

An Application of Constraint-Based Task Specification and Estimation for Sensor-Based Robot Systems

Tinne De Laet, Wilm Decré, Johan Rutgeerts, Herman Bruyninckx and Joris De Schutter

Abstract—This paper shows the application of a systematic approach for constraint-based task specification for sensor-based robot systems [1] to a laser tracing example. This approach integrates both task specification and estimation of geometric uncertainty in a unified framework. The framework consists of an *application independent control and estimation scheme*. An automatic derivation of controller and estimator equations is achieved, based on a geometric task model that is obtained using a *systematic task modeling procedure*. The paper details the systematic modeling procedure for the laser tracing task and elaborates on the task specific choice of two types of task coordinates: feature coordinates, defined with respect to object and feature frames, which facilitate the task specification, and uncertainty coordinates to model geometric uncertainty. Furthermore, the control and estimation scheme for this specific task is studied. Simulation and real world experimental results are presented for the laser tracing example.

I. INTRODUCTION

Robotic tasks of limited complexity, such as simple positioning tasks, trajectory following or pick-and-place applications in well structured environments, are straightforward to program. For these kinds of tasks extensive programming support is available, as the specification primitives for these tasks are present in current commercial robot control software. While these robot capabilities already fulfill some industrial needs, research focuses on specification and execution of much more complex tasks. The goal of our recent research is to open up new robot applications in industrial as well as domestic and service environments. Examples of complex tasks include sensor-based navigation and 3D manipulation in partially or completely unknown environments, using redundant robotic systems such as mobile manipulator arms, cooperating robots, robotic hands or humanoid robots, and using multiple sensors such as vision, force, torque, tactile and distance sensors. Little programming support is available for these kinds of tasks. As a result, the task programmer has to rely on extensive knowledge in multiple fields such as spatial kinematics, 3D modeling of objects, geometric uncertainty and sensor systems, dynamics and

All authors are with the Department of Mechanical Engineering, Katholieke Universiteit Leuven, Belgium.

Corresponding author: Tinne De Laet (tinne.delaet@mech.kuleuven.be)

All authors gratefully acknowledge the financial support by K.U.Leuven's Concerted Research Action GOA/05/10

Tinne De Laet is a Research Assistant of the Research Foundation - Flanders (FWO-Vlaanderen). Wilm Decré's research is funded by a Ph.D. grant of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT- Vlaanderen).

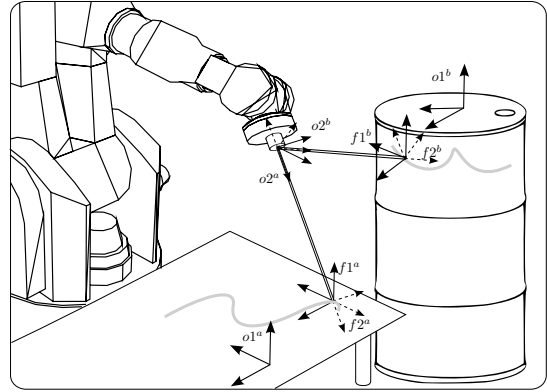


Fig. 1. The object and feature frames for simultaneous laser tracing on a plane and a barrel.

control, estimation, as well as resolution of redundancy and of conflicting constraints.

The goal of our recent research is to fill this gap. We want to develop programming support for the implementation of complex, sensor-based robotic tasks in the presence of geometric uncertainty. The foundation for this programming support is a generic and systematic approach [1] to specify and control a task while dealing properly with geometric uncertainty.

Previous work on specification of sensor-based robot tasks, such as force controlled manipulation [2]–[5] or force controlled compliant motion combined with visual servoing [6], was based on the concept of the *compliance frame* [7] or *task frame* [8]. In this frame, different control modes, such as trajectory following, force control, visual servoing or distance control, are assigned to each of the translational directions along the frame axes and to each of the rotational directions about the frame axes. The task frame concept has proved to be very useful for the specification of a variety of practical robot tasks. However, the drawback of the task frame approach is that it only applies to task geometries with limited complexity, that is, task geometries for which separate control modes can be assigned independently to three pure translational and three pure rotational directions along the axes of a *single* frame.

A more systematic approach is to assign control modes and corresponding constraints to *arbitrary* directions in the six dimensional manipulation space. This approach, known as *constraint-based programming*, opens up new applications involving a much more complex geometry and/or involving multiple sensors that control different directions in space

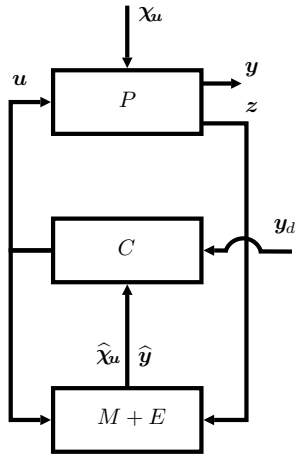


Fig. 2. General control and estimation scheme including plant P , controller C , and model update and estimation block $M+E$.

simultaneously.

Seminal theoretical work on constraint-based programming of robot tasks was done by Ambler and Popplestone [9] and by Samson and coworkers [10]. Our own preliminary work on a task specification framework was presented in [11], while the mature framework, of which this paper shows an application, is thoroughly discussed in [1].

II. APPLICATION

This paper shows the application of a systematic task specification framework [1] to a geometrically complex task involving an underconstrained specification as well as estimation of uncertain geometric parameters. The goal is to trace simultaneously a path on a plane as well as on a cylindrical barrel with known radius R using two lasers which are rigidly attached to the robot end effector as shown in Figure 1. The tracing of a figure on an object only requires 2 degrees of freedom (dof), since only the position of the intersection of the beam with the object surface has to be specified (for instance the x and y position of the laser dot on the plane). Since the robot has 6 dof and only 4 dof are required for tracing the two figures, the task is underconstrained. The lasers attached to the robot end-effector are actually laser distance sensors. These lasers measure the distance from the robot end-effector to the object surfaces. The position and orientation of the plane are unknown. While the barrel is known to be in vertical position, the exact position of the barrel is unknown. The positions of the plane and the barrel cannot be estimated from a single distance measurement, as a single measurement does not provide enough information about these positions. However, a consecutive set of measurements does contain all necessary information. This paper shows how the positions of the plane and the barrel are recursively estimated from the set of distance measurements.

III. CONTROL AND ESTIMATION SCHEME

Figure 2 shows the general control and estimation scheme presented in [1] and used throughout this paper. This scheme

includes the Plant P , the Controller C , and the Model Update and Estimation block $M+E$. The Plant P represents both the robot system (where q represents the *internal state*) and the environment.

The *control input* to the plant is u , in the case of a velocity-based control scheme, this input corresponds to the set of desired joint velocities. The *system output* is y , which represents the controlled variables. Task specification consists of imposing constraints to the system output y . These constraints take the form of desired values y_d . The plant is observed through *measurements* z . Not all system outputs are directly measured, and an estimator is needed to generate estimates \hat{y} . These estimates are needed in the control law C .

In general the plant is disturbed by various disturbance inputs. Here we focus on *geometric disturbances*, represented by coordinates χ_u . These coordinates represent modeling errors, uncontrolled degrees of freedom in the robot system or geometric disturbances in the robot environment. As with the system outputs, not all these disturbances can be measured directly, but they can be estimated by including a disturbance observer in the estimator block $M+E$.

IV. TASK MODELING

This section presents the application of a *systematic task modeling* procedure to the laser tracing task. Additional task coordinates, denoted as *feature coordinates* χ_f , are introduced to facilitate the modeling of constraints and measurements by the user. Furthermore, *uncertainty coordinates* χ_u are introduced which represent modeling errors, uncontrolled degrees of freedom in the robot system or geometric disturbances in the robot environment. These two types of coordinates are defined in *object frames* and *feature frames* that are chosen by the task programmer in a way that simplifies the specification of the task at hand. This section presents the systematic 4-step procedure proposed in [1] for the assignment of these frames and task coordinates applied to the laser tracing task.

A. Object and feature frames

A typical robot task accomplishes a relative motion between objects. A systematic procedure is used to introduce a set of reference frames in which these relative motions are easily expressed.

The first frame is the “world” reference frame, denoted by w . In this application the world frame is placed at the base frame of the robot. The other frames are *object and feature frames* that are relevant for the task. In the framework an *object* can be any rigid object in the robot system (for example a robot end effector or a robot link) or in the robot environment. In this example four objects are relevant: the two lasers, the plane and the barrel. A *feature* is linked to an *object*, and indicates a *physical entity* on that object (such as a vertex, edge, face, surface), or an *abstract geometric property* of a physical entity (such as the symmetry axis of a hollow cylinder, or the reference frame of a sensor connected to the object, such as a camera).

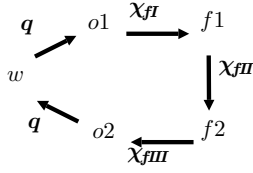


Fig. 3. Object and feature frames and feature coordinates.

The relative motion between two objects is *specified by imposing constraints* on the feasible relative motion between one feature on the first object and a corresponding feature on the second object. Each such constraint needs four frames: *two object frames* (called $o1$ and $o2$, each attached to one of the objects), and *two feature frames* (called $f1$ and $f2$, each attached to one of the corresponding features of the objects). For an application in 3D space, there are in general six degrees of freedom between $o1$ and $o2$. The connection $o1 \rightarrow f1 \rightarrow f2 \rightarrow o2$ forms a *kinematic chain*, that is, the degrees of freedom between $o1$ and $o2$ are distributed over three submotions: the relative motion of $f1$ with respect to $o1$ (I), the relative motion of $f2$ with respect to $f1$ (II), and the relative motion of $o2$ with respect to $f2$ (III), as shown in Figure 3.

In this application relevant features are the intersection points between the first laser and the plane and between the second laser and the barrel since these points have to trace a specified path.

Furthermore, two kinematic chains are recognized, one for the laser-plane combination (a) and a second for the laser-barrel combination (b).

In the framework four rules are given for choosing the features and for assigning object and feature frames.

Figure 1 shows the object and feature frames for the laser tracing example chosen according to these rules. For the laser-plane feature:

- frame $o1^a$ fixed to the plane and with the z -axis perpendicular to the plane,
- frame $o2^a$ fixed to the first laser on the robot end effector and with its z -axis along the laser beam,
- frame $f1^a$ has the same orientation as $o1^a$, but is located at the intersection of the laser with the plane,
- frame $f2^a$ has the same position as $f1^a$, but the same orientation as $o2^a$,

and for the laser-barrel feature:

- frame $o1^b$ fixed to the barrel and with its origin on and the x -axis along the axis of the barrel,
- frame $o2^b$ fixed to the second laser on the robot end effector and with its z -axis along the laser beam,
- frame $f1^b$ located at the intersection of the laser with the barrel, z -axis perpendicular to the barrel surface and x -axis parallel to the barrel axis,
- frame $f2^b$ with the same position as $f1^b$ and the same orientation as $o2^b$.

B. Feature coordinates

Feature coordinates χ_f are introduced to facilitate the task specification by the user. These coordinates represent the submotions between $o1$ and $o2$. This subsection shows that the choice of the object and feature frames simplifies the mathematical representation of the submotions.

For the laser-plane combination the feature coordinates expressing the submotions are:

$$\chi_{fI}^a = (x^a \ y^a)^T, \quad (1)$$

$$\chi_{fII}^a = (\phi^a \ \theta^a \ \psi^a)^T, \quad (2)$$

$$\chi_{fIII}^a = (z^a), \quad (3)$$

where x^a and y^a are expressed in $o1^a$ and represent the position of the laser dot on the plane, while z^a is expressed in $o2^a$ and represents the distance of the robot to the plane along the laser beam. ϕ^a, θ^a, ψ^a represent Euler angles between $f1^a$ and $f2^a$ and are expressed in $f2^a$.

For the laser-barrel combination feature coordinates expressing the submotions are:

$$\chi_{fI}^b = (x^b \ \alpha^b)^T, \quad (4)$$

$$\chi_{fII}^b = (\phi^b \ \theta^b \ \psi^b)^T, \quad (5)$$

$$\chi_{fIII}^b = (z^b), \quad (6)$$

where x^b and α^b are cylindrical coordinates expressed in $o1^b$ representing the position of the laser dot on the barrel, while z^b is expressed in $o2^b$ and represents the distance of the robot to the barrel along the laser beam. ϕ^b, θ^b, ψ^b represent Euler angles between $f1^b$ and $f2^b$ and are expressed in $f2^b$.

All feature coordinates are grouped into a single vector χ_f :

$$\chi_f = (\chi_{fI}^a \ \chi_{fII}^a \ \chi_{fIII}^a \ \chi_{fI}^b \ \chi_{fII}^b \ \chi_{fIII}^b)^T. \quad (7)$$

C. Uncertainty coordinates

Uncertainty coordinates χ_u are introduced to represent modeling errors, uncontrolled degrees of freedom in the robot system or geometric disturbances in the robot environment. In the laser tracing example uncertainty coordinates are needed to represent the unknown position and orientation of the plane and the position of the barrel. The unknown position and orientation of the plane is modeled by:

$$\chi_{uI}^a = (h^a \ \alpha^a \ \beta^a)^T, \quad (8)$$

with h^a the z -position of a reference point on the plane with respect to the world, and α^a and β^a the two Euler angles which determine the orientation of the plane with respect to the world¹. The unknown position of the barrel is modeled by:

$$\chi_{uI}^b = (x_u^b \ y_u^b)^T, \quad (9)$$

with x_u^b and y_u^b the x - and y -position of the barrel with respect to the world.

All introduced uncertainty coordinates are grouped into a single vector χ_u .

¹Since the plane is considered infinite, the rotation around the normal of the plane is irrelevant.

D. Task specification

To specify the desired task, constraints have to be specified on the set of outputs. Constraints are specified as:

$$y_i(t) = y_{di}(t), \quad (10)$$

where $y_i(t)$ represents a system output, and $y_{di}(t)$ the imposed constraint².

To generate the desired path on the plane, constraints have to be specified on the following outputs, in this case equal to some of the introduced feature coordinates:

$$y_1 = x^a \quad \text{and} \quad y_2 = y^a, \quad (11)$$

while for the path on the barrel constraints have to be specified on:

$$y_3 = x^b \quad \text{and} \quad y_4 = \alpha^b. \quad (12)$$

The measurement equations for the lasers measuring the distance to the plane and the barrel are easily expressed using the feature coordinates of features a and b :

$$z_1 = z^a \quad \text{and} \quad z_2 = z^b. \quad (13)$$

From the above it becomes clear that the task specification equations (11-13) become very simple due to the introduction of the task coordinates.

V. CONTROL BLOCK

This section shows how to derive the velocity based control equation based on the obtained task modeling³. The first part of this section details the application independent derivation of the control equation for the case of a velocity based control scheme as presented in [1]. The second part applies the approach to the laser tracing example.

The plant is assumed to be an ideal velocity controlled system, that is, the robot dynamics are neglected. Hence, the *system equation* is given by:

$$\dot{\mathbf{q}} = \mathbf{u} = \dot{\mathbf{q}}_d, \quad (14)$$

where the control input \mathbf{u} corresponds to the desired joint velocities $\dot{\mathbf{q}}_d$.

On the other hand, the *output equation* relates the position based system state to the outputs \mathbf{y} :

$$\mathbf{f}(\mathbf{q}, \chi_f) = \mathbf{y}, \quad (15)$$

The system state, consisting of \mathbf{q} and χ_f is nonminimal, because a dependency relation exists between \mathbf{q} and χ_f . This dependency relation corresponds to the loop closure equations which are perturbed by the uncertainty coordinates χ_u , and is expressed as:

$$\mathbf{l}(\mathbf{q}, \chi_f, \chi_u) = \mathbf{0}. \quad (16)$$

In the laser tracing example two such independent loop closures exist, one for the laser-plane and one for the laser-barrel feature.

²Further on we omit time dependency in the notation.

³Other control schemes are discussed in detail in [12].

For the derivation of velocity based control the output (15) and the loop closure (16) equations are differentiated with respect to time to obtain equations at velocity level. The output equation at velocity level is written as:

$$\mathbf{C}_q \dot{\mathbf{q}} + \mathbf{C}_f \dot{\chi}_f = \dot{\mathbf{y}}, \quad (17)$$

with $\mathbf{C}_q = \frac{\partial \mathbf{f}}{\partial \mathbf{q}}$ and $\mathbf{C}_f = \frac{\partial \mathbf{f}}{\partial \chi_f}$.

On the other hand, velocity loop constraint becomes:

$$\mathbf{J}_q \dot{\mathbf{q}} + \mathbf{J}_f \dot{\chi}_f + \mathbf{J}_u \dot{\chi}_u = \mathbf{0}, \quad (18)$$

with $\mathbf{J}_q = \frac{\partial \mathbf{l}}{\partial \mathbf{q}}$, $\mathbf{J}_f = \frac{\partial \mathbf{l}}{\partial \chi_f}$ and $\mathbf{J}_u = \frac{\partial \mathbf{l}}{\partial \chi_u}$. \mathbf{J}_q is known since it represents the robot Jacobian. Furthermore, thanks to the introduction of the feature coordinates and uncertainty coordinates the above Jacobians \mathbf{J}_f and \mathbf{J}_u can be easily constructed as shown for this example in the next paragraphs. $\dot{\chi}_f$ is solved from (18), yielding:

$$\dot{\chi}_f = -\mathbf{J}_f^{-1} (\mathbf{J}_q \dot{\mathbf{q}} + \mathbf{J}_u \dot{\chi}_u). \quad (19)$$

Substituting (19) into (17) yields the modified output equation:

$$\mathbf{A} \dot{\mathbf{q}} = \dot{\mathbf{y}} + \mathbf{B} \dot{\chi}_u, \quad (20)$$

where $\mathbf{A} = \mathbf{C}_q - \mathbf{C}_f \mathbf{J}_f^{-1} \mathbf{J}_q$ and $\mathbf{B} = \mathbf{C}_f \mathbf{J}_f^{-1} \mathbf{J}_u$ are introduced for simplicity of notation.

Constraint equation (10) is also expressed at velocity level. As a result, the constraint equation has to include feedback at position level to compensate for drift, modeling errors and disturbances:

$$\dot{\mathbf{y}} = \dot{\mathbf{y}}_d + \mathbf{K}_p (\mathbf{y}_d - \mathbf{y}) = \dot{\mathbf{y}}_d^\circ, \quad (21)$$

with \mathbf{K}_p a matrix of feedback constants (in this case a simple diagonal matrix) and $\dot{\mathbf{y}}_d^\circ$ the modified constraint at velocity level. Since the outputs \mathbf{y} cannot be measured directly, they are replaced in (21) by their estimates $\hat{\mathbf{y}}$ provided by the estimator (as explained in Section VI).

Applying constraint (21) to (20), while also substituting system equation (14) and replacing $\dot{\chi}_u$ by its estimate $\hat{\chi}_u$ results in: $\mathbf{A} \dot{\mathbf{q}}_d = \dot{\mathbf{y}}_d^\circ + \mathbf{B} \hat{\chi}_u$. Solving for the control input $\dot{\mathbf{q}}_d$ yields:

$$\dot{\mathbf{q}}_d = \mathbf{A}_W^\# \left(\dot{\mathbf{y}}_d^\circ + \mathbf{B} \hat{\chi}_u \right), \quad (22)$$

where $\mathbf{A}_W^\#$ denotes the weighted pseudoinverse [13], [14] with weighting matrix \mathbf{W} . Furthermore, since the plane and the barrel are not moving $\hat{\chi}_u = \mathbf{0}$ in this example. Therefore the control input reduces to:

$$\dot{\mathbf{q}}_d = \mathbf{A}_W^\# \dot{\mathbf{y}}_d^\circ. \quad (23)$$

Obtaining the constraint matrices \mathbf{C}_q and \mathbf{C}_f : In the laser tracing example the matrices \mathbf{C}_q and \mathbf{C}_f are easily found by inspection. Since no constraints exist directly on the joint level, $\mathbf{C}_q = \mathbf{0}$. Since each constraint is directly expressed on one feature coordinate, \mathbf{C}_f becomes a simple selection matrix which selects the appropriate components of χ_f (7):

$$\mathbf{C}_f = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \mathbf{0}_{4 \times 4} & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (24)$$

Obtaining the feature Jacobian \mathbf{J}_f : The feature Jacobian is composed using the Jacobians of the submotions. For instance for the laser-plane combination: ${}_{ref}\mathbf{J}_f^a = (\mathbf{S}_{ref}^I \mathbf{J}_{fI}^a \mid \mathbf{S}_{ref}^{II} \mathbf{J}_{fII}^a \mid \mathbf{S}_{ref}^{III} \mathbf{J}_{fIII}^a)$, where the screw transformation matrix \mathbf{S}_{ref}^x transforms the Jacobians of the submotions $x = I, II, III$ to a common reference frame ref . A similar expression for ${}_{ref}\mathbf{J}_f^b$ holds for the laser-barrel combination. The above two feature Jacobians are finally grouped into one Jacobian:

$${}_{ref}\mathbf{J}_f = \begin{pmatrix} {}_{ref}\mathbf{J}_f^a & \mathbf{0}_{6 \times 6} \\ \mathbf{0}_{6 \times 6} & {}_{ref}\mathbf{J}_f^b \end{pmatrix}. \quad (25)$$

The Jacobians of the submotions are easily found by inspection using the notion of twists. A twist is a six-vector containing a translational and a rotation velocity: $\mathbf{t} = (v_x \ v_y \ v_z \ \omega_x \ \omega_y \ \omega_z)$. A twist represents an instantaneous motion of one frame relative to another; these frames are indicated by a right super- and subscript respectively. The twist has a given reference frame indicated with left subscript. For the laser-plane combination:

$$\begin{cases} {}_{o1a}\mathbf{t}_{o1a}^{f1a} = {}_{o1a}\mathbf{J}_{fI}^a \dot{\chi}_{fI}^a = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \dot{\chi}_{fI}^a, \\ {}_{f1a}\mathbf{t}_{f1a}^{f2a} = {}_{f1a}\mathbf{J}_{fII}^a \dot{\chi}_{fII}^a = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \mathbf{E}_3 & \mathbf{0} & \mathbf{0} \end{pmatrix} \dot{\chi}_{fII}^a, \\ {}_{o2a}\mathbf{t}_{o2a}^{f2a} = {}_{o2a}\mathbf{J}_{fIII}^a \dot{\chi}_{fIII}^a = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \dot{\chi}_{fIII}^a, \end{cases} \quad (26)$$

where \mathbf{E}_3 is a 3×3 matrix used to convert Euler angle rates to angular velocities. For the laser-barrel combination:

$$\begin{cases} {}_{o1b}\mathbf{t}_{o1b}^{f1b} = {}_{o1b}\mathbf{J}_{fI}^b \dot{\chi}_{fI}^b = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \dot{\chi}_{fI}^b, \\ {}_{f1b}\mathbf{t}_{f1b}^{f2b} = {}_{f1b}\mathbf{J}_{fII}^b \dot{\chi}_{fII}^b = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \mathbf{E}_3 & \mathbf{0} & \mathbf{0} \end{pmatrix} \dot{\chi}_{fII}^b, \\ {}_{o2b}\mathbf{t}_{o2b}^{f2b} = {}_{o2b}\mathbf{J}_{fIII}^b \dot{\chi}_{fIII}^b = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \dot{\chi}_{fIII}^b, \end{cases} \quad (27)$$

Obtaining the uncertainty Jacobian \mathbf{J}_u : Similarly, the Jacobian \mathbf{J}_u can be constructed using the subJacobians for the uncertainties: ${}_{ref}\mathbf{J}_u = (\mathbf{S}_{ref}^{Ia} \mathbf{J}_{uI}^a \mid \mathbf{S}_{ref}^{Ib} \mathbf{J}_{uI}^b)$. For the laser-plane combination the Jacobian is derived using:

$${}_{w}\mathbf{t}_w^{o1a} = {}_{w}\mathbf{J}_{uI}^a \dot{\chi}_{uI}^a = \begin{pmatrix} 0 & \mathbf{0}_{3 \times 2} \\ 1 & \mathbf{0}_{3 \times 2} \\ \mathbf{0}_{3 \times 1} & \mathbf{E}_{3 \times 2} \end{pmatrix} \dot{\chi}_{uI}^a, \quad (28)$$

where $\mathbf{E}_{3 \times 2}$ is a 3×2 matrix used to convert Euler angle rates $\dot{\alpha}^a$ and $\dot{\beta}^a$ to angular velocities. Similarly for the laser-barrel combination:

$${}_{w}\mathbf{t}_w^{o1b} = {}_{w}\mathbf{J}_{uI}^b \dot{\chi}_{uI}^b = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \dot{\chi}_{uI}^b. \quad (29)$$

VI. MODEL UPDATE AND ESTIMATION BLOCK

The goal of model update and estimation is threefold: (i) to provide an estimate for the system outputs \mathbf{y} to be used in the feedback terms of constraint equations (21), (ii) to provide an estimate for the uncertainty coordinates χ_u and their derivatives, to be used in the control input (22), and (iii) to maintain the consistency between the joint and feature coordinates \mathbf{q} and χ_f based on the loop constraints. Model update and estimation makes use of a prediction correction procedure [1] and is based on an extended system model and on measurement equations. The extended system model for this example is given by⁴:

$$\frac{d}{dt} \begin{pmatrix} \mathbf{q} \\ \chi_f \\ \chi_u \end{pmatrix} = \mathbf{O} \begin{pmatrix} \mathbf{q} \\ \chi_f \\ \chi_u \end{pmatrix} + \begin{pmatrix} \mathbf{I} \\ -\mathbf{J}^{-1} \mathbf{I} \\ \mathbf{0} \end{pmatrix} \dot{\mathbf{q}}_d. \quad (30)$$

This model consists of three parts: the first line corresponds to the system model (14), the second line corresponds to the velocity loop constraint (19), and the last line corresponds to the model used for the uncertainty coordinates. Since in this case the uncertainty coordinates which are estimated are constant (plane position and orientation and barrel position) the model for the uncertainty coordinates is very simple: $\chi_u = C^{te}$ or $\frac{d}{dt} \chi_u = \mathbf{0}$. In the case of the laser tracing example each laser distance measurement fits to one feature coordinate. Therefore the measurement equations are given by a selection matrix:

$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} \mathbf{0}_{2 \times 5} & \begin{vmatrix} 1 \\ 0 \end{vmatrix} & \mathbf{0}_{2 \times 5} & \begin{vmatrix} 0 \\ 1 \end{vmatrix} \end{pmatrix} \chi_f. \quad (31)$$

Given the system model (30) and the measurement model (31) different estimation techniques can be used to obtain optimal estimates of the geometric uncertainties. In this paper extended Kalman filtering is used.

VII. RESULTS

A simulation experiment is carried out in which a desired path is traced on a plane and on a barrel with radius $0.4m$. The desired path on the plane is a circle with radius $0.4m$. The desired path on the barrel's surface is a circle with radius $0.1m$. The angular frequency with which the circles are traced is $45 \frac{rad}{s}$ for both the plane and the barrel. Results of this simulation are shown in Figure 4 (plane) and Figure 5 (barrel). The initial estimate of the plane differs from the real one by $0.4m$ for the z -position of the reference point of the plane, 20° for Euler angle α^a and 30° for Euler angle α^b . The initial estimate of the barrel differs from the real one by $0.4m$ in the x -direction and $0.1m$ in the y -direction. An extended Kalman filter is used for the estimation in which extra process uncertainty is applied (by multiplying the covariance with a fading factor 1.1 for the plane and 1.12 for the barrel, as explained in [15]). As soon as the locations of the plane and the barrel are estimated correctly, after approximately one circle (9s), the circles traced by the laser beams equal the desired ones. Remark that due to the

⁴In general the extended system model is more involved as is explained in [1].

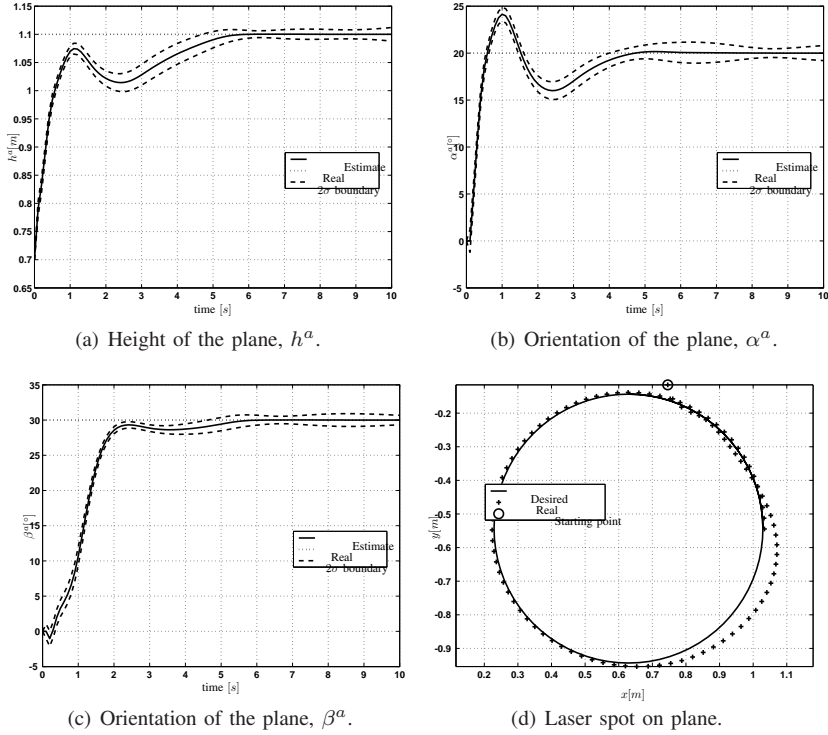


Fig. 4. Laser tracing on plane: estimation of plane height (h^a) and plane orientation (α^a and β^a) (simulation).

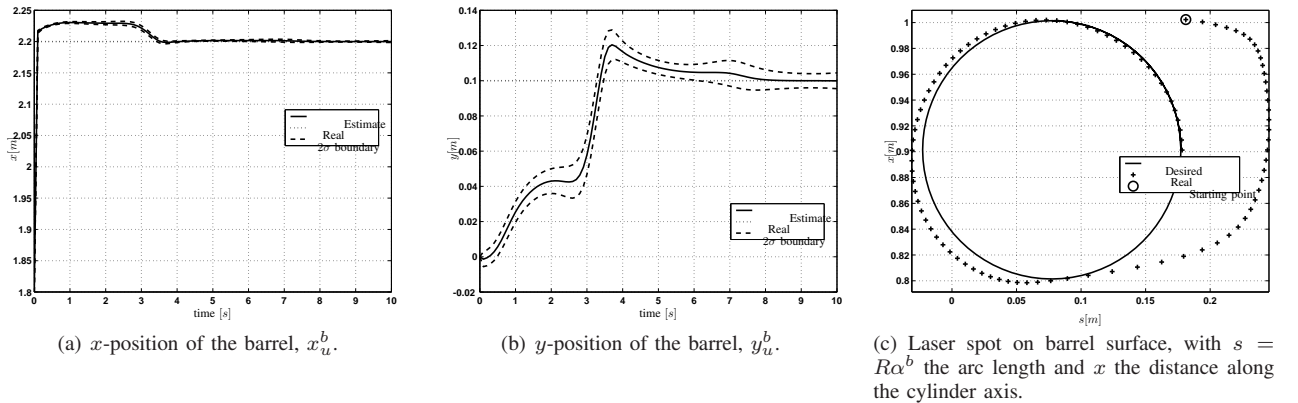


Fig. 5. Laser tracing on barrel: estimation of x - and y -position of the barrel (x_u^b and y_u^b) (simulation).

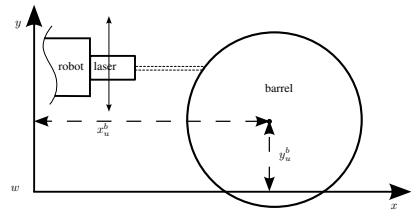
inability of the extended Kalman filter to obtain informative covariance boundaries in the case of non linear models, the covariance of the estimates are underestimated from the first time step.

Furthermore a real world experiment is carried out in which the position of a barrel with radius $0.285m$ is estimated. To estimate this barrel the laser is moved back and forth along the y -axis of the world at a fixed height as shown in Figure 6a. Figure 6 also shows experimental results for the estimation of the barrel position with a Baumer laser distance sensor (OADM 2016480/S14F) mounted on a Kuka361 industrial robot. The initial estimation errors for the barrel are approximately $0.4m$ in the x -direction and

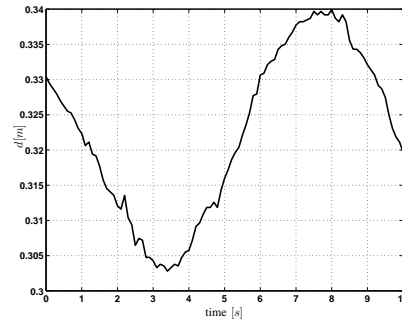
$0.15m$ in the y -direction. The same filtering procedure as explained above is used for the experiment. The figure shows the position of the laser spot on the barrel surface as a function of time, the distance to the surface measured by the laser, and the estimation results. The position of the barrel is estimated with the desired accuracy after approximately $7s$.

VIII. CONCLUSION

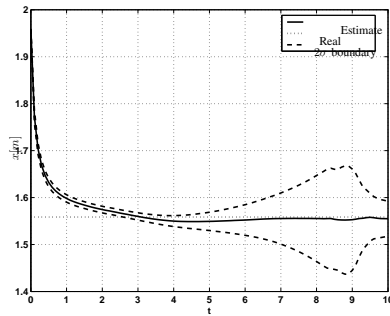
This paper shows details of the application of a systematic approach [1] for constraint-based task specification and control in the presence of geometric uncertainty to a laser tracing example. The task specification is greatly simplified by introducing auxiliary feature coordinates, while uncertainty is modeled using uncertainty coordinates. The uncertainty



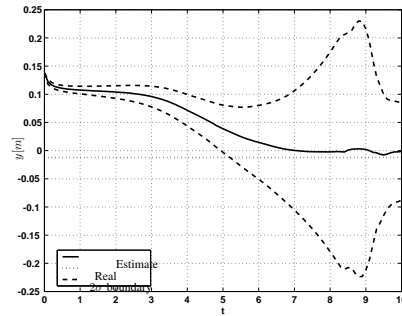
(a) Top view of the motion of laser with respect to barrel during experiment.



(b) Laser distance measurement z_2 .



(c) x -position of the barrel, x_u^b .



(d) y -position of the barrel, y_u^b .

Fig. 6. Laser tracing on barrel: estimation of x - and y - position of the barrel (x_u^b and y_u^b) (experiment). The sudden increase of the covariance as observed from 7s to 9s in (c,d) is due to the limited movement on the barrel resulting in little extra measurement information (b).

coordinates are estimated using a state estimator. Simulation and experimental results for a laser tracing application show the validity and the potential of the approach. [1] presents a detailed theoretical analysis of the presented framework. In this paper other example applications (mobile robot, human-robot interaction, visual servoing, ...) are presented which gives an idea of the generality of the approach.

Future work includes the development of programming support for the systematic task modeling procedure and the efficient implementation of the automatic derivation of the equations for the controller and estimator. For example, the derivation matrices and Jacobians in Section V-VI, will be fully automated.

[16]

REFERENCES

- [1] J. De Schutter, T. De Laet, J. Rutgeerts, W. Decré, R. Smits, E. Aertbeliën, K. Claes, and H. Bruyninckx, "Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty," *Int. J. Robotics Research*, vol. 26, no. 5, pp. 433–455, 2007.
- [2] J. De Schutter and H. Van Brussel, "Compliant Motion I, II," *Int. J. Robotics Research*, vol. 7, no. 4, pp. 3–33, Aug 1988.
- [3] N. Hogan, "Impedance control: An approach to manipulation. Parts I-III," *Trans. ASME J. Dyn. Systems Meas. Control*, vol. 107, pp. 1–24, 1985.
- [4] —, "Stable execution of contact tasks using impedance control," in *Int. Conf. Robotics and Automation*, Raleigh, NC, 1987, pp. 1047–1054.
- [5] H. Kazerooni, "On the robot compliant motion control," *Trans. ASME J. Dyn. Systems Meas. Control*, vol. 111, pp. 416–425, 1989.
- [6] J. Baeten, H. Bruyninckx, and J. De Schutter, "Integrated vision/force robotics servoing in the task frame formalism," *Int. J. Robotics Research*, vol. 22, no. 10, pp. 941–954, 2003.
- [7] M. T. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. SMC-11, no. 6, pp. 418–432, 1981.
- [8] H. Bruyninckx and J. De Schutter, "Specification of force-controlled actions in the "Task Frame Formalism": A survey," *IEEE Trans. Rob. Automation*, vol. 12, no. 5, pp. 581–589, 1996.
- [9] A. P. Ambler and R. J. Popplestone, "Inferring the positions of bodies from specified spatial relationships," *Artificial Intelligence*, vol. 6, pp. 157–174, 1975.
- [10] C. Samson, M. Le Borgne, and B. Espiau, *Robot Control, the Task Function Approach*. Oxford, England: Clarendon Press, 1991.
- [11] J. De Schutter, J. Rutgeerts, E. Aertbelien, F. De Groote, T. De Laet, T. Lefebvre, W. Verdonck, and H. Bruyninckx, "Unified constraint-based task specification for complex sensor-based robot systems," in *Int. Conf. Robotics and Automation*, Barcelona, Spain, 2005, pp. 3618–3623.
- [12] T. De Laet and J. De Schutter, "Control schemes for constraint-based task specification in the presence of geometric uncertainty using auxiliary coordinates," Dept. Mech. Eng., Katholieke Univ. Leuven, Belgium, Internal report 07RP001, 2007.
- [13] K. L. Doty, C. Melchiorri, and C. Bonivento, "A theory of generalized inverses applied to robotics," *Int. J. Robotics Research*, vol. 12, no. 1, pp. 1–19, 1993.
- [14] Y. Nakamura, *Advanced robotics: redundancy and optimization*. Reading, MA: Addison-Wesley, 1991.

- [15] Y. Bar-Shalom and X. Li, *Estimation and Tracking, Principles, Techniques, and Software*. Norwood, MA: Artech House, 1993.
- [16] T. P. Koninckx, P. Peers, P. Dutre, and L. Van Gool, "Scene-adapted structured light," in *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recogn.*, San Diego, USA, 2005, pp. 611–618.