i^\theta & c_i^\theta\widehat{\rho}_i \\ s_i^\theta & c_i^\theta & s_i^\theta\widehat{\rho}_i \\ 0 & 0 & 1 \end{bmatrix},$$

and

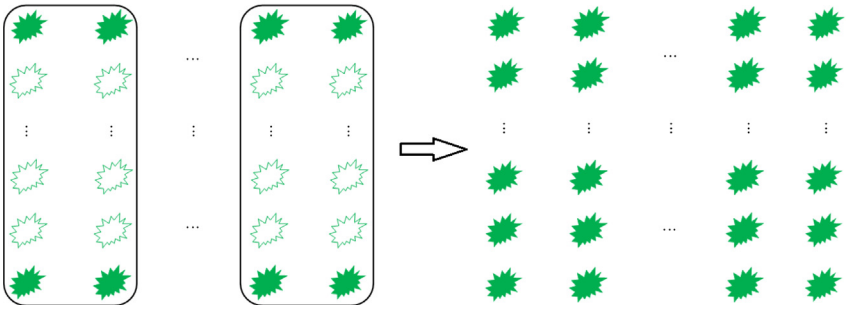
$$A_1^{\theta,\rho} := \begin{bmatrix} c_1^\theta & -s_1^\theta & c_1^\theta\widehat{\rho}_1 \\ s_1^\theta & c_1^\theta & s_1^\theta\widehat{\rho}_1 \end{bmatrix}.$$

The notations $R_{k:m}^\theta$ and $A_{k:m}^{\theta,\rho}$ will be used to denote the respective products $\prod_{i=k}^m R_i^\theta$ and $\prod_{i=k}^m A_i^{\theta,\rho}$, provided that $k \leqslant m$. Let $A_i^{-,\theta,\rho}$ denote the matrix that results from eliminating the last row of $A_i^{\theta,\rho}$ when $i > 1$, and by convention $A_1^{-,\theta,\rho} = A_1^{\theta,\rho}$. To reduce the derivative notation, let f'_{θ_i} be the partial derivative of f with respect to $\Delta\theta_i$, where θ_i is the i-th entry of any vector $\theta \in \Theta$.

Consider the fixed global coordinate axis system with center p_0 and abscissa axis on the line connecting the lower extremal trees.



(a) Step 1 : Horizontal adjustments : the extremal tree positions are estimated.



(b) Step 2 : Vertical adjustments : the position of all the remaining trees are estimated.

Fig. 4. Fusion procedure stages of the adjusted measures.

For each $0 \leqslant n \leqslant N$, let p_n and q_n be the points diagrammed in Fig. 3. Then, making an analogy with the homogeneous transformations used for an open-chain manipulator in robotics (Sciavicco and Siciliano, 2000), and considering each tree as a joint, it results that:

$$\widehat{p}_n = A_{1:n}^{\alpha,d_1} e_3 \quad \text{y} \quad \widehat{q}_n = A_{1:n}^{\beta,d_2} e_3 + [\widehat{\ell}_0, 0]^T, \tag{2}$$

where $e_3 := [0,0,1]^T$ denotes the homogeneous representation of the local coordinate system origin over each tree, and $\widehat{p}_n$ and $\widehat{q}_n$ are position estimations of p_n and q_n, respectively, in the global coordinate system.

5.1.2. Constraints

The constraints that arise are of four types: basic, framing, alignment and rectification. The basic constraints are

$$h_1(\Theta) := \widehat{\alpha}_1 - \widehat{\beta}_1 + \widehat{\gamma}_1 + \widehat{\vartheta}_1 - 180 = 0; \tag{3}$$

$$h_n(\Theta) := \widehat{\alpha}_n - \widehat{\beta}_n + \widehat{\gamma}_n + \widehat{\vartheta}_n - \widehat{\gamma}_{n-1} - \widehat{\vartheta}_{n-1} = 0; \; 2 \leqslant n \leqslant N;$$

which mean that the sum of interior angles of each quadrilateral diagrammed in Fig. 3 is 360°.

Second one, the framing constraints are the following N bidimensional identities

$$f_n(\Theta) := A_{1:n}^{\alpha,d_1} a_n^{\gamma,\ell} - \widehat{q}_n = \mathbf{0}, \quad 1 \leqslant n \leqslant N, \tag{4}$$

where $a_n^{\gamma,\ell} := [-c_n^{\gamma}\widehat{\ell}_n, -s_n^{\gamma}\widehat{\ell}_n, 1]^T$. These equations seek the framing of the measurements obtained, as diagrammed in Fig. 5 (note that there is a restriction for each n). The first term of (4) represents the expected location of q_n following the path of the left row, that is, the vector $[-c_n^{\gamma}\widehat{\ell}_n, -s_n^{\gamma}\widehat{\ell}_n]^T$ expressed in the principal coordinate system. On the other hand, the second term is $\widehat{q}_n$, which was defined in (2), taking account of the right path.

Additionally, the alignment constraint is given by

$$g(\Theta) := \widehat{p}_N + (p_0 - p_N) = \mathbf{0}. \tag{5}$$

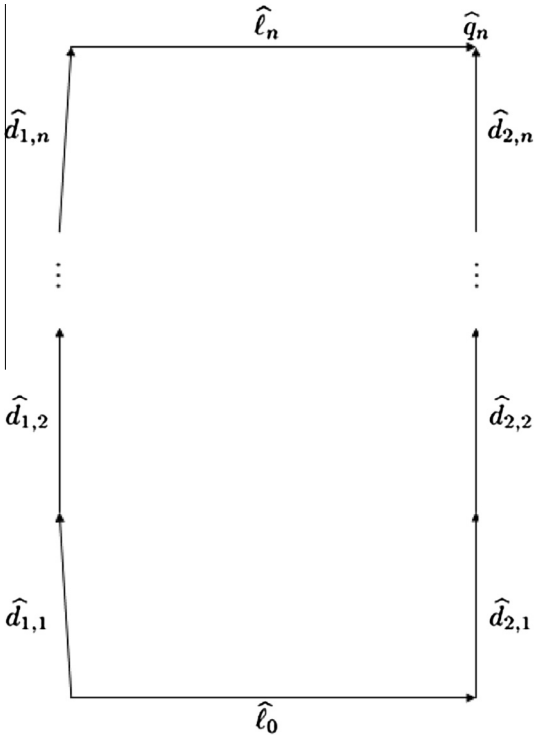


Fig. 5. Diagram of a framing constraint.

This constraint seeks that the corrections are compatible with the start and end points of the left row, is diagramed in Fig. 6.

Finally, rectification constraints are given by the following equations

$$a_1^{\varepsilon,\delta,\gamma} = \widehat{q}_1;$$

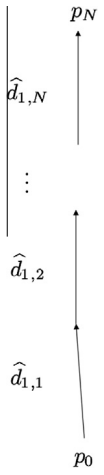


Fig. 6. Diagram of the alignment constraint.

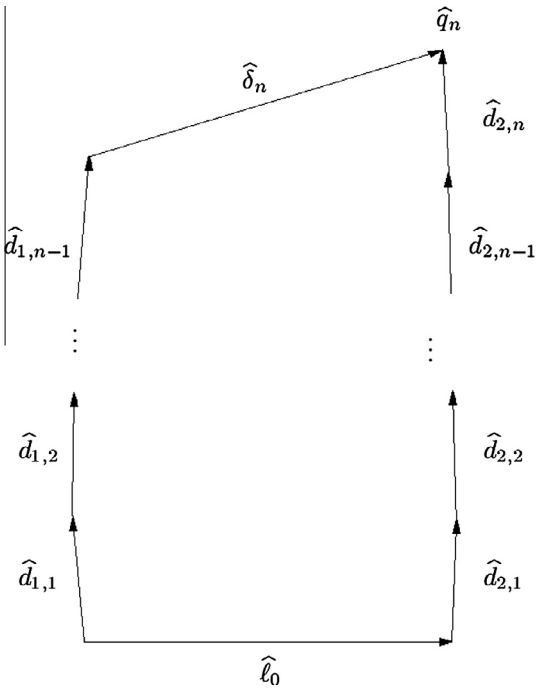


Fig. 7. Diagram of a rectification constraints.

$$A_{1:n-1}^{\alpha,d_1} a_n^{\varsigma,\delta,\gamma} = \widehat{q}_n, \quad 2 \leqslant n \leqslant N;$$

where $a_1^{\varsigma,\delta,\gamma} := \widehat{\delta}_1[c_1^{\varsigma}, s_1^{\varsigma}]^T$ and $a_n^{\varsigma,\delta,\gamma} := [-c_n^{\varsigma,\gamma}\widehat{\delta}_n, -s_n^{\varsigma,\gamma}\widehat{\delta}_n, 1]^T$ if $n > 1$, being $c_n^{\varsigma,\gamma} := \cos(\varsigma_n + \gamma_{n-1})$ and $s_n^{\varsigma,\gamma} := \sin(\varsigma_n + \gamma_{n-1})$. In this equation, $\widehat{q}_n$ is expressed following two different paths, as in (4), and such as diagrammed in Fig. 7 (note that there is a restriction for each n). These restrictions prevent the diagonals of the quadrilaterals stretch or shorten more than necessary in the adjustment process. From (4), the rectification constraints can be simplified as follows:

$$\phi_n(\Theta) := A_n^{-,\alpha,d_1} a_n^{\gamma,\ell} - a_n^{-,\varsigma,\delta,\gamma} = \mathbf{0}, \quad 1 \leqslant n \leqslant N; \tag{6}$$

where $a_1^{-,\varsigma,\delta,\gamma} = a_1^{\varsigma,\delta,\gamma}$ and $a_n^{-,\varsigma,\delta,\gamma} = -\widehat{\delta}_n[c_n^{\varsigma,\gamma}, s_n^{\varsigma,\gamma}]^T$ if $n > 1$.

5.1.3. Solution

The problem is reduced to an optimization problem, which consists in minimizing (1) under the constraints given in (3)–(6). To solve this problem, the following Lagrange function is defined

$$\Phi(\Theta, \Lambda) := \Delta^2(\Theta) + \sum_{\psi \in \mathcal{F}} \sum_{n \in I_\psi} \lambda_{\psi,n}^T \psi_n(\Theta), \tag{7}$$

where $\mathcal{F} := \{f, g, h, \phi\}, I_\psi$ is an index set which varies according to ψ, encompassing all its coordinates, and $\Lambda = \{\lambda_f, \lambda_g, \lambda_h, \lambda_\phi\}$ is the set containing the vectors of Lagrange multipliers, which has $5N + 2$ scalar entries in total.

Then, the solution to the problem is given by the solution of the following system:

$$\frac{\partial}{\partial \Delta\theta_i} \Phi(\Theta, \Lambda) = 0, \quad \forall \theta \in \Theta,\ i \in I_\theta;$$

$$\psi_n(\Theta) = \mathbf{0}, \quad \forall \psi \in \mathcal{F},\ n \in I_\psi. \tag{8}$$

This system contains $14N + 3$ equations and $14N + 3$ variables, including the entries of Λ and the entry adjustments in each vector of Θ. The equations are listed in Section A.1.

To avoid solving a system of this magnitude, it is found a subset $\Gamma \subset \Theta \cup \Lambda$ of independent variables, as small as possible, which allows writing the other variables as a function of them. Then, replacing in (7) the entries of the vectors of $(\Theta \cup \Lambda) \setminus \Gamma$ in terms

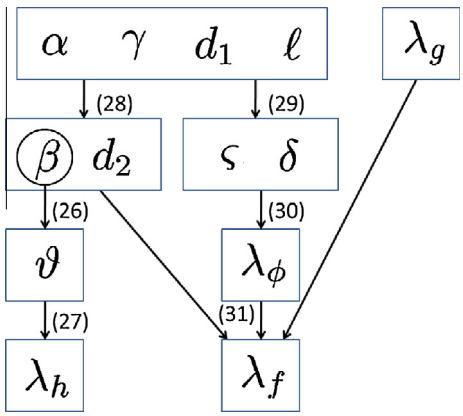


Fig. 8. Variables dependance diagram. Dependence functions are given in the Appendix (see equation number indicated in parentheses).

of the entries of the vectors of Γ, it results a smaller system from which a solution in Γ can be obtained.

According with the above argument, in the measurement setting problem it is considered $\Gamma = \{\alpha, \gamma, d_1, \ell, \lambda_g\}$ with the other variables expressed in function of them, resulting the dependency diagram that is shown in Fig. 8. Therein, it is seen that there are no cycles involving inconsistencies in the dependence flow. Details of this can be found in Section A.1. Thus the resulting system contains $4N + 1$ equations and $4N + 1$ variables. After solving the optimization problem, with variables of Γ obtained, the other variables can be calculated using the dependance functions given in the Appendix.

5.2. Vertical adjustment

The vertical adjustments can also be performed using the procedure outlined above, however, these adjustments can be performed more efficiently because the horizontal adjustments have assigned a location for each extremal tree diagrammed in Fig. 4(b). This extra information can be exploited by solving a similar optimization problem, which is briefly discussed below.

Suppose that the extremal tree positions p_0, q_0, p_N and q_N are known (see Fig. 3). Consequently, the distances ℓ_0 and ℓ_N are also known, but they are not considered in the adjustment process. Furthermore, for simplicity and practical arguments, the variables δ_i and ς_i are not taken into account. Hence, in this problem, $\Theta := \{\alpha, \beta, \gamma, \vartheta, d_1, d_2, \ell\}$.

Three types of constrains are considered: basic, framing and alignment. Basic and framing constrains are the same as defined in (3) and (4) respectively. Since in this case two extra position of extremal trees are known, it is added to (5) the following similar alignment constraint:

$$\widehat{q}_N + (q_0 - q_N) = \mathbf{0}. \tag{9}$$

These new hypotheses generate an analogous optimization problem which is similarly solved.

6. Simulations

To simulate the proposal, Matlab is used for calculations, and MobileSim program is used to simulate the dynamic model of a robot Pioner p3dx-sh (http://robots.mobilerobots.com/wiki/MobileSim). For the connection between Matlab and MobileSim a C++ interface based on shared memory resource is used, which allows to read the laser range sensor and generate online control commands.

6.1. First experience: regular olive grove

For the first experience, it is generated a virtual olive grove composed of 10 rows of 30 trees each. It is assumed that the grove is regular, that is, all quadrilaterals that make up each path (see Fig. 3) are equal rectangles. In Fig. 9, the simulated robot navigates successfully through this virtual scenario using the reactive control system previously presented.

The horizontal adjustment, which was explained in Section 5.1, aims at estimating the positions of the extremal olive trees. To do this, it should be considered the measurements of the quadrilaterals highlighted in Fig. 4(a), and to apply twice the adjustment procedure developed in Section 5.1.

The structure diagrammed in Fig. 10(a) shows that the measurements acquired in the row of upper extremal quadrilaterals are inconsistent. It can be seen that the basic constraints (3) are not satisfied, since the quadrilaterals are not closed. The two segment sets with approximately vertical direction are plotted according to the vectors α, d_1, β and d_2, obtaining approximations $\widehat{p}_n$ and $\widehat{q}_n$ of the points p_n and q_n diagrammed in Fig. 3. The left curve shows that alignment constraint (5) is not verified. The segments with approximately horizontal direction are plotted with starting point in $\widehat{p}_n$, direction according to γ_n and module ℓ_n, for $n = 1, \ldots, N$. It can be seen that the framing constraints (4) are not satisfied. Segments with approximately diagonal direction are plotted with starting point in $\widehat{p}_{n-1}$, direction according to γ_{n-1} and ς_n, and module δ_n, for $n = 1, \ldots, N$. It can be seen that the rectification constraints (6) are not verified.

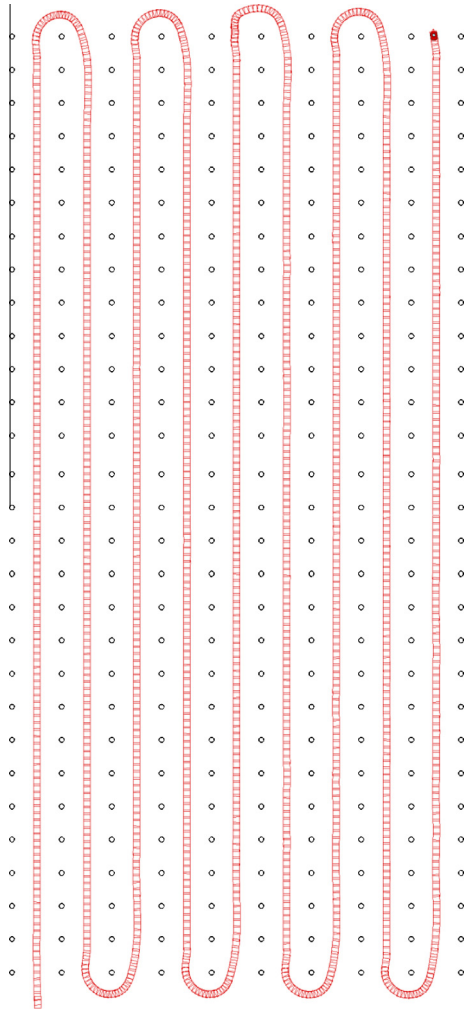
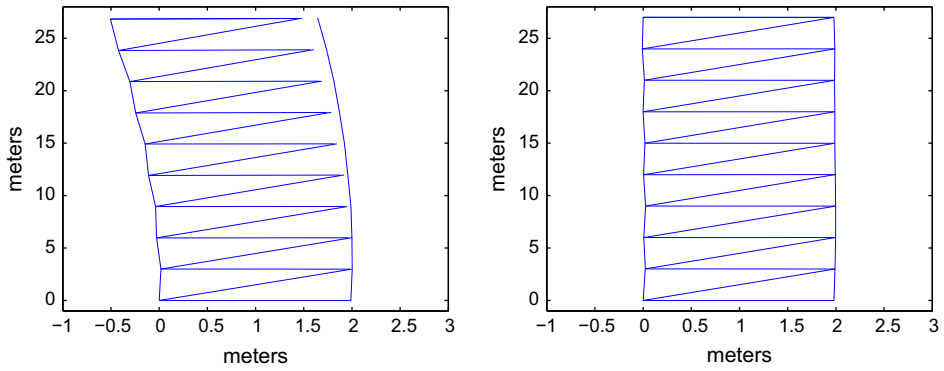


Fig. 9. Robot navigation path.

An analogous schematization is performed in Fig. 10(b), but considering the adjusted measurements. At this point, it can be seen that these measurements verify the constraints posed in Section 5.1.2. While an alignment constraint for right vertical curve was not proposed, it achieves an acceptable adjustment caused by the other constraints.

The final estimations $\widehat{p}_n$ determine locations to the extremal trees (see Fig. 4(a)). In the example presented, the average error distance is 1.30 cm, reaching a maximum error of 2.61 cm.

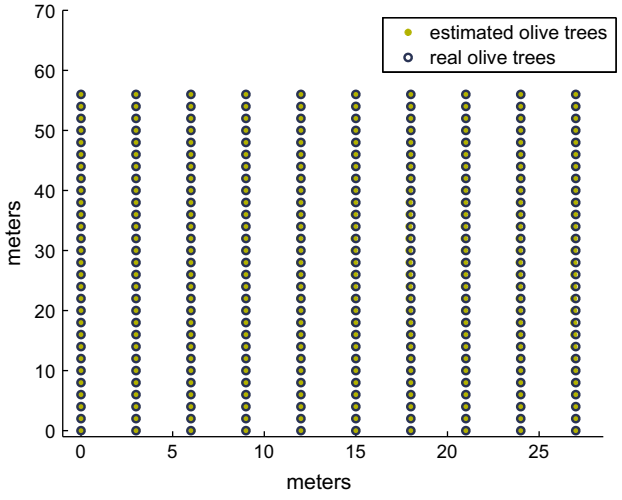
Later, the vertical adjustments were made (see Fig. 4(b)). The estimated and real locations of all olive are illustrated in Fig. 11(a). The average error is 4.30 cm, reaching a maximum error of 11.67 cm. A histogram of the mapping error is shown in Fig. 11(b), in which it can be observed that the error remains close to 0.



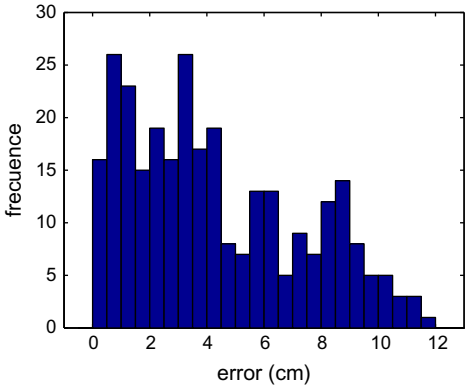
(a) Tree structure according to the original measurements.

(b) Tree structure according to the estimated measurements.

Fig. 10. Outlining of the performance of the measure adjustment method based on knowledge of the position of the two left extremal olive trees.



(a) Real and estimated positions of the olive trees.



(b) Error histogram in the grove mapping procedure.

Fig. 11. Comparison between actual and estimated positions of the olive trees.

6.2. Second experience: irregular olive grove

As complementary experience, a new virtual olive grove composed of trees irregularly located was simulated. Their locations, obtained by randomly displacing of the tree locations of a regular olive grove, and a mobile robot with its laser simulated in the MobileSim software are shown in Fig. 12(a). A comparative graph of the actual and estimated localization by the procedure is given in Fig. 12(b). The average location error was of 13.76 cm, peaking at 33.02 cm.

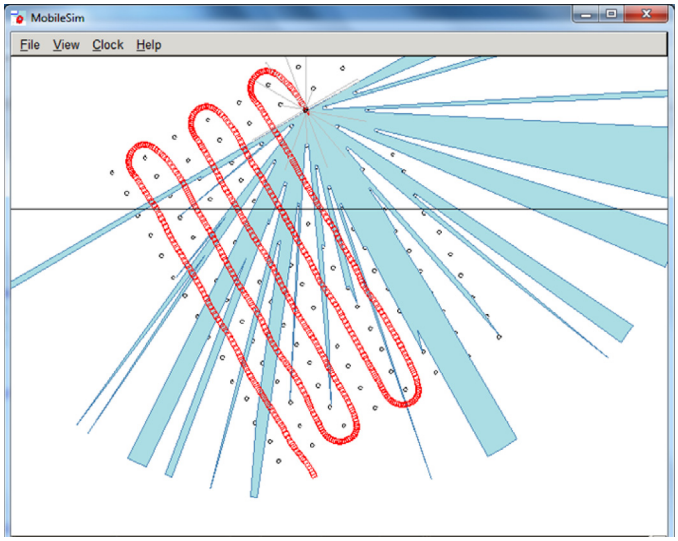
7. Experimentation with real data

In the previous sections it has been detailed the mapping procedure, which consists in the following steps: laser sensing, measurement, navigation, adjustment and grove mapping. Working with real data inevitably needs to consider a filtering process between the sensing and measurement stages. This procedure should be robust and should provide a position estimation of the 4 closest olive trees within the path by which the robot navigates. Also, it should be able to diagnose the lack of a tree in an expected location, and the virtual assignment of a tree in such place, to ensure efficient continuity of navigation and measurement steps. Details of the used of the filtering procedure are given in the Appendix (Section A.2).

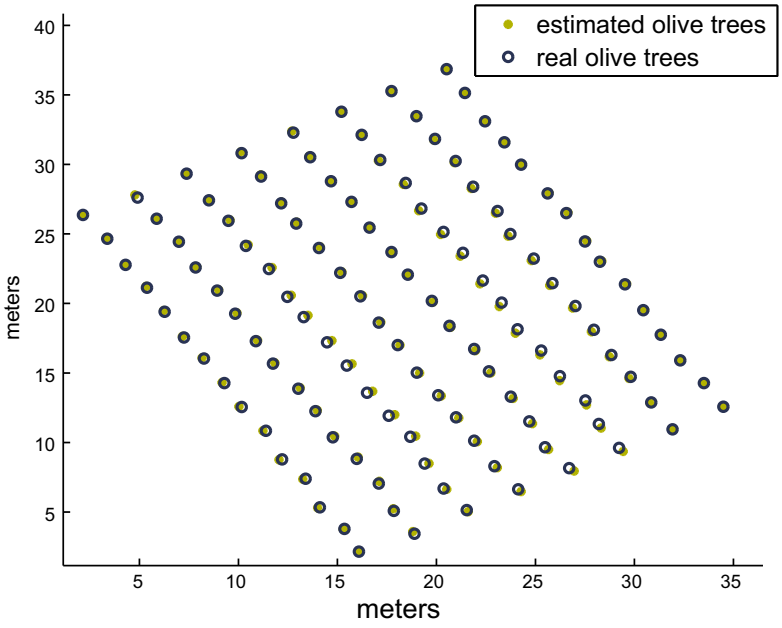
To verify the performance of the proposed algorithm, an experiment was carried out in the experimental groves at the National Institute of Agricultural Technology (INTA) located in San Juan, Argentina. It was used a robot Pioneer 3-AT (P3-AT) from ActiveMedia equipped with a laser rangefinder sensor (see Fig. 13(a)). In Fig. 13(b), an image of the olive grove was overlaid on the resultant mapping. Qualitatively, the results are consistent and satisfactory for a practical application in precision agriculture. In the simulations, the peak errors were observed in the central tree rows. Hence, to objectively quantify the performance of the method, the GPS coordinates and the estimated positions of three central rows of trees were compared. The average error was 36.03 cm and error standard deviation was 17.91 cm, peaking at 52.03 cm.

8. Conclusions and future work

In this paper, it has been presented a mapping procedure that exploits the semi-structurality of an intensive agricultural environment. It is based only on laser measurements and the knowledge of absolute positions of extremal corner trees, to obtain an absolute map of the entire grove. The methodology is based on an optimization technique that is applied off-line and that uses the metric information gathered by a robot during an autonomous navigation stage. Additionally, in experiments with real data, a filtering technique of laser observations was implemented. This allows to identify the positions of trees in a robust and consistent manner, thus obtaining a useful map of the grove to be used in precision agriculture applications, both for autonomous navigation and for plant treatment and monitoring.



(a) Window of the MobileSim software in which a robot-mobile is navigating between olive trees randomly located.



(b) Real and estimated positions of the olive trees.

Fig. 12. Distribution of fruit trees and results obtained in the second simulation.

The performance of the technique was tested by simulations and by experimentation on a real grove, providing satisfactory results with estimation errors within the acceptable ones for typical precision agriculture applications. Furthermore, it has great potential for future improvements and can be used as a complementary tool to procedures that include the localization problem as SLAM, simplifying issues such as the observations-features matching.

As future work, it is desired to incorporate a localization procedure by triangulation regarding the location of the closest trees. This would further reduce the mapping errors incorporating additional information, such as odometry and visual data.

Appendix A

A.1. Solution and simplification of the optimization problem

In this section, the resolution procedure of the optimization problem is deduced. The system (8) consists of the following equations:

$$1 \leqslant i \leqslant N \Rightarrow \frac{2\Delta\alpha_i}{V[\alpha_i]} + \lambda_{h,i} + \lambda_g^T (A_{1:N}^{\alpha,d_1})'_{\alpha_i} e_3 + \sum_{n=i}^{N} \lambda_{f,n}^T (A_{1:n}^{\alpha,d_1})'_{\alpha_i} a_n^{\gamma,\ell} + \lambda_{\phi,i}^T (A_i^{-,\alpha,d_1})'_{\alpha_i} a_i^{\gamma,\ell} = 0; \tag{10}$$

$$1 \leqslant i \leqslant N \;\Rightarrow\; \frac{2\Delta\beta_i}{V[\beta_i]} + \lambda_{h,i} - \sum_{n=i}^{N} \lambda_{f,n}^T (A_{1:n}^{\beta,d_2})'_{\beta_i} e_3 = 0 \tag{11}$$

(a) Robot Pioneer 3-AT (P3-AT) used for experiment.

(b) Comparison between real and estimated positions of the olive trees.

Fig. 13. Images of the experiment with real data.

$$1 \leqslant i \leqslant N-1 \;\Rightarrow\; \frac{2\Delta\gamma_i}{V[\gamma_i]} + \lambda_{h,i} - \lambda_{h,i+1}\lambda_{f,i}^T A_{1:i}^{\beta,d_2}\left(a_i^{\gamma,\ell}\right)'_{\gamma_i} + \lambda_{\phi,i}^T A_i^{-,\alpha,d_1}\left(a_i^{\gamma,\ell}\right)'_{\gamma_i} - \lambda_{\phi,i+1}^T\left(a_{i+1}^{-,\varsigma,\delta,\gamma}\right)'_{\gamma_i} = 0 \tag{12}$$

$$\frac{2\Delta\gamma_N}{V[\gamma_N]} + \lambda_{h,N} + \lambda_{f,N}^T A_{1:N}^{\beta,d_2}\left(a_N^{\gamma,\ell}\right)'_{\gamma_N} + \lambda_{\phi,N}^T A_N^{-,\alpha,d_1}\left(a_N^{\gamma,\ell}\right)'_{\gamma_N} = 0 \tag{13}$$

$$1 \leqslant i \leqslant N-1 \;\Rightarrow\; \frac{2\Delta\vartheta_i}{V[\vartheta_i]} + \lambda_{h,i} - \lambda_{h,i+1} = 0 \tag{14}$$

$$\frac{2\Delta\vartheta_N}{V[\vartheta_N]} + \lambda_{h,N} = 0 \tag{15}$$

$$1 \leqslant i \leqslant N \;\Rightarrow\; \frac{2\Delta\varsigma_i}{V[\varsigma_i]} - \lambda_{\phi,i}^T\left(a_i^{-,\varsigma,\delta,\gamma}\right)'_{\varsigma_i} = 0; \tag{16}$$

$$1 \leqslant i \leqslant N \;\Rightarrow\; \frac{2\Delta\delta_i}{V[\delta_i]} - \lambda_{\phi,i}^T\left(a_i^{-,\varsigma,\delta,\gamma}\right)'_{\delta_i} = 0; \tag{17}$$

$$\frac{2\Delta\ell_0}{V[\ell_0]} - \sum_{n=1}^{N}[1,0]\lambda_{f,n} = 0; \tag{18}$$

$$1 \leqslant i \leqslant N \;\Rightarrow\; \frac{2\Delta\ell_i}{V[\ell_i]} - \lambda_{f,i}^T A_{1:i}^{\alpha,d_1}\left(a_i^{\gamma,\ell}\right)'_{\ell_i} + \lambda_{\phi,i}^T A_i^{-,\alpha,d_1}\left(a_i^{\gamma,\ell}\right)'_{\ell_i} = 0; \tag{19}$$

$$1 \leqslant i \leqslant N \;\Rightarrow\; \frac{2\Delta d_{1,i}}{V[d_{1,i}]} + \lambda_g^T\left(A_{1:N}^{\alpha,d_1}\right)'_{d_{1,i}} e_3 + \sum_{n=i}^{N}\lambda_{f,n}^T\left(A_{1:n}^{\alpha,d_1}\right)'_{d_{1,i}} a_n^{\gamma,\ell} + \lambda_{\phi,i}^T\left(A_i^{-,\alpha,d_1}\right)'_{d_{1,i}} a_n^{\gamma,\ell} = 0; \tag{20}$$

$$1 \leqslant i \leqslant N \;\Rightarrow\; \frac{2\Delta d_{2,i}}{V[d_{2,i}]} - \sum_{n=i}^{N}\lambda_{f,n}^T\left(A_{1:n}^{\beta,d_2}\right)'_{d_{2,i}} e_3 = 0; \tag{21}$$

$$1 \leqslant n \leqslant N \;\Rightarrow\; f_n(\Theta) = \mathbf{0}; \tag{22}$$

$$g(\Theta) = \mathbf{0}; \tag{23}$$

$$1 \leqslant n \leqslant N \;\Rightarrow\; h_n(\Theta) = 0; \tag{24}$$

$$1 \leqslant n \leqslant N \;\Rightarrow\; \phi_n(\Theta) = 0; \tag{25}$$

which is a system of $14N + 3$ equations and $14N + 3$ variables.

From (24), recursively it is obtained that

$$\Delta\vartheta_1 = 180 - \widehat{\alpha}_1 + \widehat{\beta}_1 - \widehat{\gamma}_1 - \vartheta_1;$$

$$\Delta\vartheta_i = -\widehat{\alpha}_i + \widehat{\beta}_i - \widehat{\gamma}_i - \vartheta_i + \widehat{\gamma}_{i-1} + \widehat{\vartheta}_{i-1},\ i = 2, \ldots, N; \tag{26}$$

which expresses the variables $\Delta\vartheta_i$ in terms of the other variables of the system and reduces the problem complexity.

Similarly, from (14) and (15), the variables $\lambda_{h,i}$ are isolated as a function of the variables $\Delta\vartheta_i$, resulting the following recursive expression

$$\lambda_{h,N} = -\frac{2\Delta\vartheta_N}{V[\vartheta_N]};$$

$$\lambda_{h,i} = \lambda_{h,i+1} - \frac{2\Delta\vartheta_i}{V[\vartheta_i]}, \quad i = N-1, \ldots, 1. \tag{27}$$

Using (22), the variables $\Delta d_{2,i}$ and $\Delta\beta_i$ can be recursively calculated, through the following equations:

$$\Delta d_{2,1} = \left\| A_1^{\alpha,d_1} a_1^{\gamma,\ell} - [\widehat{\ell}_0, 0]^T \right\|_2 - d_{2,1};$$

$$\Delta\beta_1 = \operatorname{ang}\left(A_1^{\alpha,d_1} a_1^{\gamma,\ell} - [\widehat{\ell}_0, 0]^T \right) - \beta_1; \tag{28}$$

$$\Delta d_{2,i} = \left\| R_{1:i-1}^{-\beta,d_2} (A_{1:i}^{\alpha,d_1} a_i^{\gamma,\ell} - [\widehat{\ell}_0, 0]^T - b_i^{\beta,d_2}) \right\|_2 - d_{2,i};$$

$$\Delta\beta_i = \operatorname{ang}\left(R_{1:i-1}^{-\beta,d_2} \left(A_{1:i}^{\alpha,d_1} a_i^{\gamma,\ell} - [\widehat{\ell}_0, 0]^T - b_i^{\beta,d_2} \right) \right) - \beta_i;$$

where $\operatorname{ang}([x, y]) = \operatorname{atan2}(y, x)$ for each $x, y \in IR$, and b_i^{β,d_2} is the third column of $A_{1:i-1}^{\beta,d_2}$.

Analogously, from (25), if $i > 1$, then

$$-\widehat{\ell}_i R_i^{\alpha} \begin{bmatrix} c_i^{\gamma} \\ s_i^{\gamma} \end{bmatrix} + \widehat{d}_{1,i} \begin{bmatrix} c_i^{\alpha} \\ s_i^{\alpha} \end{bmatrix} = -\widehat{\delta}_i R_{i-1}^{\gamma} \begin{bmatrix} c_i^{\varsigma} \\ s_i^{\varsigma} \end{bmatrix}.$$

Then, considering that

$$R_i^{\alpha} \begin{bmatrix} c_i^{\gamma} \\ s_i^{\gamma} \end{bmatrix} = \begin{bmatrix} c_i^{\alpha+\gamma} \\ s_i^{\alpha+\gamma} \end{bmatrix},$$

it is obtained that

$$\Delta\delta_i = \left\| R_{i-1}^{-\gamma} \left(\widehat{\ell}_i \begin{bmatrix} c_i^{\alpha+\gamma} \\ s_i^{\alpha+\gamma} \end{bmatrix} - \widehat{d}_{1,i} \begin{bmatrix} c_i^{\alpha} \\ s_i^{\alpha} \end{bmatrix} \right) \right\|_2 - \delta_i;$$

$$\Delta\varsigma_i = \operatorname{ang}\left(R_{i-1}^{-\gamma} \left(\widehat{\ell}_i \begin{bmatrix} c_i^{\alpha+\gamma} \\ s_i^{\alpha+\gamma} \end{bmatrix} - \widehat{d}_{1,i} \begin{bmatrix} c_i^{\alpha} \\ s_i^{\alpha} \end{bmatrix} \right) \right) - \varsigma_i. \tag{29}$$

In the same way

$$\Delta\delta_1 = \left\| \widehat{d}_{1,1} \begin{bmatrix} c_1^{\alpha} \\ s_1^{\alpha} \end{bmatrix} - \widehat{\ell}_1 \begin{bmatrix} c_1^{\alpha+\gamma} \\ s_1^{\alpha+\gamma} \end{bmatrix} \right\|_2 - \delta_1;$$

$$\Delta\varsigma_1 = \operatorname{ang}\left(\widehat{d}_{1,1} \begin{bmatrix} c_1^{\alpha} \\ s_1^{\alpha} \end{bmatrix} - \widehat{\ell}_1 \begin{bmatrix} c_1^{\alpha+\gamma} \\ s_1^{\alpha+\gamma} \end{bmatrix} \right) - \varsigma_1.$$

From (16) and (17), it follows that

$$\lambda_{\phi,1} = \begin{bmatrix} -s_1^{\varsigma} & c_1^{\varsigma} \\ c_1^{\varsigma} & s_1^{\varsigma} \end{bmatrix} \begin{bmatrix} \frac{2\Delta\varsigma_1}{\widehat{\delta}_1 V[\varsigma_1]} \\ \frac{2\Delta\delta_1}{V[\delta_1]} \end{bmatrix};$$

$$2 \leqslant i \leqslant N \;\Rightarrow\; \lambda_{\phi,i} = \begin{bmatrix} s_i^{\varsigma,\gamma} & -c_i^{\varsigma,\gamma} \\ -c_i^{\varsigma,\gamma} & -s_i^{\varsigma,\gamma} \end{bmatrix} \begin{bmatrix} \frac{2\Delta\varsigma_i}{\widehat{\delta}_i V[\varsigma_i]} \\ \frac{2\Delta\delta_i}{V[\delta_i]} \end{bmatrix}. \tag{30}$$

In order to isolate the variables $\lambda_{f,i}$, the following variables are defined

$$\eta_i = \sum_{n=i}^{N} \lambda_{f,n}.$$

Following with this, given $1 \leqslant i \leqslant N$, from (20) and (21), it is deduced that

$$\frac{2\Delta d_{1,i}}{V[d_{1,i}]} + \lambda_g^T \begin{bmatrix} c_{1:i}^{\alpha} \\ s_{1:i}^{\alpha} \end{bmatrix} + \eta_i^T \begin{bmatrix} c_{1:i}^{\alpha} \\ s_{1:i}^{\alpha} \end{bmatrix} + \lambda_{\phi,i}^T \begin{bmatrix} c_i^{\alpha} \\ s_i^{\alpha} \end{bmatrix} = 0;$$

$$\frac{2\Delta d_{2,i}}{V[d_{2,i}]} - \eta_i^T \begin{bmatrix} c_{1:i}^{\beta} \\ s_{1:i}^{\beta} \end{bmatrix} = 0,$$

where $s_{1:i}^{\theta} := \sin(\sum_{j=1}^{i} \widehat{\theta}_i)$ and $c_{1:i}^{\theta} := \cos(\sum_{j=1}^{i} \widehat{\theta}_i)$. Then

$$\eta_i = \begin{bmatrix} c_{1:i}^{\alpha} & s_{1:i}^{\alpha} \\ -c_{1:i}^{\beta} & -s_{1:i}^{\beta} \end{bmatrix}^{-1} \begin{bmatrix} \upsilon_1 \\ \upsilon_2 \end{bmatrix},$$

where

$$\upsilon_1 := -\frac{2\Delta d_{1,i}}{V[d_{1,i}]} - \lambda_g^T \begin{bmatrix} c_{1:i}^{\alpha} \\ s_{1:i}^{\alpha} \end{bmatrix} - \lambda_{\phi,i}^T \begin{bmatrix} c_i^{\alpha} \\ s_i^{\alpha} \end{bmatrix};$$

$$\upsilon_2 := -\frac{2\Delta d_{2,i}}{V[d_{2,i}]}.$$

Hence

$$\lambda_{f,i} = \begin{cases} \eta_i - \eta_{i+1}, & \text{si } i < N \\ \eta_N & \text{si } i = N. \end{cases} \tag{31}$$

Finally, as an extra and complementary step to what is diagrammed in Fig. 8, from (23) it is obtained that:

$$\Delta d_{1,N} = \left\| R_{1:N-1}^{-\alpha} (P_{f,1} - P_{i,1} - b_{N-1}^{\alpha,d_1}) \right\|_2 - d_{1,N},$$

$$\Delta\alpha_N = \operatorname{ang}\left(R_{1:N-1}^{-\alpha} (P_{f,1} - P_{i,1} - b_{N-1}^{\alpha,d_1}) \right) - \alpha_N, \tag{32}$$

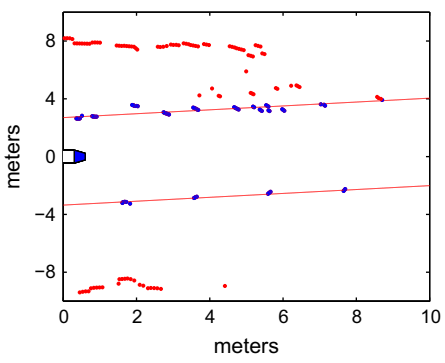
where b_{N-1}^{α,d_1} is the third column of $A_{1:N-1}^{\alpha,d_1}$.

Substituting Eqs. (26)–(32), in Eqs. (10)–(13) and (18),(19), the problem is reduced to solving a system with $4N + 1$ equations and $4N + 1$ unknowns.

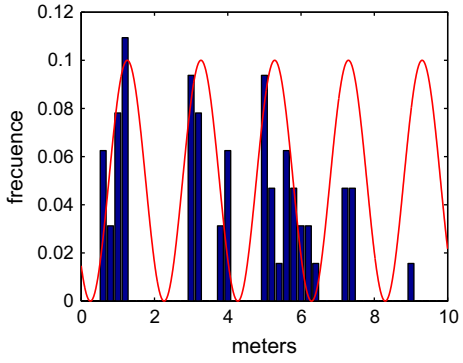
A.2. Data filtering

In this section, the data filtering process used in the mapping experience with real data is detailed. The idea is to keep working with laser information only, with the specific aim of estimating the positions of the four nearest olive trees needed for navigation and measurement steps. It works under the assumption that the robot knows an estimate of the average distance between rows, which can be initially estimated by the robot.

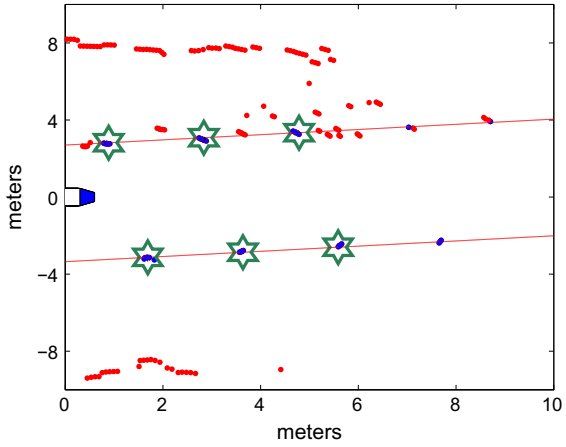
Considering the stability of the navigation algorithm, it is assumed that the robot navigates approximately through the path center and with orientation parallel to the tree rows. Then it becomes easy to identify and delete data from adjacent corridors. Moreover, it can be distinguished the data from the left and right rows of the actual corridor, and fit a straight line by least squares for each data subset. The average direction between them is regarded as estimation of the tree row orientations, which is temporarily filtered with a predetermined window size. Data that are beyond a maximum tolerance of the lines are discarded (see



(a) Orientation estimation of the olive trees rows.



(b) Histogram of the trees locations orthogonally projected with respect to the previously estimated orientation and plotting of the (appropriately scaled) curve fitted by ML.



(c) Estimated location of the nearest trees.

Fig. 14. Diagramation of the filtering procedure.

Fig. 14(a)). If unfiltered observations are projected over a line with the orientation resulting of the filtering process, a distribution of points $X_1, \ldots, X_n$ with approximately sinusoidal histogram is obtained (see Fig. 14(b)).

Then it can be fitted a density of the form

$$f(x) \propto \cos\left(2\pi\frac{x-\mu}{\sigma}\right) + 1, x \in [0, M];$$

where M is the maximum range of the laser, μ a particular peak of the density, and σ the distance between two consecutive peaks. Maximum likelihood estimators (ML) of the parameters μ and σ of this multimodal density are given by the solution of the following system:

$$\sum_{i=1}^{n} \tan\left(\pi\frac{X_i-\mu}{\sigma}\right) = 0,$$

$$\sum_{i=1}^{n} X_i \tan\left(\pi\frac{X_i-\mu}{\sigma}\right) = 0. \quad (33)$$

This fitting is also diagrammed in Fig. 14(b) and determines the sites over the fitted lines where the olive trees should be. Those observations that are located at a distance greater than a predetermined amount are discarded (see Fig. 14(c)). In this way, there is a cluster of observations for each site at which there is high probability to find a tree. For each cluster, a representative central observation is selected as estimation of the corresponding tree position. In the absence of an observation close to the expected site, it is assigned one at that site. This can happen for two reasons: lack of a tree or errors in sensing and filtering steps. In the absence of an olive tree, it must be assigned an observation in the expected site in order to complete the information necessary to perform the navigation and the measurement adjustment procedure. If this situation is repeatedly given in the same place, the corresponding olive tree is removed from the resulting mapping. If the lack of observations is due to errors in the sensing and filtering steps, this allocation redresses the error, which occurs rarely in practice.

References

Adams, M., Vo, B., Mahler, R.P.S., Mullane, J., 2014. SLAM gets a PHD: new concepts in map estimation. IEEE Robot. Autom. Mag. 21 (2), 26–37.
Auat Cheein, F., Carelli, R., 2013. Agricultural robotics: unmanned robotic service units in agricultural tasks. IEEE Indus. Electron. Mag. 7 (3), 48–58.
Auat Cheein, F., Steiner, G., Perez Paina, G., Carelli, R., 2011. Optimized EIF-SLAM algorithm for precision agriculture mapping based on stems detection. Comput. Electron. Agri. 78, 195–207.
Ayache, N., Faugeras, O., 1989. Maintaining a representation of the environment of a mobile robot. IEEE Trans. Robot. Autom. 5 (6), 804–819.
Brezhneva, O., Tretyakov, A.A., Wright, S.E., 2012. A short elementary proof of the lagrange multiplier theorem. Optim. Lett. 6 (8), 1597–1601.
Bryson, M., Sukkarieh, S., 2008. Observability analysis and active control for airborne SLAM. IEEE Trans. Aerospace Electron. Syst. 44 (1), 261–280.
Carelli, R., Oliveira Freire, E., 2003. Corridor navigation and wall-following stable control for sonar-based mobile robots. Robot. Autonom. Syst. 45 (3), 235–247.

Chatila, R., Laumond, J., 1985. Position referencing and consistent world modeling for mobile robots. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 2, pp. 138–145.

Durrant-Whyte, H., Bailey, T., 2006. Simultaneous localization and mapping (slam): part i essential algorithms. IEEE Robot. Autom. Mag. 13 (2), 99–108.

Durrant-Whyte, H., Bailey, T., 2006. Simultaneous localization and mapping (slam): part ii state of the art. IEEE Robot. Autom. Mag. 13 (3), 108–117.

Jin, J., Tang, L., 2009. Corn plant sensing using real-time stereo vision. J. Field Robot. 26 (6–7), 591–608.

Lee, S., Yeoh, K., Soetedjo, A., 2008. Developing a blind robot: study on 2d mapping. In: IEEE Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA), pp. 12–14.

Libby, J., Kantor, G., 2011. Deployment of a point and line feature localization system for an outdoor agriculture vehicle. In: IEEE International Conference on Robotics and Automation (ICRA).

Ouellette, R., Hirasawa, K., 2008. Mayfly: a small mapping robot for japanese office environments. IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), pp. 880–885.

Ross, S.M., 1997. Simulation. In: Statistical Modeling and Decision Science, second ed. Academic Press.

Rovira-Mas, F., 2009. Recent innovations in off-road intelligent vehicles: in-field automatic navigation. Recent Patents Mech. Eng. 2 (3), 169–178.

Sciavicco, L., Siciliano, B., 2000. Modelling and Control of Robot Manipulators, second ed. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Srinivasan, A., 2006. Handbook Of Precision Agriculture: Principles And Applications. CRC Press.

Thrun, S., Burgard, W., Fox, D., 2005. Probabilistic Robotics. MIT Press.

Xiaogang, R., Xuetao, X., 2009. Application of autonomous mapping algorithm on a desktop robot system. In: Fifth International Conference on Natural Computation (ICNC), vol. 3, pp. 448–453.

Zhang, J., Maeta, S., Bergerman, M., Singh, S., 2014. Mapping orchards for autonomous navigation. In: ASABE and CSBE/SCGAB Annual International Meeting.