Interval analysis for guaranteed and robust nonlinear estimation in robotics

Eric Walter, Luc Jaulin, Michel Kieffer

HAL Id: hal-00845193

https://hal.science/hal-00845193

Submitted on 16 Jul 2013

Interval analysis for guaranteed and robust nonlinear estimation in robotics

Eric Walter [a], Luc Jaulin [c] and Michel Kieffer [a]

[a] *Laboratoire des Signaux et Systèmes, CNRS–SUPELEC–UPS*

91192 Gif-sur-Yvette, France

[c] *LISA, Université d'Angers, 2 Bd Lavoisier, 49045 Angers, France*

Abstract

One of the challenges of interval analysis is to explore and bridge the gap between trivial illustrative examples for which it is not really needed and actual complicated applications for which it is still powerless. Two examples of applications pertaining to this gap are presented in this paper. The first one corresponds to the forward kinematic problem for a Stewart-Gough platform, a benchmark for numerical and symbolical computations. All real solutions are isolated in a guaranteed manner. The second example is relative to the localization and tracking of a vehicle in a partially known environment from distance measurements provided by sonars. The unavoidable presence of outliers is taken into account, which makes the method actually applicable. None of these problems can be solved satisfactorily by the usual local numerical methods based on iterative refinements, and the advantages provided by an approach based on interval analysis are evidenced.

Key words: interval analysis, outliers, robotics, robust estimation, state estimation, Stewart-Gough platform

1 Introduction

Interval analysis (IA) makes it possible to obtain numerical solutions on computers to such basic problems as solving sets of nonlinear equations or inequalities or minimizing nonconvex cost functions. These numerical solutions are provided under the form of sets *guaranteed* to contain *all* actual solutions of the initial mathematical problem. This is a considerable advantage over the usual numerical methods that deliver a point estimate obtained by iterative refinement of some initial guess, without any guarantee of exhaustivity.

Unfortunately, IA suffers from the curse of dimensionality, and many problems of practical interest turn out to be too complex to be handled. One of the challenges of IA is thus to explore and bridge the gap between trivial illustrative examples for which it is not really needed and actual complicated applications for which it is still powerless.

The purpose of this paper is to present two nontrivial but workable examples taken from the field of robotics. Neither of them can be solved satisfactorily by the usual local numerical methods based on iterative refinements. Given the space available, their presentation will be sketchy, but references are provided for more information. The first application, considered in Section 2, is a classical problem of parallel robotics, which has become a benchmark for numerical and symbolical computations. The second application, described in Section 3, is the localization and tracking of a mobile robot from distance measurements provided by sonars.

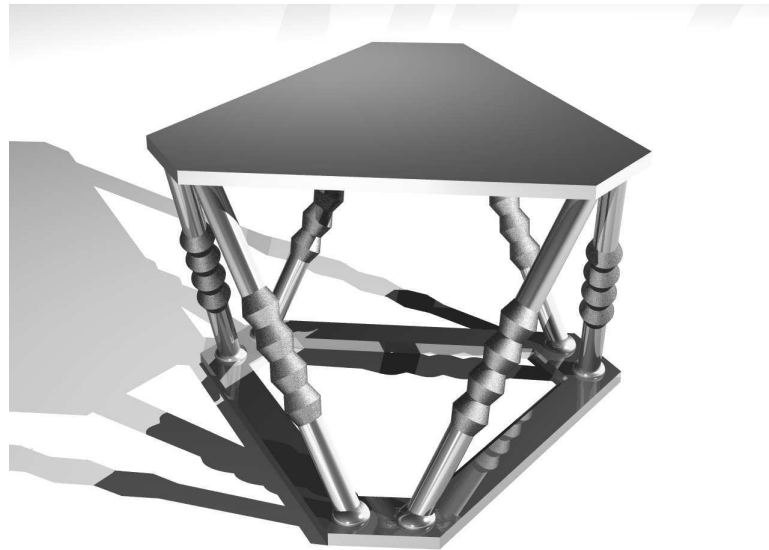## 2 Forward kinematic problem for a Stewart-Gough platform

Fig. 1. Stewart-Gough platform

A Stewart-Gough platform (SGP) consists of a base and a mobile plate, connected by six limbs with variable lengths (Figure 1). By acting on the lengths of these limbs, one can modify the position of the mobile plate relative to the base. This device is an example of a parallel robot, as opposed to an articulated arm where the effectors attached to the articulations act in series. SGPs are used in flight simulators, as well as in many other applications where force and precision are required. What is known as the forward (or direct) kinematic

problem for an SGP is the computation of *all* the possible configurations of the mobile plate relative to the base given

- the positions $\mathbf{a}(i)$ $(i = 1, \ldots, 6)$ of the connections between the limbs and base, defined in a frame $\mathcal{R}_0$ attached to the base by the numbers $a_1^0(i)$, $a_2^0(i)$ and $a_3^0(i)$;
- the positions $\mathbf{b}(i)$ $(i = 1, \ldots, 6)$ of the connections between the limbs and mobile plate, defined in a frame $\mathcal{R}_1$ attached to the mobile plate by the numbers $b_1^1(i)$, $b_2^1(i)$ and $b_3^1(i)$;
- and the lengths y_i of the limbs.

The configuration of the platform is specified by the vector

$$\mathbf{x} = \left(c_1^0, c_2^0, c_3^0, \psi, \theta, \phi\right)^{\mathrm{T}}, \tag{1}$$

where c_1^0, c_2^0 and c_3^0 are the coordinates of the origin of the frame of the mobile plate in $\mathcal{R}_0$, and where ψ, θ and ϕ are the Euler angles of the transformation from $\mathcal{R}_1$ to $\mathcal{R}_0$. The model computing the vector $\mathbf{y}_{\mathrm{m}}$ of the lengths of the limbs as a function of the configuration $\mathbf{x}$ can be written as in Table 1.

The forward kinematic problem can now be formulated as that of computing all $\mathbf{x}$'s such that $\mathbf{y}_{\mathrm{m}}(\mathbf{x}) = \mathbf{y}$, where the numerical value of $\mathbf{y}$ is known. This problem has generated a lot of heat among mathematicians, mechanicians and computer algebraists. It is known that there are at most 40 complex solutions in the most general case, but of course only the real solutions are of interest. Hansen's algorithm for sets of nonlinear equations [1] can be used to solve it [2] and [3]. This involves a guaranteed numerical search in a six-dimensional box of configuration space, which is chosen large enough to enclose all solutions. Among the advantages of the IA approach, one may note that the problem is easily treated on a personal computer even in the most general case where the base and mobile plate are nonplanar, that all the real solutions are obtained (and only them), that the trigonometric functions are handled as such without having to perform an overparametrization to make the equations polynomial and that the numerical results are provided with a reliable evaluation of their accuracy (each configuration vector consistent with the data is isolated in a very small box guaranteed to contain it). Last and not least, IA allows uncertainty in measurements and geometric parameters to be taken into account.

This problem evidences the capability of IA to solve complicated sets of nonlinear equations in an exhaustive and guaranteed manner. IA is already competitive with methods based on computed algebra, over which it has the advantage of providing a guaranteed evaluation of the numerical accuracy of the solutions that it delivers. Much remains to be done, however, to speed up computations

Table 1
Model computing the lengths of the limbs as functions of the configuration vector

`input:` $c_1^0, c_2^0, c_3^0, \psi, \theta, \phi$

$r_{11} := \cos\psi \cos\phi - \sin\psi \cos\theta \sin\phi;$
 $r_{12} := -\cos\psi \sin\phi - \sin\psi \cos\theta \cos\phi;$
 $r_{13} := \sin\psi \sin\theta;$
 $r_{21} := \sin\psi \cos\phi + \cos\psi \cos\theta \sin\phi;$
 $r_{22} := -\sin\psi \sin\phi + \cos\psi \cos\theta \cos\phi;$
 $r_{23} := -\cos\psi \sin\theta;$
 $r_{31} := \sin\theta \sin\phi;$
 $r_{32} := \sin\theta \cos\phi;$
 $r_{33} := \cos\theta;$
 `for` $i := 1$ `to` 6
    $b_1^0(i) := c_1^0 + r_{11} b_1^1(i) + r_{12} b_2^1(i) + r_{13} b_3^1(i);$
    $b_2^0(i) := c_2^0 + r_{21} b_1^1(i) + r_{22} b_2^1(i) + r_{23} b_3^1(i);$
    $b_3^0(i) := c_3^0 + r_{31} b_1^1(i) + r_{32} b_2^1(i) + r_{33} b_3^1(i);$
    $y_\mathrm{m}(i) := \sqrt{(a_1^0(i) - b_1^0(i))^2 + (a_2^0(i) - b_2^0(i))^2 + (a_3^0(i) - b_3^0(i))^2};$
 `end for`

`output:` $y_\mathrm{m}(i), \qquad i = 1, \ldots, 6.$

that take up to a quarter of an hour on present day personal computers. The next section will illustrate the capability of IA to deal with complicated sets of nonlinear inequations.

3   Localization and tracking of a vehicle

The autonomous localization of a vehicle in a partially known environment is a key issue in mobile robotics. The problem considered in this section is the estimation of the position and orientation of a vehicle from distance measurements provided by a belt of on-board sonars. The simpler case where the vehicle is immobile is treated first, before extending the methodology to accommodate motion.

The three-dimensional vector $\mathbf{x}$ to be estimated comprises the position of the vehicle in the room, specified by the Cartesian coordinates x_1 and x_2 of the middle of the axis between the front wheels in the world frame (in meters), and the angle θ of the rotation aligning the axes of a frame attached to the robot with those of the world frame (in radians). The n_{s} sonars of the vehicle deliver a vector $\mathbf{y}$ of n_{s} distances to landmarks of its environment in directions that are specified in the robot frame. To estimate $\mathbf{x}$ from $\mathbf{y}$, one needs a model $\mathbf{y}_{\mathrm{m}}(\mathbf{x})$, describing how the distance measurements are expected to depend on the configuration vector $\mathbf{x}$, and a map of the environment. The map available to the robot consists of a collection of line segments at known positions in the world frame, which represent the landmarks (walls, pillars, pieces of furniture...). Our (admittedly fairly simplistic) measurement model assumes that the waves emitted by the sonars propagate inside cones, and that the distance reported by a given sonar corresponds to that to the closest line segment at least partly located in the emission cone. For any given point configuration $\mathbf{x}$, it is then possible to compute the n_{s} expected distances $\mathbf{y}_{\mathrm{m}}(\mathbf{x})$, which should match the n_{s} actual distances $\mathbf{y}$. Since $\dim \mathbf{x} \ll \dim \mathbf{y}$, the equation $\mathbf{y}_{\mathrm{m}}(\mathbf{x}) = \mathbf{y}$ usually has no solution for $\mathbf{x}$, because of the uncertainty in the measurements and of the approximate nature of the model. It is therefore desirable to find all values of $\mathbf{x}$ that are consistent with the distance measurements given their uncertainty. Based on laboratory measurements, it is possible to characterize the uncertainty of the distance y_i provided by the i-th sonar by using an interval $[y_i]$ instead of a single numerical value. The vector $\mathbf{y}$ is then replaced by an interval vector (or box) $[\mathbf{y}]$, and we are looking for the set $\mathbb{S}$ of *all* configurations that are consistent with the map and distance measurements

$$\mathbb{S} = \big\{ \mathbf{x} \in [\mathbf{x}_0] \mid \mathbf{y}_{\mathrm{m}}(\mathbf{x}) \in [\mathbf{y}] \big\}, \tag{2}$$

where $[\mathbf{x}_0]$ is a search box in configuration space, chosen large enough to be guaranteed to contain all configurations of interest.

The SIVIA algorithm (for set inverter via interval analysis [4], [5]) can be used to partition $[\mathbf{x}_0]$ into three sets of nonoverlapping boxes (subpavings), namely $\underline{\mathbb{S}}$ consisting of those that have been proved to belong to $\mathbb{S}$, $\Delta\mathbb{S}$ consisting of those boxes for which nothing has been proved yet and a set of boxes that have been proved not to belong to $\mathbb{S}$ and can thus be discarded. As a result, $\mathbb{S}$ is bracketed between inner and outer approximations:

$$\underline{\mathbb{S}} \subset \mathbb{S} \subset \overline{\mathbb{S}} = \underline{\mathbb{S}} \cup \Delta\mathbb{S}. \tag{3}$$

Unfortunately, in practice, $\overline{\mathbb{S}}$ often turns out to be empty, which proves that there is no configuration consistent with all measurements. If $\mathbf{x}^*$ is the (unknown) actual configuration of the robot, then $\mathbf{y}_m(\mathbf{x}^*) \notin [\mathbf{y}]$. This is due to the presence in $\mathbf{y}$ of *outliers*, *i.e.*, of distances that do not satisfy either our model or our bounds on the measurement errors or both. There are many reasons for the presence of such outliers, besides the already mentioned simplistic nature of the model. The map may be partly outdated, a sensor may be faulty, people may have intercepted beams with their clothes, or there may be multiple reflections... The point is that unless the presence of outliers is taken into account, the localization procedure remains an academic exercise without potential for application. The strategy that we have elected consists of accepting that q out of the n_s distance measurements may be outliers, and of characterizing the set $\mathbb{S}^q$ of all $\mathbf{x}$ in $[\mathbf{x}_0]$ that are consistent with $n_s - q$ of the distance measurements (see [6], [7]). It is important to understand that this can be done by SIVIA without specifying which of the n_s distance measurements are outliers, so combinatorial explosion is avoided. A possible policy is to start assuming that there is no outlier ($q = 0$) and to increment q until $\overline{\mathbb{S}^q}$ becomes nonempty. As some outliers may go undetected, it is safer to increase q beyond this minimal value, but this increases the size of $\mathbb{S}^q$, so a compromise must be struck between robustness and accuracy of the localization.
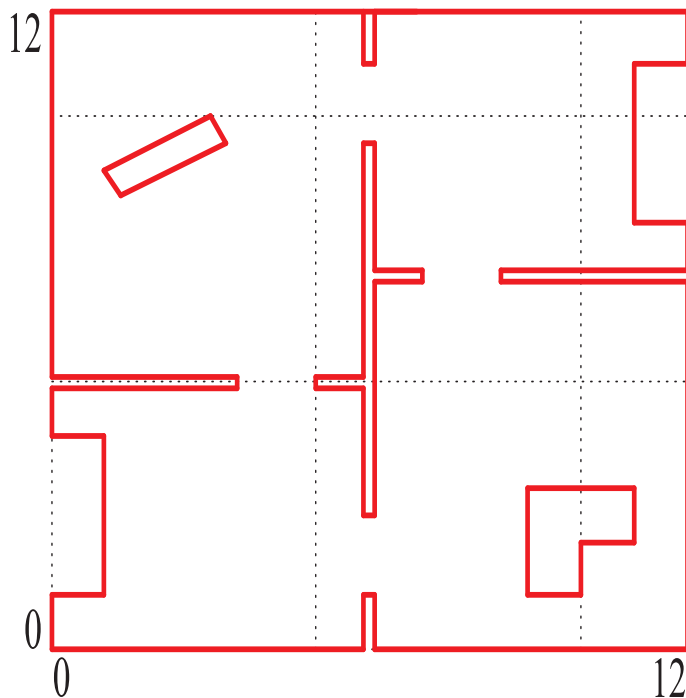

Fig. 2. Map of the environment of the robot

Example 1 *Figure 2 presents the map of the environment in which the $n_s = 24$ sensors of the robot have produced the emission diagram of Figure 3. If there were no outliers, there would be a line segment of the map at least partly between each of the pairs of arcs of circles that materialize the uncertainty in*
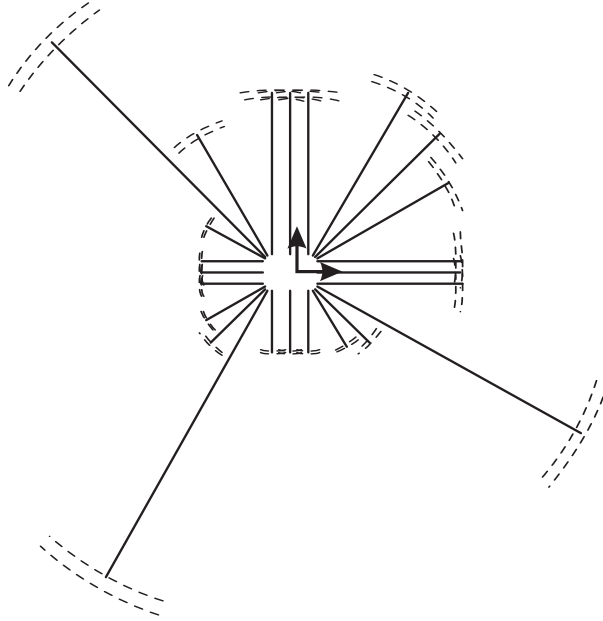
6

Fig. 3. Emission diagram

the distance and direction of measurements for any given sonar. It is necessary to assume that there are at least $q = 3$ outliers to obtain a nonempty set for the estimated configuration. Figure 4 represents $\overline{\mathbb{S}^3}$ as computed by SIVIA and its two-dimensional projections, and Figure 5 shows two configurations that are consistent with all measurements but three. As can be seen, there are two types of radically different solutions, and this is due to a local symmetry in the map. Let us stress that the fact that the solution is not unique is not a defect of the method. One should instead be thankful that the ambiguity in the data has been revealed. ∎
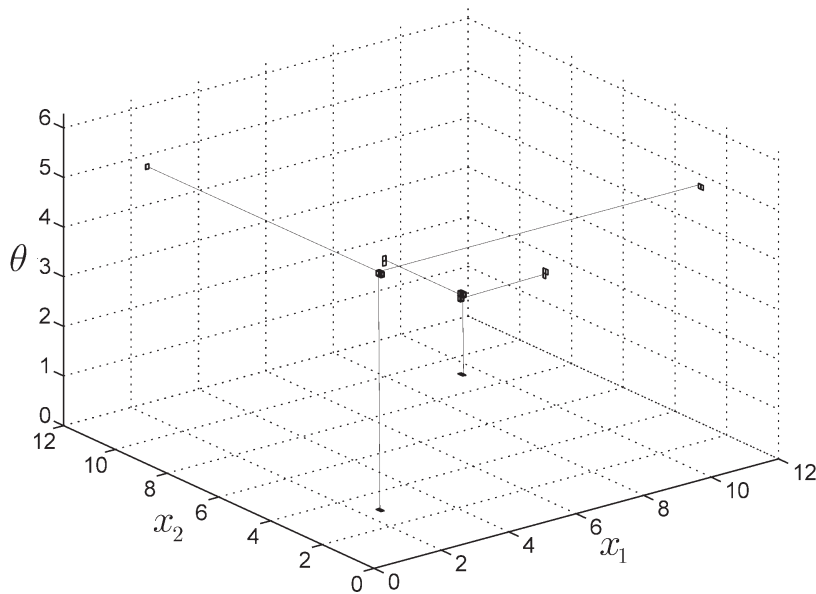

Fig. 4. Solution set in configuration space and its 2D projections
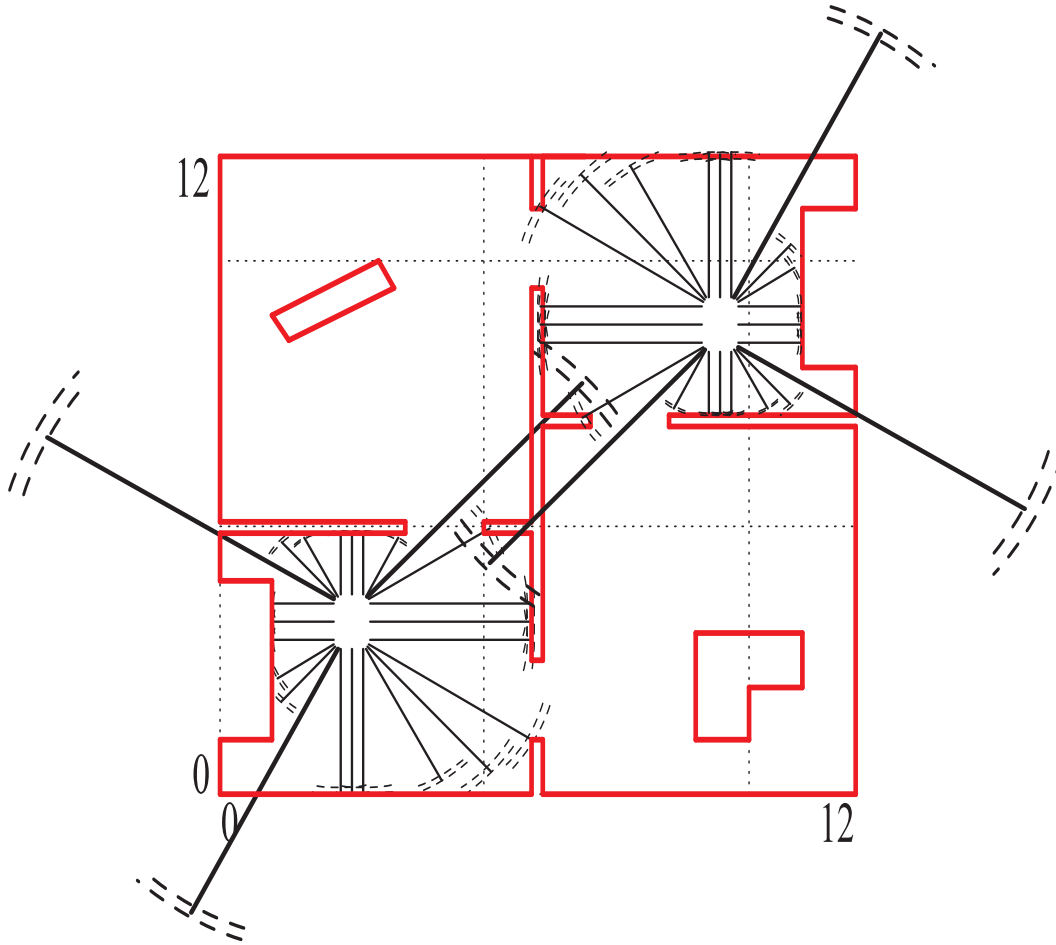
7

Fig. 5. Two possible configurations, outliers are indicated in bold

3.2   Tracking

Assume now that the vehicle is in (slow) motion. By exact discretization of the kinematic equations, a nonlinear discrete-time state-space model can be obtained as

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k), \tag{4}$$

where $\mathbf{x}_k$ is the configuration of the vehicle (now a function of time); $\mathbf{u}_k$ is a known two-dimensional control vector, constant between times k and $k+1$; and $\mathbf{v}_k$ is an unknown state perturbation vector that accounts for the uncertain description of reality by this model. Assume further that a vector $\mathbf{y}_k$ of n_s distance measurements is obtained at time k, which will be modeled by the observation equation

$$\mathbf{y}_k = \mathbf{y}_\mathrm{m}(\mathbf{x}_k) + \mathbf{w}_k, \tag{5}$$

8

where the model output $\mathbf{y}_\mathrm{m}(.)$ is as in the static case and where the vector $\mathbf{w}_k$ is the measurement noise. The problem to be treated is then to estimate $\mathbf{x}_k$ in real time from the information available up to time k, *i.e.*,

$$\mathcal{I}_k = \left\{ [\mathbf{x}_0], \{\mathbf{u}_i, \mathbf{y}_i, [\mathbf{v}_i], [\mathbf{w}_i]\}_{i=0}^k \right\}, \tag{6}$$

where $[\mathbf{v}_i]$ and $[\mathbf{w}_i]$ are known boxes respectively assumed to contain $\mathbf{v}_i$ and $\mathbf{w}_i$.

As in Kalman filtering, the procedure for state estimation alternates a *prediction phase*, during which an outer approximation $\overline{\mathbb{S}_{k+}}$ of the set $\mathbb{S}_{k+}$ of all $\mathbf{x}_{k+1}$ that are consistent with $\mathcal{I}_k$ is built, and a *correction phase*, during which $\mathcal{I}_{k+1}$, which includes the new data vector $\mathbf{y}_{k+1}$, is taken into account to update $\overline{\mathbb{S}_{k+}}$ into $\overline{\mathbb{S}_{k+1}}$. The actual state $\mathbf{x}_{k+1}$ is not changed by this operation, so the correction algorithm boils down to the algorithm for static localization, with $[\mathbf{y}]$ replaced by $[\mathbf{y}_{k+1}] = \mathbf{y}_{k+1} - [\mathbf{w}_{k+1}]$ and $[\mathbf{x}_0]$ replaced by $\overline{\mathbb{S}_{k+}}$. The same strategy can be used for protection against outliers, with q replaced by q_{k+1}. The fact that $\overline{\mathbb{S}_{k+}}$ is usually much smaller than $[\mathbf{x}_0]$ speeds up the process. During prediction, $\overline{\mathbb{S}_{k+}}$ is computed as an outer approximation of the set $\mathbf{f}_k(\overline{\mathbb{S}_k}, \mathbf{u}_k, [\mathbf{v}_k])$ by using an inclusion function associated to $\mathbf{f}_k$. As the pessimism of this inclusion function decreases with the widths of its box arguments, $[\mathbf{v}_k]$ and the boxes of $\overline{\mathbb{S}_k}$ are split into smaller subboxes before computing their contributions to $\overline{\mathbb{S}_{k+}}$. The resulting image boxes overlap, and a final transformation is performed to make $\overline{\mathbb{S}_{k+}}$ a subpaving. The resulting state estimator is a bounded-error nonlinear counterpart to Kalman filtering, which has no equivalent to the best of our knowledge. For more detail, see [8], [9].

Example 2 *Figures 6 and 7 illustrate the tracking of a robot starting from the situation described in Example 1. Their right-hand sides show the projections onto the (x_1, x_2) plane of the solution sets from $k = 0$ to the value indicated. Their left-hand sides show, in continuous lines, the emission diagrams of configurations belonging to the solution set for the value of k indicated. Up to $k = 7$, there are two radically different types of configurations that are consistent with the data. One of them is eliminated by the data collected at $k = 8$, see the emission diagram in dashed lines at the bottom of Figure 6. The presence of outliers does not preclude accurate tracking.* ■

Autonomous robot localization and tracking are well amenable to solution via IA because the number of parameters or state variables to be estimated is small. The results obtained are global, and no configuration compatible with prior information and measurements can be missed. They are extremely robust, and the estimator used can even handle a majority of outliers. The present computing times allow real time implementation for slowly moving vehicles, but there is ample room for improvement of the methodology, for

example, by refining the algorithms, incorporating additional information on the physics of the problem and accommodating other types of sensors.

These problems, as well as other typical robotic problems such as path planing, can thus serve as benchmarks for further studies of the global guaranteed methods for nonlinear analysis provided by IA.

References

[1] E. R. Hansen, Global Optimization using Interval Analysis, Marcel Dekker, New York, 1992.

[2] O. Didrit, Analyse par intervalles pour l'automatique; résolution globale et garantie de problèmes non linéaires en robotique et commande robuste, Thèse de doctorat, Université Paris-Sud, Orsay (juin 1997).

[3] O. Didrit, M. Petitot, E. Walter, Guaranteed solution of direct kinematic problems for general configurations of parallel manipulators, IEEE Trans. on Robotics and Automation 14 (2) (1998) 259–266.

[4] L. Jaulin, E. Walter, Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis, Math. and Comput. in Simulation 35 (1993) 1923–1937.

[5] L. Jaulin, E. Walter, Set inversion via interval analysis for nonlinear bounded-error estimation, Automatica 29 (4) (1993) 1053–1064.

[6] M. Kieffer, L. Jaulin, E. Walter, D. Meizel, Robust autonomous robot localization using interval analysis, Reliable Computing 6 (3) (2000) 337–362.

[7] M. Kieffer, L. Jaulin, E. Walter, D. Meizel, Nonlinear identification based on unreliable priors and data, with application to robot localization, in: A. Garulli, A. Tesi, A. Vicino (Eds.), Robustness in Identification and Control, Springer, London, 1999, pp. 190–203, LNCIS 245.

[8] M. Kieffer, Estimation ensembliste par analyse par intervalles, application à la localisation d'un véhicule, PhD dissertation, Université Paris-Sud, Orsay (1999).

[9] M. Kieffer, L. Jaulin, E. Walter, Guaranteed recursive nonlinear state estimation using interval analysis, in: Proc. 37th IEEE Conference on Decision and Control, Tampa, December 16-18, 1998, pp. 3966–3971.
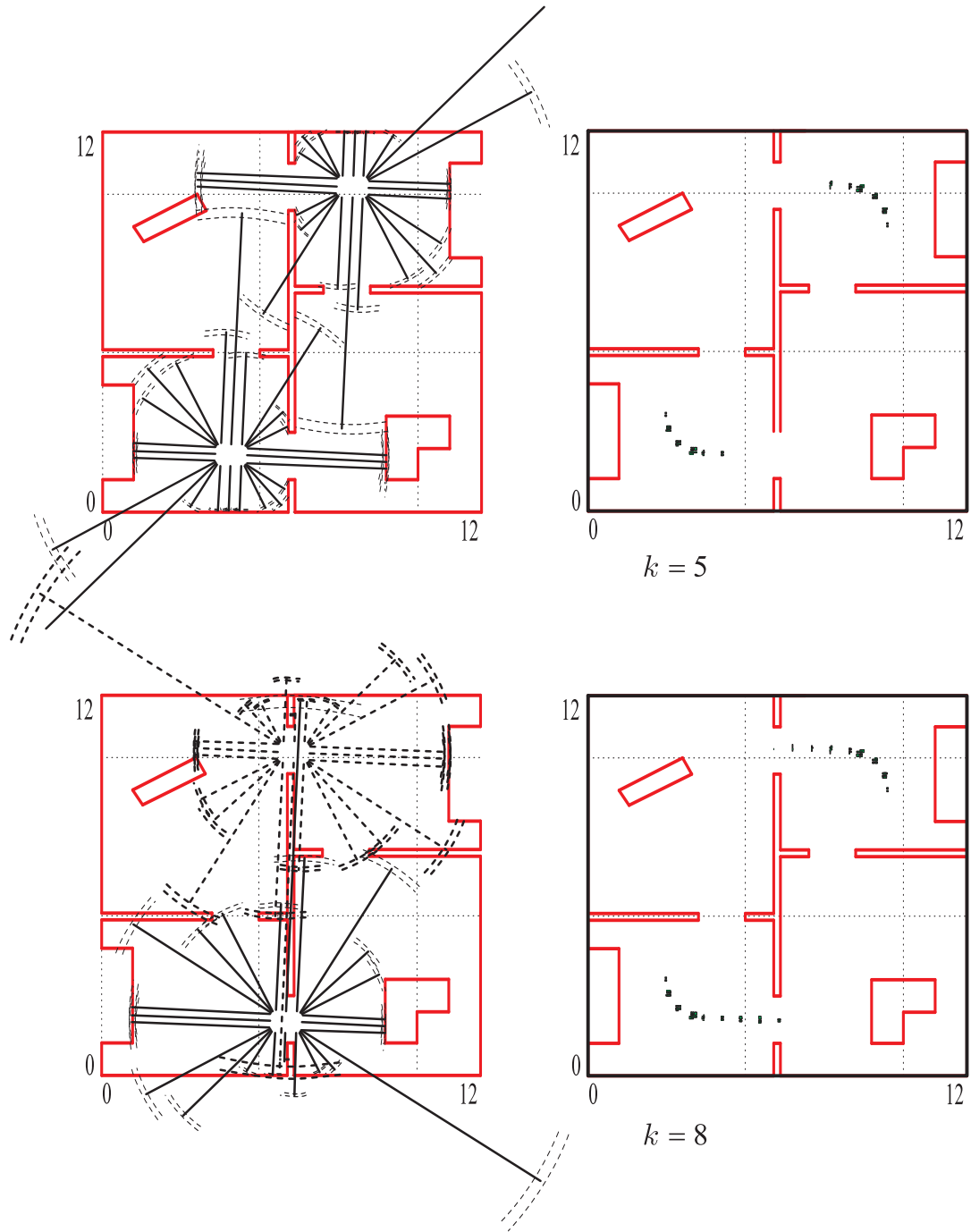
Fig. 6. Tracking - Part I; top: one of the measurements is so widely off the mark that the emission diagram is not completely represented; bottom: the emission diagram in dashed lines corresponds to a configuration that is no longer consistent with the data
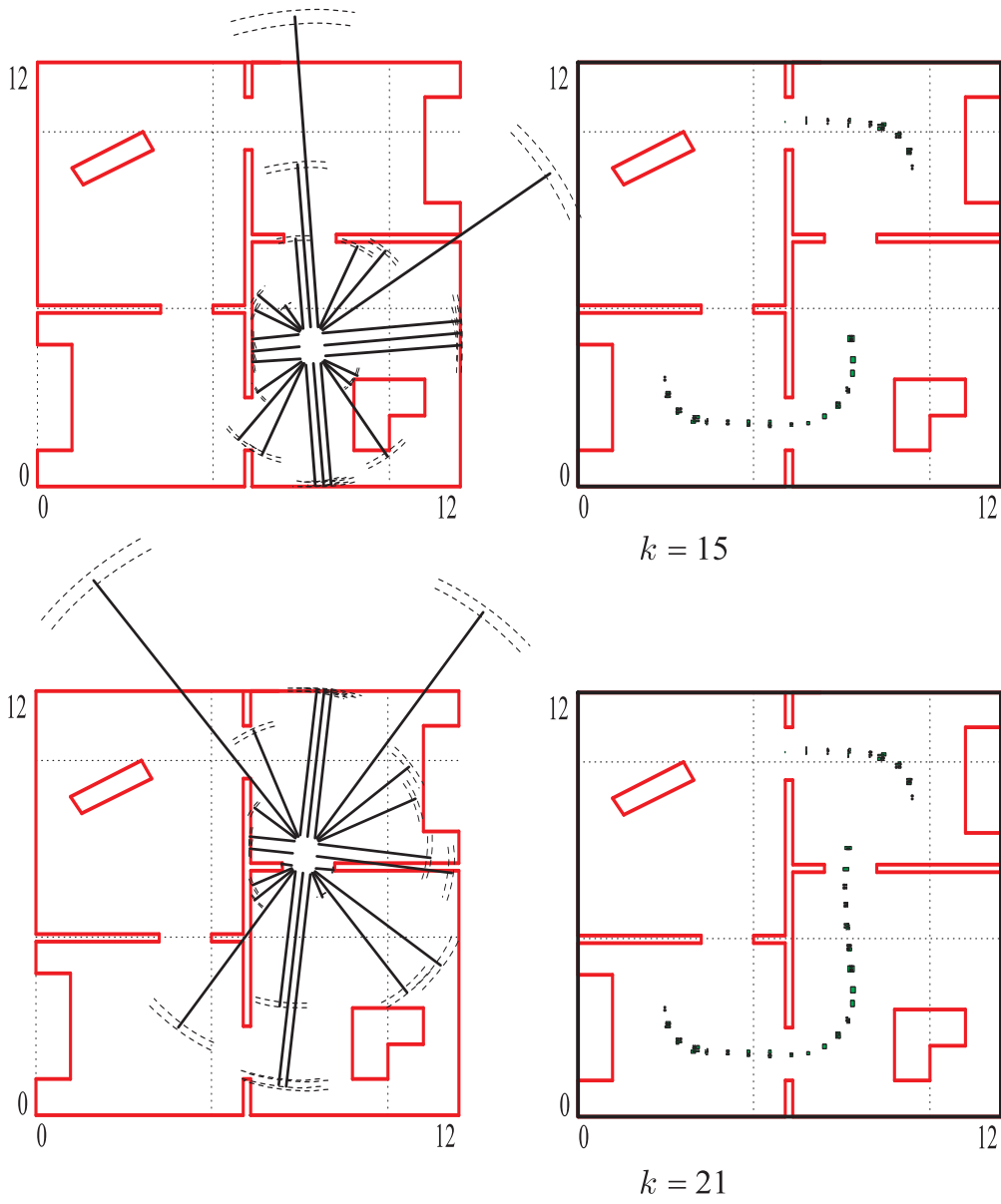
11

Fig. 7. Tracking - Part II