Constrained Optimal Fitting of Three-Dimensional Vector Patterns

G. Calafiore and B. Bona

Abstract—This paper addresses the problem of finding whether a given set of three-dimensional (3-D) vectors (the *object*) can be brought to match a second set of vectors (the *template*) by means of an affine motion, minimizing a measure of the mismatch error and satisfying an assigned set of geometrical constraints. This problem is encountered in many applications of computer vision, robotics, and manufacturing processes, and has been tackled by several authors in the unconstrained case. Spherical, ellipsoidal and polyhedral constraints are here introduced in the problem, and a solution scheme based on an efficient convex optimization algorithm is proposed. An example of application of the proposed methodology to a manufacturing tolerancing problem is also provided.

Index Terms—Absolute orientation, automated tolerancing, computer vision, convex optimization, least squares, visual servo control.

I. INTRODUCTION

A central aspect of many problems related to computer vision and robotics is that of determination of object position and orientation with respect to some standard reference frame. This information may be needed for such purposes as object tracking, visual servo control, precision manipulation, parts pickup in automated manufacturing or as one step in the process of object identification. Many robotic tasks, such as autonomous navigation or inspection of a workpiece on a moving conveyor, require determination of the relative position and orientation between a robot and its surrounding objects [13], [14], [16], [22]. The above information is provided by the homogeneous transformation relating the workpiece (called in the sequel the *object*) to a representation of a theoretical model of the object (called the *template*) given in a standard reference frame.

If the template $\mathcal{A}$ is represented by an ordered set of n characteristic vectors $\mathbf{a}_i \in \mathbb{R}^3$, and the object $\mathcal{B}$ is represented by an ordered set of n characteristic vectors $\mathbf{b}_i \in \mathbb{R}^3$, and point correspondence is given, then one is faced with the classical problem of displacement parameters estimation, that is to find the rotation and translation that minimize a suitable error measure between the template $\mathcal{A}$ and the displaced object $\mathcal{B}_d$. Typical applications of problems of this kind are found in aerospace engineering for the attitude determination of a spacecraft using automatic matching of stellar fields to a CCD image of the current sky sector [5], [16], and in manufacturing processes for the inspection of tolerance of machined parts: the actual geometrical characteristics of the produced items are checked against some template model in order to ascertain if the mechanical tolerances are acceptable [2].

The common solution to the cited problems consists in determining the unknown rotation and translation parameters via unconstrained minimization of a least-squares like cost function. If limits are imposed over the amount of residual error, however, the minimization of the cost function must be subject to geometrical constraints.

Manuscript received January 15, 1997; revised June 25, 1998. This work was supported by Ministero Università Ricerca Scientifica Tecnologica. This paper was recommended for publication by Associate Editor Y. F. Zheng and Editor V. Lumelsky upon evaluation of the reviewers' comments.

The authors are with the Dipartimento di Automatica e Informatica, Politecnico di Torino, Torino 10129, Italy.

Publisher Item Identifier S 1042-296X(98)07437-0.

The purpose of this paper is to improve upon the existing results by providing a theoretical framework into which various types of constraints can be introduced in the optimization problem. It is then shown that the resulting nonlinear optimization problem can be appropriately approximated by a convex constrained problem and a solution algorithm based on sequential unconstrained minimization technique (SUMT) is presented.

II. PRELIMINARIES AND NOTATION

Assume that two ordered vector sets $\mathcal{A}_p = \{\mathbf{a}_i \in \mathbb{R}^3, i = 1, \dots, n_p\}$, $\mathcal{B}_p = \{\mathbf{b}_i \in \mathbb{R}^3, i = 1, \dots, n_p\}$ represent characteristic points of a template and an object, respectively. From these characteristic points one can extract other features of interest, such as unit vectors representing line directions, unit vectors representing normals to planes, and vectors representing relative difference between any other two features. Supposing n_l line directions are represented by the unit vectors $\mathbf{a}_i, i = n_p + 1, \dots, n_p + n_l$; n_f plane normal directions are represented by the unit vectors $\mathbf{a}_i, i = n_p + n_l + 1, \dots, n_p + n_l + n_f$; and n_r relative difference vectors are represented by vectors $\mathbf{a}_i, i = n_p + n_l + n_r + 1, \dots, n$, the total template model will be represented by the set $\mathcal{A} = \{\mathbf{a}_i \in \mathbb{R}^3, i = 1, \dots, n\}$ and the object model by $\mathcal{B} = \{\mathbf{b}_i \in \mathbb{R}^3, i = 1, \dots, n\}$, where $n = n_p + n_l + n_f + n_r$.

Let $\Gamma(\cdot)$ represent a rigid displacement operator acting on the object set, such that the displaced object $\mathcal{B}_d$ is given by $\mathcal{B}_d = \Gamma(\mathcal{B})$, where

$$\begin{aligned} \mathbf{b}_{di} &= \mathbf{R}\mathbf{b}_i + \mathbf{t}, & i &= 1, \dots, n_p \\ \mathbf{b}_{di} &= \mathbf{R}\mathbf{b}_i, & i &= n_p + 1, \dots, n \end{aligned} \quad (1)$$

and $\mathbf{R} \in \text{so}(3)^1$, $\mathbf{t} \in \mathbb{R}^3$ represent, respectively, the rotation matrix and translation vector characterizing the displacement operator Γ . Each feature set can be represented as a matrix containing the ordered feature vectors by columns, so that the relations (1) can be expressed in compact form as

$$\mathbf{B}_d = \Gamma(\mathbf{B}) = \mathbf{R}\mathbf{B} + \mathbf{t}\mathbf{u}' \quad (2)$$

where $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{3,n}$ and $\mathbf{u} \in \mathbb{R}^n$ is a vector of components $u_i = 1$ for $i = 1, \dots, n_p$ and $u_i = 0$ for $i = n_p + 1, \dots, n$. It is now possible to define a *distance* $f(\mathbf{R}, \mathbf{t})$ between the template set $\mathcal{A}$ and the displaced object $\mathcal{B}_d$, as the weighted sum of the squares of the residuals $r_i = \|\mathbf{a}_i - \mathbf{b}_{di}\|$ between the i th template feature and the corresponding displaced object feature. In particular, for $i = 1, \dots, n_p$ and $i = n_p + n_l + n_f + 1, \dots, n$, r_i represents the euclidean distance between corresponding characteristic points or relative difference vectors, while for $i = n_p + 1, \dots, n_p + n_l + n_f$, r_i is related to the angle between the unit vectors $\mathbf{a}_i$ and $\mathbf{b}_{di}$ and therefore to the angular alignment of the relative line directions and plane normals: $r_i^2 = 2(1 - \cos \theta)$, where θ is the angle between $\mathbf{a}_i$ and $\mathbf{b}_{di}$. If w_i are nonnegative weights then

$$f(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^n w_i^2 \|\mathbf{a}_i - \mathbf{b}_{di}\|^2 = \sum_{i=1}^n w_i^2 r_i^2 \quad (3)$$

or, using matrix notation

$$f(\mathbf{R}, \mathbf{t}) = \|(\mathbf{A} - \mathbf{R}\mathbf{B} - \mathbf{t}\mathbf{u}')\mathbf{W}\|_F^2 \quad (4)$$

¹ $\text{so}(3)$ is the special orthogonal group of real 3×3 rotation matrices.

where $\mathbf{W} \in \mathbb{R}^{n,n}$ is a diagonal matrix containing the weights $w_i \geq 0$ on the diagonal, and the subscript F indicates the Frobenius matrix norm².

A key problem is then to determine $\mathbf{R}, \mathbf{t}$ that minimize $f(\mathbf{R}, \mathbf{t})$. This problem is sometimes encountered in the literature as the *absolute orientation problem*, and various solutions have been proposed, remarkably an iterative algorithm [12], a noniterative algorithm based on quaternions [7], [10] and noniterative algorithms based on singular value decomposition [1], [11], [20]. The above problem will be referred to as the unconstrained displacement problem (UDP) and a solution which is useful for the subsequent developments will be outlined in Section III.

A. Introducing Constraints

The central point of the paper is to introduce geometric constraints into the displacement determination problem and to propose an efficient algorithm for its solution. Although the treatment will be mainly theoretical and of general validity, some terminology from tolerancing and manufacturing problems will be used to clarify the developments.

Assume that the solid model created by a CAD system represents the nominal part geometry and data from the model are used to build the template matrix $\mathbf{A}$. Due to the manufacturing process a part instance is produced and the object matrix $\mathbf{B}$ is constructed by inspection of the part instance. Together with the template model a variational model must be provided that represents the class of allowable manufactured instances that may be produced. The variational model is described defining a *tolerance set* around each (or some) describing feature of the template, as specified in the sequel. The object model $\mathcal{B}$ satisfies the specified tolerances if and only if there exists a rigid displacement that brings the object $\mathcal{B}$ into the feasible tolerance sets. While this setting may not be completely general for all tolerance control problems (it is for instance hard to see how tolerance material conditions would be specified in the proposed approach) it does allow to represent some typical tolerance specifications such as limit locating tolerances or parallelism and form tolerances. If applied to tolerancing problems, the proposed framework has some resemblance to the *feasibility space* theory developed by Turner and Wozny [19]. In their approach the feasible space is defined in terms of constraints on the model variables, while in the proposed method the constraints are imposed indirectly on the rotation and translation parameters. The following constraint sets may be imposed on the template features.

- 1) *Spherical*: The i th roto-translated feature $\mathbf{b}_{di} = \mathbf{R}\mathbf{b}_i + \mathbf{t}u_i$ must lie in a sphere of center $\mathbf{a}_i$ and radius ε_i : $\{(\mathbf{R}, \mathbf{t}): \|\mathbf{a}_i - \mathbf{b}_{di}\|^2 < \varepsilon_i^2\}$;
- 2) *Ellipsoidal*: $\mathbf{b}_{di}$ must lie in an ellipsoid centered in $\mathbf{a}_i$, defined by a given positive definite matrix $\mathbf{E}_i$: $\{(\mathbf{R}, \mathbf{t}): (\mathbf{a}_i - \mathbf{b}_{di})' \mathbf{E}_i (\mathbf{a}_i - \mathbf{b}_{di}) < 1\}$;
- 3) *Box, slab or polyhedral*: $\mathbf{b}_{ti}$ must lie inside a polyhedron described by given matrix $\mathbf{P}_i$ and vector $\mathbf{c}_i$: $\{(\mathbf{R}, \mathbf{t}): \mathbf{P}_i \mathbf{b}_{di} < \mathbf{c}_i\}$, where the $<$ sign indicates elementwise inequality;
- 4) Intersections of the above sets.

All the previous relations define basic convex sets whose intersection may be used to build very general (convex) constraints sets. In Section IV, a solution method for the problem with spherical constraints is described in detail and a linear matrix inequalities (LMI) formulation for general constraints is provided.

²For a matrix $\mathbf{X} \in \mathbb{R}^{m,n}$ the square of the Frobenius norm is defined as $\|\mathbf{X}\|_F^2 = \text{tr}(\mathbf{X}\mathbf{X}') = \text{tr}(\mathbf{X}'\mathbf{X})$ and is also equal to the sum of the squares of the euclidean norms of the columns of $\mathbf{X}$.

III. UNCONSTRAINED DISPLACEMENT PROBLEM

Before stepping to the treatment of the constrained problem, the solution of the unconstrained displacement problem (UDP) is revised in this section. Given the template matrix $\mathbf{A} \in \mathbb{R}^{3,n}$, the object matrix $\mathbf{B} \in \mathbb{R}^{3,n}$, and a diagonal matrix $\mathbf{W}$ of nonnegative weights w_i , we call, respectively, $\mathbf{R}_{\text{wls}}$ and $\mathbf{t}_{\text{wls}}$ the rotation matrix and translation vector that solve

$$\begin{aligned} f_{\text{opt}} &= \min_{\mathbf{R} \in \text{so}(3), \mathbf{t} \in \mathbb{R}^3} f(\mathbf{R}, \mathbf{t}) \\ &= \min_{\mathbf{R} \in \text{so}(3), \mathbf{t} \in \mathbb{R}^3} \|(\mathbf{A} - \mathbf{R}\mathbf{B} - \mathbf{t}\mathbf{u}')\mathbf{W}\|_F^2. \end{aligned} \quad (5)$$

It was shown in [1], [2], and [12] that the minimization problem (5) is separable in the two decision variables and can be decoupled as

$$\begin{aligned} f_{\text{opt}} &= \min_{\mathbf{R} \in \text{so}(3)} \min_{\mathbf{t} \in \mathbb{R}^3} \|(\mathbf{A} - \mathbf{R}\mathbf{B} - \mathbf{t}\mathbf{u}')\mathbf{W}\|_F^2 \\ &= \min_{\mathbf{R} \in \text{so}(3)} f_t(\mathbf{R}). \end{aligned} \quad (6)$$

Computing

$$f_t(\mathbf{R}) = \min_{\mathbf{t} \in \mathbb{R}^3} \|(\mathbf{A} - \mathbf{R}\mathbf{B} - \mathbf{t}\mathbf{u}')\mathbf{W}\|_F^2 \quad (7)$$

is a standard norm minimization problem in the Hilbert space $\mathbb{R}^{3,n}$ with inner product $\langle \mathbf{X}, \mathbf{Y} \rangle = \text{tr}(\mathbf{X}\mathbf{Y}')$ and the solution may be found imposing the orthogonality condition $\text{tr}((\mathbf{P} - \mathbf{t}\mathbf{q}')\mathbf{q}\mathbf{t}') = 0 \forall \mathbf{t}$, where $\mathbf{P} = \mathbf{A}\mathbf{W} - \mathbf{R}\mathbf{B}\mathbf{W}$ and $\mathbf{q} = \mathbf{W}\mathbf{u}$. The orthogonality condition implies $\mathbf{P}\mathbf{q} = \mathbf{t}\mathbf{q}'\mathbf{q}$, then substituting and isolating $\mathbf{t}$ one gets

$$\mathbf{t}_{\text{wls}} = (\mathbf{A}\mathbf{W} - \mathbf{R}\mathbf{B}\mathbf{W})\mathbf{W}'\mathbf{u}(\mathbf{u}'\mathbf{W}\mathbf{W}'\mathbf{u})^{-1}. \quad (8)$$

Recalling the structure of $\mathbf{W}$ and $\mathbf{u}$ it follows that $(\mathbf{u}'\mathbf{W}\mathbf{W}'\mathbf{u}) = \gamma = \sum_{i=1}^n w_i^2$, then

$$\mathbf{t}_{\text{wls}} = \frac{1}{\gamma} (\mathbf{A}\mathbf{W} - \mathbf{R}\mathbf{B}\mathbf{W})\mathbf{W}'\mathbf{u} \quad (9)$$

which gives the optimal translation vector $\mathbf{t}_{\text{wls}}$ for any given rotation $\mathbf{R}$. It is possible to show that the so found $\mathbf{t}_{\text{wls}}$ is in fact the translation that makes the weighted centroids (barycentres) of $\mathbf{A}$ and $\mathbf{R}\mathbf{B}$ coincide [1], [12]. Substituting (9) into (7), we get

$$f_t(\mathbf{R}) = \|(\mathbf{A} - \mathbf{R}\mathbf{B})\mathbf{W}\mathbf{K}\|_F^2 \quad (10)$$

where $\mathbf{K} = \mathbf{I} - (1/\gamma)\mathbf{W}'\mathbf{u}\mathbf{u}'\mathbf{W}$. Next we need to determine $\mathbf{R}_{\text{wls}}$ such that

$$f_{\text{opt}} = \min_{\mathbf{R} \in \text{so}(3)} f_t(\mathbf{R}) = \min_{\mathbf{R} \in \text{so}(3)} \|\bar{\mathbf{A}} - \mathbf{R}\bar{\mathbf{B}}\|_F^2 \quad (11)$$

where $\bar{\mathbf{A}} = \mathbf{A}\mathbf{W}\mathbf{K}$ and $\bar{\mathbf{B}} = \mathbf{B}\mathbf{W}\mathbf{K}$. This problem is sometimes referred to as the *procrustean transformation problem* [9]: given two matrices $\bar{\mathbf{A}}, \bar{\mathbf{B}} \in \mathbb{R}^{3,n}$, we wish to find the optimal rotation $\mathbf{R}_{\text{wls}}$ that minimizes the Frobenius norm of $\bar{\mathbf{A}} - \mathbf{R}\bar{\mathbf{B}}$

$$\|\bar{\mathbf{A}} - \mathbf{R}\bar{\mathbf{B}}\|_F^2 = \|\bar{\mathbf{A}}\|_F^2 - 2\text{tr}(\bar{\mathbf{A}}\bar{\mathbf{B}}'\mathbf{R}) + \|\bar{\mathbf{B}}\|_F^2. \quad (12)$$

Let $\mathbf{U}\mathbf{S}\mathbf{V}'$ be the singular value decomposition (SVD) of $\bar{\mathbf{A}}\bar{\mathbf{B}}'$, then (recalling commutativity under the trace operator)

$$\text{tr}(\bar{\mathbf{A}}\bar{\mathbf{B}}'\mathbf{R}) = \text{tr}(\mathbf{S}\mathbf{V}'\mathbf{R}'\mathbf{U}) = \sum_{i=1}^3 \sigma_i t_{i,i} \quad (13)$$

where $\mathbf{V}'\mathbf{R}'\mathbf{U} \triangleq \mathbf{T} = [t_{i,j}]$ is an orthogonal matrix and σ_i are the singular values of $\bar{\mathbf{A}}\bar{\mathbf{B}}'$ (diagonal elements of $\mathbf{S}$). Minimizing (12) is then equivalent to maximizing the sum (13), and this is clearly achieved by maximizing $t_{i,i}, i = 1, \dots, 3$. As $\mathbf{T}$ is orthogonal, $\mathbf{T}'\mathbf{T} = \mathbf{I}$, so that $\sum_{j=1}^3 t_{i,j}^2 = 1$ for $i = 1, 2, 3$, therefore $t_{i,i} \leq 1$

and the maximum is achieved setting $t_{i,i} = 1$ and $t_{i,j} = 0, i \neq j$, i.e., $\mathbf{T}$ must be the identity matrix: $\mathbf{T} = \mathbf{V}'\mathbf{R}'\mathbf{U} = \mathbf{I}$ and

$$\mathbf{R}_{\text{wls}} = \mathbf{R} = \mathbf{UV}' \quad (14)$$

which gives the desired optimal orthogonal matrix. It should be noticed that we are here looking for a rotation matrix ($\det \mathbf{R} = +1$), while (14) can sometimes give a reflection ($\det \mathbf{R} = -1$). This uncertainty may be resolved using the method proposed in [1] (for a treatment of this topic see also [20]).

IV. THE CONSTRAINED DISPLACEMENT PROBLEM

In this section we describe a method for solving the constrained displacement problem (CDP) based on an affine approximation of the $\text{so}(3)$ set around a suitable initial rotation matrix, allowing for a convex formulation of the resulting approximated optimization problem. Without loss of generality, the spherical constrained case is treated in detail in the sequel while an LMI formulation is provided for the general constrained case.

The CDP is formulated as follows. Given a template matrix $\mathbf{A} \in \mathbb{R}^{3,n}$, an object matrix $\mathbf{B} \in \mathbb{R}^{3,n}$ and m spherical constraints at the template features numbered $c_1, \dots, c_m$, find the rotation $\mathbf{R}$ and translation $\mathbf{t}$ that minimize (15), subject to the constraints (16), (17), or state that no solution exists (problem is unfeasible)

$$f_{\text{opt}} = \min_{\mathbf{R}, \mathbf{t}} \|\mathbf{A} - \mathbf{RB} - \mathbf{t}\mathbf{u}'\|_F^2 \quad (15)$$

$$\text{subj. to } \|\mathbf{a}_i - \mathbf{Rb}_i - \mathbf{t}u_i\|^2 < \varepsilon_i^2, \quad i = c_1, \dots, c_m \quad (16)$$

$$\mathbf{R} \in \text{so}(3). \quad (17)$$

If the minimization of (15) is not of interest, but the testing of existence of an admissible displacement is priority (as in tolerance testing), then we formulate a feasibility displacement problem: find, if they exist, $(\mathbf{R}, \mathbf{t})$ such that

$$\|\mathbf{a}_i - \mathbf{Rb}_i - \mathbf{t}u_i\|^2 < \varepsilon_i^2, \quad i = c_1, \dots, c_m \quad (18)$$

$$\mathbf{R} \in \text{so}(3). \quad (19)$$

The feasibility problem above is put in an equivalent minimization form as

$$\eta_{\text{opt}} = \min_{\mathbf{R}, \mathbf{t}, \eta} \eta, \quad \text{subject to:} \quad (20)$$

$$\|\mathbf{a}_i - \mathbf{Rb}_i - \mathbf{t}u_i\|^2 - \varepsilon_i^2 - \eta < 0$$

$$i = c_1, \dots, c_m; \quad (21)$$

$$\mathbf{R} \in \text{so}(3). \quad (22)$$

It can be easily verified that problem (18)–(19) is feasible if and only if problem (20)–(22) has a nonpositive optimal objective value $\eta_{\text{opt}} \leq 0$. The latter problem will be denoted as FDP (feasibility displacement problem) in the sequel. The advantage of the formalization in minimization form resides in the fact that numerical optimization algorithms usually require an initial feasible point, which, for problem (20)–(22), may always be found for a sufficiently large value of η (see Section IV-A). Both CDP and FDP are nonlinear programming problems, for which a solution scheme is presented in Section IV-A.

A. Representation and Approximation of $\text{so}(3)$

A standard result from group theory states that the set $\text{sk}(3)$ of real 3×3 skew-symmetric matrices can be transformed by an exponential mapping onto the group $\text{so}(3)$; [21]. In particular, every matrix $\mathbf{R} \in \text{so}(3)$ may be written in terms of a 3×3 skew-symmetric matrix $\mathbf{S} \in \text{sk}(3)$ as

$$\mathbf{R} = \mathbf{I} + \sin \theta \mathbf{H} + (1 - \cos \theta) \mathbf{H}^2; \quad \mathbf{S} = \theta \mathbf{H} \quad (23)$$

where $\mathbf{H}$ is a skew-symmetric matrix of the components of the unit vector $\mathbf{h} = [h_1 \ h_2 \ h_3]'$ representing the axis of rotation, and θ is the angle of rotation around $\mathbf{h}$

$$\mathbf{H} = \text{skew}(\mathbf{h}) = \begin{bmatrix} 0 & -h_3 & h_2 \\ h_3 & 0 & -h_1 \\ -h_2 & h_1 & 0 \end{bmatrix}, \quad \|\mathbf{H}\|_F^2 = 2. \quad (24)$$

It is important to note that $\text{sk}(3)$ is an affine space as $\text{sk}(3) = \{\mathbf{S} \in \mathbb{R}^{3,3} : \mathbf{S} + \mathbf{S}' = \mathbf{O}\}$, and, from (23), $\|\mathbf{S}\|_F^2 = 2\theta^2$. Given a positive scalar ξ , the space $\mathcal{S}_\xi = \{\mathbf{S} \in \text{sk}(3) : \|\mathbf{S}\|_F^2 < \xi\}$ is convex in $\mathbf{S}$ and every point in $\mathcal{S}_\xi$ represents (by means of the exponential mapping) a rotation of angle $|\theta| = \|\mathbf{S}\|/\sqrt{2} < \sqrt{\xi}/2$ around the direction $\mathbf{h}$ given by the components of $\mathbf{H} = \sqrt{2}(\mathbf{S}/\|\mathbf{S}\|)$. Searching over the space of rotation matrices with rotation angle limited by $\sqrt{\xi}/2$ is then equivalent to searching over the (convex) space $\mathcal{S}_\xi$.

A total rotation $\mathbf{R}$ can be represented as a composition of an initial rotation $\mathbf{R}_0$ and a further “corrective” rotation $\mathbf{R}_s$, so that $\mathbf{R} = \mathbf{R}_s \mathbf{R}_0$. If the angle of corrective rotation is “small” (ξ small), $\mathbf{R}_s$ can be approximated by its first order expansion $\mathbf{R}_s \simeq \mathbf{R}_s = \mathbf{I} + \mathbf{S}$, where $\mathbf{S} = \text{skew}(\mathbf{s}), \mathbf{s} = [s_1 \ s_2 \ s_3]'$, and the total rotation $\mathbf{R}$ is approximately given by $\mathbf{R} \simeq \mathbf{R}_0 + \mathbf{S}\mathbf{R}_0$. With the introduced approximation the objective function (15) is substituted by

$$\begin{aligned} \tilde{f} &= \|\mathbf{A} - \tilde{\mathbf{R}}_s \mathbf{R}_0 \mathbf{B} - \mathbf{t}\mathbf{u}'\|_F^2 \\ &= \|\mathbf{A} - (\mathbf{I} + \mathbf{S})\mathbf{R}_0 \mathbf{B} - \mathbf{t}\mathbf{u}'\|_F^2 \\ &= \|\mathbf{A} - \mathbf{R}_0 \mathbf{B} - \mathbf{S}\mathbf{R}_0 \mathbf{B} - \mathbf{t}\mathbf{u}'\|_F^2 \\ &= \|\mathbf{E} - \mathbf{S}\mathbf{B}_r - \mathbf{t}\mathbf{u}'\|_F^2 \end{aligned} \quad (25)$$

where $\mathbf{E} = [\mathbf{e}_1 \ \mathbf{e}_2 \ \dots \ \mathbf{e}_n] = \mathbf{A} - \mathbf{R}_0 \mathbf{B}$ and $\mathbf{B}_r = [\mathbf{b}_{r1} \ \mathbf{b}_{r2} \ \dots \ \mathbf{b}_{rn}] = \mathbf{R}_0 \mathbf{B}$.

Given an initial rotation matrix $\mathbf{R}_0$, the CDP (15)–(17) is approximated by the following convex programming problem (CDP*): determine, if they exist, $\mathbf{S}$ and $\mathbf{t}$ such that

$$\tilde{f}_{\text{opt}} = \min_{\mathbf{S}, \mathbf{t}} \tilde{f}(\mathbf{S}, \mathbf{t}), \quad \text{subject to:} \quad (26)$$

$$\|\mathbf{e}_i - \mathbf{S}\mathbf{b}_{ri} - \mathbf{t}u_i\|^2 < \varepsilon_i^2, \quad i = c_1, \dots, c_m \quad (27)$$

$$\mathbf{S} \in \text{sk}(3) \quad (28)$$

$$\|\mathbf{S}\|_F^2 < \xi \quad (29)$$

where ξ is a parameter that bounds the angle of corrective rotation $\theta^2 \leq \xi/2$.

The feasibility problem FDP (20)–(22) is approximated by the following convex programming problem (FDP*):

$$\eta_{\text{opt}} = \min_{\mathbf{S}, \mathbf{t}, \eta} \eta, \quad \text{subject to:} \quad (30)$$

$$\|\mathbf{e}_i - \mathbf{S}\mathbf{b}_{ri} - \mathbf{t}u_i\|^2 - \varepsilon_i^2 - \eta < 0, \quad i = c_1, \dots, c_m \quad (31)$$

$$\mathbf{S} \in \mathcal{S} \quad (32)$$

$$\|\mathbf{S}\|_F^2 - \xi - \eta < 0 \quad (33)$$

therefore CDP* is feasible if and only if FDP* has a nonpositive optimal objective value $\eta_{\text{opt}} \leq 0$. An initial admissible point for FDP* is given by

$$\mathbf{S} = \mathbf{0}, \mathbf{t} = \mathbf{t}_0, \eta > \bar{\eta}$$

where $\bar{\eta}$ is the maximum constraints deviation relative to the point $(\mathbf{R}_0, \mathbf{t}_0)$

$$\bar{\eta} = \max_{i=\{c_1, \dots, c_m\}} \{\|\mathbf{e}_i - \mathbf{t}_0 u_i\|^2 - \varepsilon_i^2\}. \quad (34)$$

The effects of the introduced approximation on the solution of the original problems CDP and FDP are discussed in Section IV-C. The procedure of solution is as follows.

- 1) Determine an initial point $(\mathbf{R}_0, \mathbf{t}_0)$ solving an unconstrained displacement problem (UDP), as described in Section III. The

choice of the weights matrix to use in this first step is not critical, as the resulting solution serves only as a starting point for the algorithm. A reasonable choice is, for instance, to give more weight to the feature vectors with stricter constraints, thus taking the w_i 's to be inversely proportional to the tolerance radii ε_i .

- 2) If $(\mathbf{R}_0, \mathbf{t}_0)$ is feasible go to 3, otherwise solve FDP* using the affine approximation of $\text{so}(3)$ around $\mathbf{R}_0$. If $\eta_{\text{opt}} \leq 0$ go to 3, else go to 5.
- 3) If only feasibility is requested go to 4, else solve CDP* starting from the given feasible point.
- 4) A solution is found, finish.
- 5) No feasible solution to the problem instance, finish.

B. A Scheme for SUMT Algorithm

Problems FDP* and CDP* are solved using a SUMT-type algorithm [8]. The logical scheme of the algorithm is here briefly discussed, with reference to the following standard problem

$$f_{\text{opt}} = \min_{\mathbf{x}} f_0(\mathbf{x}) \quad (35)$$

$$\text{subject to } f_i(\mathbf{x}) < 0, \quad i = 1, \dots, m \quad (36)$$

where $f_i(\mathbf{x}), i = 0, \dots, m$ is a set of convex smooth functions of the decision vector $\mathbf{x}$ and we assume to have an initial feasible point $\mathbf{x}_0$. The constraints form a convex set $\mathcal{C} = \{\mathbf{x}: f_i(\mathbf{x}) < 0, i = 1, \dots, m\}$, and a *logarithmic barrier function* is defined for the constraints as

$$\phi(\mathbf{x}) = \begin{cases} -\sum_{i=1}^m \log(-f_i(\mathbf{x})) & \mathbf{x} \in \mathcal{C} \\ +\infty & \mathbf{x} \notin \mathcal{C}. \end{cases} \quad (37)$$

ϕ is a smooth (as $\log(-f_i(\mathbf{x}))$ is differentiable on $\mathcal{C}$) and convex³ function on $\mathcal{C}$ and $\phi \rightarrow \infty$ as $\mathbf{x}$ approaches the boundary of $\mathcal{C}$. We now form a weighted sum of the objective function and the barrier: for $\alpha > 0$

$$\psi(\mathbf{x}) = \alpha f_0(\mathbf{x}) + \phi(\mathbf{x}) \quad (38)$$

ψ is also smooth and convex, and given an initial feasible point we can compute the minimum of ψ via Newton's method

$$\mathbf{x}^*(\alpha) = \text{argmin}(\psi) \quad (39)$$

as α increases toward ∞ , $f_0(\mathbf{x}^*(\alpha)) \rightarrow f_{\text{opt}}$. The scheme of the algorithm is as follows:

- 1) given feasible $\mathbf{x}_0$, set $\mathbf{x} = \mathbf{x}_0, \alpha > 0, \beta > 1, M \gg 1$ (α, β, M problem dependent constants);
- 2) compute $\mathbf{x}^*(\alpha)$ via Newton's method, starting at $\mathbf{x}$ [inner iteration];
- 3) set $\mathbf{x} = \mathbf{x}^*(\alpha)$;
- 4) increase α : $\alpha = \alpha\beta$;
- 5) if $\alpha > M$, exit; else goto step 2 [outer iteration].

The algorithm generates a sequence of points $\mathbf{x}^*(\alpha)$ leading to the optimal solution as $\alpha \rightarrow \infty$. In each outer iteration an unconstrained minimization problem is solved via Newton's method (inner iteration), so the actual constrained problem is solved by means of a sequence of unconstrained minimizations. An important feature of this method is the possibility of exploiting duality (see [15] and the

³As f_i is convex, $-f_i$ is concave and this implies that $-f_i$ is log-concave [17], that is $\log(-f_i)$ is concave, so that $-\log(-f_i)$ is convex. The statement then follows recalling that the sum of convex functions is still a convex function.

derivation in [4]). Writing the first order optimality conditions for (39), we have

$$\alpha \nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \frac{1}{-f_i(\mathbf{x}^*)} \nabla f_i(\mathbf{x}^*) = 0 \quad (40)$$

rewriting it as $\nabla f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i \nabla f_i(\mathbf{x}^*) = 0$, where $\lambda_i = -1/(\alpha f_i(\mathbf{x}^*)) > 0$, we see that $\mathbf{x}^*$ also minimizes the lagrangian function $f_0(\mathbf{x}) + \sum \lambda_i f_i(\mathbf{x})$. Thus we know the Lagrange dual function at λ

$$f_{\text{opt}} \geq g(\lambda) = \inf_{\mathbf{x}} \left(f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}) \right) \quad (41)$$

$$= f_0(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}^*) = f_0(\mathbf{x}^*) - \frac{m}{\alpha}. \quad (42)$$

Then for each computed $\mathbf{x}^*(\alpha)$ we also get for free a lower bound on the optimal solution

$$f_0(\mathbf{x}^*) \geq f_{\text{opt}} \geq f_0(\mathbf{x}^*) - \frac{m}{\alpha} \quad (43)$$

this is useful to set up a nonheuristic stopping criterion for the optimization algorithm, i.e. the algorithm exits when the required tolerance on the optimal objective value is reached. The required exit tolerance $\varepsilon = m/\alpha$ can be a priori specified setting $M = m/\varepsilon$ (step 5 of the algorithm). It is also important to note that the possibility of having a dual point (43) at each outer iteration can significantly speed up the detection of unfeasibility in problem FDP*. At any outer iteration we compute a dual objective value $f_d(\mathbf{x}^*(\alpha)) = f_0(\mathbf{x}^*(\alpha)) - m/\alpha$, which is a lower bound on the optimal objective value f_{opt} : $f_{\text{opt}} \geq f_d(\mathbf{x}^*(\alpha))$. If $f_d > 0$ then f_{opt} will certainly be positive, which in the FDP* case means that $\eta_{\text{opt}} > 0$ and CDP* is infeasible. This fact greatly enhances the efficiency of the algorithm for feasibility determination, as, if at any iteration a positive dual objective is computed, one can immediately exit and detect infeasibility, with no need to find the actual value of η_{opt} .

Another useful consequence of convex duality is that the Lagrange multipliers λ_i represent the sensitivity of the objective function to constraints variations [15]. When applied to FDP* this gives the following interesting interpretation: suppose that a positive optimal objective value η_{opt} is found, with associated Lagrange multipliers $\lambda_i > 0$; the fact that η_{opt} is positive indicates that CDP* is globally infeasible while the λ_i add information about which features are more critical or which tolerances need to be relaxed in order to gain feasibility: small value of λ_i indicates small sensitivity of η_{opt} to variations of the tolerance radius ε_i , while large value of λ_i means that small increase of ε_i moves η_{opt} toward the feasible region.

C. Approximation Issues

The solution of the approximated problem CDP* does not necessarily represent a valid solution for the original problem CDP. However, it will be shown that the effects of the approximation result in a "confidence interval," or precision, on constraints satisfaction which can be a priori controlled via the parameter ξ . Let an admissible solution of CDP* be given by the matrix $\tilde{\mathbf{R}}_s = \mathbf{I} + \mathbf{S}$ and the vector $\mathbf{t}$, where $\tilde{\mathbf{R}}_s$ approximates a true rotation matrix $\mathbf{R}_s$ for small angles of rotation θ . This solution satisfies the constraints (27)–(29), but not necessarily the original constraints (16)–(17). In order to evaluate (16) we need to determine a true rotation matrix $\mathbf{R}_s$, given the approximate $\tilde{\mathbf{R}}_s$. $\mathbf{R}_s$ may be computed as the rotation matrix closest to $\tilde{\mathbf{R}}_s$ in least-squares sense. Defining the residue $\Delta \mathbf{R} = \mathbf{R}_s - \tilde{\mathbf{R}}_s$, it can be shown [9] that the rotation matrix that minimizes $\|\Delta \mathbf{R}\|_F$ is given

by SVD⁴ of $\tilde{\mathbf{R}}_s$: $\tilde{\mathbf{R}}_s = \mathbf{P}\mathbf{D}\mathbf{Q}$, then

$$\mathbf{R}_s = \mathbf{P}\mathbf{Q}. \quad (44)$$

The approximation error is easily found as

$$\|\Delta\mathbf{R}\|_F^2 = \sum_{i=1}^3 (\sigma_i - 1)^2 \quad (45)$$

where σ_i are the singular values of $\tilde{\mathbf{R}}_s$. Recalling the structure (23), (24) of $\mathbf{S} = \theta\mathbf{H}$, the σ_i 's can be computed as $\mathbf{D} = \text{diag}\{\sqrt{1+s_1^2+s_2^2+s_3^2}, \sqrt{1+s_1^2+s_2^2+s_3^2}, 1\}$. Considering that $\|\mathbf{S}\|_F^2 = 2(s_1^2+s_2^2+s_3^2)$ and $\|\mathbf{S}\|_F^2 = \theta^2\|\mathbf{H}\|_F^2 = 2\theta^2$, we have that $s_1^2+s_2^2+s_3^2 = \theta^2$ and the error can be expressed as

$$\|\Delta\mathbf{R}\|_F^2 = 2(\sqrt{1+\theta^2} - 1)^2 \simeq \frac{1}{2}\theta^4 \text{ for small } \theta. \quad (46)$$

Recalling now that the constraint (29) implies $\theta^2 < (\xi/2)$, we get a bound on the error

$$\|\Delta\mathbf{R}\|_F^2 < \frac{1}{8}\xi^2. \quad (47)$$

We now evaluate the constraints (16) using $\mathbf{R} = \mathbf{R}_s\mathbf{R}_0$. By the triangle inequality we get the following chain of inequalities:

$$\begin{aligned} \|\mathbf{a}_i - \mathbf{R}\mathbf{b}_i - \mathbf{t}u_i\|^2 & \\ & \leq \|\mathbf{a}_i - \tilde{\mathbf{R}}_s\mathbf{b}_{ri} - \mathbf{t}\|^2 + \|\Delta\mathbf{R}\mathbf{b}_{ri}\|^2 \end{aligned} \quad (48)$$

$$< \varepsilon_i^2 + \|\Delta\mathbf{R}\|_F^2 \|\mathbf{b}_{ri}\|^2 < \varepsilon_i^2 + \frac{1}{8}\xi^2 \|\mathbf{b}_{ri}\|^2. \quad (49)$$

The above bounds are computable off-line and give the maximum constraint deviations between the actual and approximated problems: assuming for ease of discussion $\|\mathbf{b}_{ri}\| = 1$, we see that if the approximated problem CDP* is feasible, then the actual problem CDP is also feasible with confidence $\xi^2/8$. The parameter ξ then regulates the range of corrective rotation angles θ that one want to consider in the problem and the precision on constraint satisfaction that one need to achieve; setting for instance $\xi = 0.01$ will cover corrective rotation angles $\theta < 4^\circ$, with precision on constraints of about 10^{-5} , while $\xi = 0.001$ yields $\theta < 1.3^\circ$, with precision of about 10^{-7} .

It should be remarked at this point that the proposed solution scheme can be used iteratively in order to increase the portion of parameter space explored by the algorithm and minimize the effects due to the initial choice of the weight matrix $\mathbf{W}$ and the resulting starting point $\mathbf{R}_0$. In this iterative setting one may start with a poor initial point $\mathbf{R}_0$, obtained for instance solving UDP with $\mathbf{W} = \mathbf{I}$, then the solution of FDP* is computed and projected onto $\text{so}(3)$ by (44), obtaining a rotation matrix $\mathbf{R}_s^{(1)}$ and an optimal objective value $\eta_{\text{opt}}^{(1)}$. Setting then $\mathbf{R}_0 \leftarrow \mathbf{R}_s^{(1)}\mathbf{R}_0$, FDP* is solved again starting at the updated initial point $\mathbf{R}_0$, and $\mathbf{R}_s^{(2)}$ and $\eta_{\text{opt}}^{(2)}$ are obtained. The process is then repeated until a feasible solution is found, or the relative variation of η_{opt} is below a predefined level and unfeasibility is detected.

Extensive computation on several test cases, however, has shown that with an appropriate choice of $\mathbf{W}$ (for instance, $w_i = 1/\varepsilon_i$ for $i = c_1, \dots, c_m$), one iteration gives the correct answer, and a second iteration may be used to confirm that the actual variation of η_{opt} is below the specified limit. Blind choice of $\mathbf{W} = \mathbf{I}$ will usually require $2 \div 4$ or less iterations to converge to the correct answer.

⁴It should be noted that, given $\mathbf{S} \in \text{sk}(3)$, one could reconstruct a rotation matrix $\mathbf{R}_s$ using the exponential mapping (23). However, this solution does not minimize the norm of the residue, i.e. it is not the closest matrix in $\text{so}(3)$ to $\mathbf{R}_s$. It can be shown that the solution found by SVD and the one found using (23) approximately coincide for small angles of rotation θ .

D. LMI Formulation for General Constraints

While the discussed SUMT algorithm may deal with the CDP* problem with generic (convex) constraints, the actual derivation of gradients and barriers for nonspherical constraints may be more involved and is not explicitly treated in this paper. Instead, it is next shown how the generic CDP* may be cast as a semidefinite programming problem (SDP) and solved using available software tools for linear matrix inequalities (LMI) optimization [6].

Recalling that $\mathbf{b}_{di} = \mathbf{R}\mathbf{b}_i + \mathbf{t}u_i = (\mathbf{I} + \mathbf{S})\mathbf{R}_0\mathbf{b}_i + \mathbf{t}u_i$, the constraints and objective function of CDP* are cast in LMI form in the following way (see [3]).

- 1) Spherical constraints:

$$\|\mathbf{a}_i - \mathbf{b}_{di}\|^2 < \varepsilon_i^2 \Leftrightarrow \begin{bmatrix} \varepsilon_i^2 \mathbf{I} & (\mathbf{a}_i - \mathbf{b}_{di}) \\ (\mathbf{a}_i - \mathbf{b}_{di})' & 1 \end{bmatrix} > 0$$

where the latter $>$ sign means "positive definite."

- 2) Ellipsoidal constraints:

$$\begin{aligned} (\mathbf{a}_i - \mathbf{b}_{di})' \mathbf{E}_i (\mathbf{a}_i - \mathbf{b}_{di}) &< 1 \\ \Leftrightarrow \begin{bmatrix} \mathbf{E}_i^{-1} & (\mathbf{a}_i - \mathbf{b}_{di}) \\ (\mathbf{a}_i - \mathbf{b}_{di})' & 1 \end{bmatrix} &> 0. \end{aligned}$$

- 3) Polyhedral constraints:

$$\mathbf{P}_i \mathbf{b}_{di} < \mathbf{c}_i \Leftrightarrow \text{diag}\{\mathbf{c}_i\} - \text{diag}\{\mathbf{P}_i \mathbf{b}_{di}\} > 0$$

where $\text{diag}\{\mathbf{x}\}$ is a diagonal matrix having the elements of vector $\mathbf{x}$ along the diagonal and the first inequality is an elementwise inequality while the second one means "positive definite."

- 4) Objective function:

$$\begin{aligned} \min \quad & \|\mathbf{E} - \mathbf{S}\mathbf{B}_r - \mathbf{t}\mathbf{u}'\|_F^2 \\ \Leftrightarrow \min \quad & \gamma \\ \text{subj. to} \quad & \|\mathbf{E} - \mathbf{S}\mathbf{B}_r - \mathbf{t}\mathbf{u}'\|_F^2 < \gamma, \gamma > 0. \end{aligned}$$

If $\mathbf{C} = \mathbf{C}(\mathbf{S}, \mathbf{t}) = \mathbf{E} - \mathbf{S}\mathbf{B}_r - \mathbf{t}\mathbf{u}'$, the minimization above may be expressed as

$$\begin{aligned} \min \quad & \gamma \\ \text{subj. to} \quad & \text{tr } \mathbf{C}' \frac{1}{\gamma} \mathbf{C} < 1, \gamma > 0 \end{aligned}$$

which can be expressed in LMI form using a slack matrix variable $\mathbf{Z} = \mathbf{Z}'$ as

$$\begin{aligned} \min \quad & \gamma \\ \text{subj. to} \quad & \text{tr } \mathbf{Z} < 1; \\ & \begin{bmatrix} \mathbf{Z} & \mathbf{C}'(\mathbf{S}, \mathbf{t}) \\ \mathbf{C}(\mathbf{S}, \mathbf{t}) & \gamma \mathbf{I} \end{bmatrix} > 0. \end{aligned}$$

In conclusion CDP* may be cast in standard LMI form and all the discussed constraint sets may be treated in this way. Numerically efficient results may then be obtained using one of the available LMI solvers (see for instance [6]).

V. NUMERICAL EXAMPLE: TOLERANCE INSPECTION

Fig. 1(a) illustrates a simple two-dimensional (2-D) example of locating tolerance specification [18]. The location of the top edge of the part is allowed to vary anywhere within the tolerance zone. The tolerance limits on the variation of the top left and top right vertices relative to the base are imposed as spherical constraints centered at the $\mathbf{a}_1, \mathbf{a}_2$ template points. If explicit parallelism tolerance is also required, as in Fig. 1(b), it can be imposed as a spherical constraint on the relative difference template vector $\mathbf{a}_5 = \mathbf{a}_2 - \mathbf{a}_1$. The complete template model is then represented by the matrix

$$\mathbf{A} = \begin{bmatrix} 0 & 2 & 2 & 0 & 2 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Fig. 1. (a) Limit tolerance governing the location of one planar feature relative to the base. (b) Parallelism tolerance applied to the top edge. The degree of nonparallelism is measured by the difference in the vertical positions of the two vertices a_1, a_2 .

and the associated variational model is given by the spherical constraints of radii $\varepsilon_1 = \varepsilon_2 = 0.1, \varepsilon_5 = 0.05$; datum points a_3, a_4 are assigned a default tolerance $\varepsilon_3 = \varepsilon_4 = 10^{-6}$.

An object model representing the manufactured instance is now constructed. We first consider an object not respecting the parallelism tolerance, given by a matrix model $B^{(0)} = [b_1^{(0)} b_2^{(0)} \dots b_5^{(0)}]$, where

$$b_1^{(0)} = \begin{bmatrix} 0 \\ 0.96 \\ 0 \end{bmatrix}, \quad b_2^{(0)} = \begin{bmatrix} 2 \\ 1.03 \\ 0 \end{bmatrix}, \quad b_3^{(0)} = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$$

$$b_4^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad b_5^{(0)} = b_2^{(0)} - b_1^{(0)}.$$

A translation $t_b = [1 \ 2 \ 3]'$ and a rotation of 30° around the z axis is then imposed to the object, so that the final object model is given by

$$B = R_{z,30}(B^{(0)} + t_b u')$$

$$= \begin{bmatrix} -0.6140 & 1.0831 & 1.5981 & -0.1340 & 1.6971 \\ 3.0634 & 4.1241 & 3.2321 & 2.2321 & 1.0606 \\ 3.0000 & 3.0000 & 3.0000 & 3.0000 & 0 \end{bmatrix}$$

where $u = [1 \ 1 \ 1 \ 1 \ 0]'$. Following now the steps of Section IV, an initial solution (R_0, t_0) is computed solving UDP with $W = I$ (see also discussion at the end of Section IV-C)

$$R_0 = \begin{bmatrix} 0.8541 & 0.5201 & 0 \\ -0.5201 & 0.8541 & 0 \\ 0 & 0 & 1.0000 \end{bmatrix}, \quad t_0 = \begin{bmatrix} -1.0577 \\ -1.9501 \\ -3.0000 \end{bmatrix}.$$

The computed R_0 represents a rotation of -31.3374° around the z axis. Running the SUMT algorithm on FDP* (Fig. 2) we get a rotation matrix R that represents a rotation of -30.6986° around the z axis, and $\eta_{\text{opt}} = 1.4866 \cdot 10^{-4} > 0$: as η_{opt} is positive, the problem is candidate for being infeasible. Remark that we could have stopped the algorithm at the third outer iteration, when a positive dual objective is obtained for the first time, as discussed in Section IV-B.

With a second run of the algorithm, starting at the current R , we get $\eta_{\text{opt}} = 9.1174 \cdot 10^{-5}$ and a third run does not modify this last value, thus confirming, as expected, the infeasibility of the problem.

Checking the Lagrange multipliers associated to η_{opt}

$$\lambda = [0.0000, 0.0000, 0.4571, 0.4571, 0.0857]'$$

indicates that to move η_{opt} toward the feasible region one needs to relax the parallelism constraint ε_5 , which is indeed related to the out-of-tolerance feature⁵. Relaxing then the parallelism tolerance to $\varepsilon_5 = 0.08$ the object model is by construction feasible with respect to the given template. Starting the SUMT algorithm with the initial least squares solution computed for $W = I$ as before we

⁵The multipliers λ_3 and λ_4 should not be considered, as they are related to the datum auxiliary constraints.

Fig. 2. Crosses represent values of the objective function η at the end of each outer loop (result of Newton minimization). Circles represent dual objective values, i.e. lower bounds for the optimal solution. At each outer loop the optimal solution is guaranteed to lie between the cross and the circle, and the algorithm exits when the primal-dual gap is smaller than $\varepsilon = m/M = 10^{-10}$.

get $\eta_{\text{opt}}^{(1)} = 7.4228 \cdot 10^{-8}$ at the first run, while a feasible solution is found at the second run with $\eta_{\text{opt}}^{(2)} = -9.5000 \cdot 10^{-8}$, yielding the feasible rotation matrix and translation vector

$$R = \begin{bmatrix} 0.8660 & 0.5000 & 0.0000 \\ -0.5000 & 0.8660 & 0.0000 \\ 0.0000 & 0.0000 & 1.0000 \end{bmatrix}, \quad t = \begin{bmatrix} -1.0000 \\ -2.0000 \\ -3.0000 \end{bmatrix}$$

where R represents, as expected, a z axis rotation of -30° .

VI. CONCLUSION

A solution method for the matching of three-dimensional (3-D) vector patterns under rigid motion and geometric constraints has been described. The method can accommodate various types of constraint specifications and is based on an efficient interior-point convex optimization algorithm. One major feature of the proposed approach is that an answer is always returned in a pre-specified number of iterations, either in form of the optimal displacement parameters (R, t) , or as a certificate of nonfeasibility for the given problem instance. The convex formulation of the problem also permits to exploit duality, thus guaranteeing ε -suboptimality for the computed solution and greatly enhancing the speed of infeasibility detection, as discussed in Section IV-B. Infeasibility is however treated as a “global” property of the object, so that it is not always possible to exactly identify which object features are responsible for it, although useful information in this respect is obtained by constraints sensitivity analysis via the Lagrange multipliers supplied by the algorithm. The algorithm for the spherical constrained case has been implemented in a MATLAB package, while an LMI formulation for the general constrained case is proposed in Section IV-D. An example of application to a manufacturing tolerancing problem has also been presented to illustrate the use of the proposed method.

ACKNOWLEDGMENT

The authors wish to thank S. Boyd for the many useful discussions. A special thank also goes to L. Vandenberghe and all the ISL group at Stanford University.

REFERENCES

- [1] K. S. Arun, T. S. Huang, and S. D. Blostein, “Least-squares fitting of two 3-D point sets,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 698–700, 1987.
- [2] B. Bona and G. Menga, “Best fitting of three-dimensional bodies,” *Proc. IX IFAC Triennial World Congress*, Budapest, Hungary, 1984, pp. 119–125.

- [3] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Philadelphia, PA: SIAM, 1994.
- [4] S. Boyd and L. Vandenberghe, "Introduction to convex optimization with engineering applications," Lecture notes for EE392x, Stanford Univ., Stanford, CA, 1995.
- [5] F. De Carlo *et al.*, "Description and analysis of an algorithm for star identification, pointing, and tracking systems," *Opt. Eng.*, vol. 33, no. 8, pp. 238–2745, Aug. 1994.
- [6] L. El Ghaoui *et al.*, *LMITool: A front end for LMI optimization, user's guide*, Feb. 1995. Available via anonymous ftp to <ftp.ensta.fr>, under [/pub/elghaoui/lmitool](ftp://pub/elghaoui/lmitool).
- [7] O. D. Faugeras and M. Hebert, "A 3-D recognition and positioning algorithm using geometrical matching between primitive surfaces," in *Proc. Int. Joint Conf. Art. Intell.*, Karlsruhe, Germany, Aug. 1983, pp. 996–1002.
- [8] A. Fiacco and G. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. New York: Wiley, 1968. Reprinted 1990 in the SIAM Classics in Applied Mathematics series.
- [9] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ., 1995.
- [10] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. Opt. Soc. Amer. A*, vol. 4, no. 4, pp. 629–642, 1987.
- [11] B. K. P. Horn *et al.*, "Closed-form solution of absolute orientation using orthonormal matrices," *J. Opt. Soc. Amer. A*, vol. 5, no. 7, pp. 1127–1135, 1988.
- [12] T. S. Huang, S. D. Blostein, and E. A. Margerum, "Least-squares estimation of motion parameters from 3-D point correspondences," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, Miami Beach, FL, 1986, pp. 24–26.
- [13] T. S. Huang and A. N. Netravali, "Motion and structure from feature correspondences: A review," *Proc. IEEE*, vol. 82, no. 2, pp. 252–268, 1994.
- [14] S. Hutchinson *et al.*, "A tutorial on visual servo control," *IEEE Trans. Robot. Automat.*, vol. 12, pp. 651–670, Oct. 1996.
- [15] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 1984.
- [16] S. D. Morgera and P. L. C. Cheong, "Rigid body constrained noisy point pattern matching," *IEEE Trans. Image Processing*, vol. 4, May 1993.
- [17] R. T. Rockafellar, *Convex Analysis*. Princeton, NJ: Princeton Univ. Press, 1970.
- [18] J. U. Turner, "A feasibility space approach for automated tolerancing," *ASME J. Eng. Ind.*, vol. 115, pp. 341–346, Aug. 1993.
- [19] J. U. Turner and M. J. Wozny, "The M -space theory of tolerances," in *Proc. 1990 ASME Design Automat. Conf.*, Chicago, IL, Sept. 1990.
- [20] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 376–380, 1991.
- [21] Y. L. Gu, "An exploration of orientation representation by lie algebra for robotic applications," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 243–248, 1990.
- [22] J. S.-C. Yuan, "A general photogrammetric method for determining object position and orientation," *IEEE Trans. Robot. Automat.*, vol. 5, Apr. 1989.