Kalyna Apkarian
Seminar Presentation: Critical Review
4/19/16

The paper entitled "A Constrained Optimization Approach to Virtual Fixtures" by Ming Li, Ankur Kapoor and Russell Taylor presents a weighted, linearized, multi-objective optimization framework for virtual fixtures[i]. Virtual fixtures are used to augment motion commands from the user, thus enhancing precision, stability, and patient safety. These virtual fixtures can be implemented given information on the instantaneous kinematics of the robot and the geometric constraints specified based on the desired behavior of the system. The authors present a set of basic geometric constrains and their implementation for sample tasks, and also provide experimental results. The control algorithm presented is useful to generate motion constraints for applications in orthopedic surgery, laparoscopic procedures, and more.

This paper has proven to be very relevant to formulating a constrained optimization problem for use in Synthetic Tracked Aperture Ultrasound (STRATUS) imaging using the UR5 robot. The goal of this project is to use the UR5 to guide a sonographer to scan a specific trajectory for a higher quality ultrasound image. Two important geometric constraints from this paper will be highlighted in detail in this review due to their direct applicability to the project at hand, followed by experimental results and conclusions.

The constrained control algorithm is presented as follows:

$$\underset{\frac{\Delta\vec{q}}{\Delta t}}{\operatorname{argmin}} \left\| W \left(\frac{\Delta\vec{x}}{\Delta t} - \frac{\Delta\overrightarrow{x_d}}{\Delta t}\right) \right\|$$

$$s.t. \quad H\frac{\Delta\vec{x}}{\Delta t} \geq \vec{h}$$

$$\frac{\Delta\vec{x}}{\Delta t} = J\,\frac{\Delta\vec{q}}{\Delta t}$$

Where $\Delta\vec{x}$ and $\Delta\vec{x}_d$ are the computed and desired incremental end effector motion, $\Delta\vec{q}$ is the desired incremental joint motion, Δt is the small time interval of the loop, J is the Jacobian matrix, and W is a diagonal weighting matrix. H and $\vec{h}$ are derived from the geometric constraints, and can be written as just one constraint or a combination of several constraints.

The paper presents the formulation of five basic geometric constraints: stay at a point, maintain a direction, move along a line, rotate around a line, and a plane related case. The "move along a line" formulation has proven to be particularly useful to the STRATUS system and is derived as follows.

The constraint as desired is to move along a line, $L = \overrightarrow{L_0} + \hat{l} * s$, where $\overrightarrow{L_0}$ is a point on the line and $\hat{l}$ is the unit vector indicating the direction of the line, as can be seen in Figure 1. With each incremental motion, P_{cl}, the closest point on the line L to the current position $\vec{x}_p$ is computed. The signed errors are then calculated according to $\overrightarrow{\delta_p} = \vec{x}_p - P_{cl}$ and $\delta_r = 0$, in this case, since there is no rotation. The vector $\overrightarrow{\delta_p} + \Delta\vec{x}_p$ is then projected onto the plane perpendicular to the line L according to the following method. A rotation matrix is computed to transform the plane to the world frame. The rotation matrix has the form

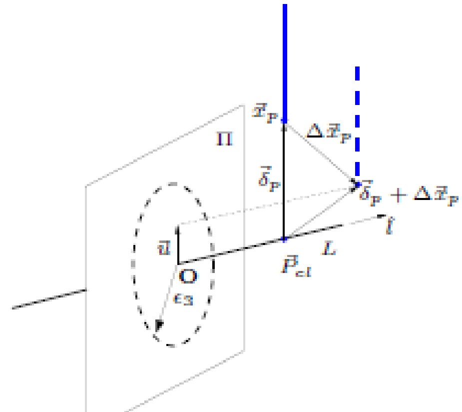

Figure 1. Move along a line

$$R = [\widehat{v_1} \quad \widehat{v_2} \quad \hat{l}], \text{ where } \widehat{v_1} = \frac{\hat{l} \ x \ \hat{l}\prime}{\|\hat{l} \ x \ \hat{l}\prime\|} \text{ and } \widehat{v_2} = \frac{\hat{l} \ x \ \widehat{v_1}}{\|\hat{l} \ x \ \widehat{v_1}\|}$$

where $\hat{l}'$ is any arbitrary unit vector not in the same direction as $\hat{l}$. This rotation matrix allows a vector to be written in the world frame by pre multiplying it by $(R * [\cos(\alpha_i); \quad \sin(\alpha_i); \quad 0])^T$. Next, we require the projection of $\overrightarrow{\delta_p} + \Delta\vec{x}_p$ to be less than some error, ε, approximated by an n-dimensional polygon. This constraint is then written as:

$[(R * [\cos(\alpha_i); \quad \sin(\alpha_i); \quad 0])^T \quad 0 \quad 0 \quad 0] \cdot \left(\overrightarrow{\delta_p} + \Delta\vec{x}\right) \leq \varepsilon$ for i=1:n, where $\alpha_i = \frac{2*\pi*i}{n}$.

Rewriting in the desired form of the general problem, $H\frac{\Delta\vec{x}}{\Delta t} \geq \vec{h}$

$$H = \begin{bmatrix} (-R * [\cos(\alpha_1); \quad \sin(\alpha_1); \quad 0])^T & 0 & 0 & 0 \\ & \vdots & & \\ (-R * [\cos(\alpha_n); \quad \sin(\alpha_n); \quad 0])^T & 0 & 0 & 0 \end{bmatrix}$$

$$\vec{h} = \begin{bmatrix} \varepsilon \\ \vdots \\ \varepsilon \end{bmatrix} - H * \vec{\delta}$$

This is just one of the geometric constraints given in the paper; the others are very similar and intuitive to understand and implement after understanding the material above. For example, the plane related case is an extension of "move along a line." It can be used to either prevent penetration of a certain plane or restrict motion to within a plane. For STRATUS, we use it to constrain motion along a plane when force constraints will be introduced. It follows the same form of "move along a line," where we instead look for the normal direction to the plane, $\hat{d}^t$, instead of the unit vector direction of the line. The closest point on the plane is similarly found, and then the problem takes the form

$$H = \begin{bmatrix} \hat{d}^t & 0 & 0 & 0 \\ -\hat{d}^t & 0 & 0 & 0 \end{bmatrix}$$

$$\vec{h} = \begin{bmatrix} 0 \\ -\varepsilon \end{bmatrix} - H * \vec{\delta}$$

The authors then go on to describe their experimental results from two sample tasks using the JHU Steady-Hand Robot. The first formulates a virtual fixture with two constraints to follow a curve with a fixed tool orientation with respect to the curve. Authors drew line segments on a plastic plate, digitized to gather sample points, and generated a 5th degree b-spline in the target coordinate frame. Constraints were generated in two task frames: tool tip and tool shaft, which was translated 100 mm in z from the tool tip frame. The tool tip constraint was generated in the manner described above to move along a line, following the tangent direction of the b-spline. The tool shaft constraint was similarly done in the manner above, and constrained to be perpendicular to the plane by constraining the origin of the frame to move along the b-spline curve. This resulted in an optimization problem of the form

$$\underset{\Delta\vec{q}}{\operatorname{argmin}} \left\| W \, (J\Delta\vec{q} - k\vec{f}) \right\|$$

$$s.t. \quad \begin{bmatrix} H_t & 0 \\ 0 & H_s \end{bmatrix} \begin{bmatrix} J_t \\ J_s \end{bmatrix} \Delta\vec{q} \geq \begin{bmatrix} \overrightarrow{h_t} \\ \overrightarrow{h_s} \end{bmatrix}$$

The error was computed as the distance from the actual tool tip position to the reference b-spline curve, as measured by an optical tracker and LEDs. Authors reported the average error of five trials to be 0.32 +- .19mm, where most of the error occurred at sharp turns where the tangent direction changed dramatically. Communication delays between the optical tracker and the robot also played a large role in the error. A second task experiment involved a virtual remote center of motion, where the tool tip is constrained to rotate around a predefined axis at a fixed angle. It follows the geometric constraint rotate around a line, which was not described in this review.

While the experiments were well thought out and clearly validated, it would have been interesting to see the results of varying some of the parameters of the geometric constraints. For example, the authors always used $n = 8$ to approximate a circle of radius $\varepsilon = 0.001$. They describe this as “stiff motion constraints” but it would have been interesting to see how changing these parameters affects the behavior of the system. Additionally, there is always a weighting matrix present in the formulation of the optimization problem, but its significance was never explained nor was it used in the experiments. Regardless, this paper has been a fundamental source in the implementation of a virtual fixture in the STRATUS system due to its applicability, clarity, and ease of implementation.

The STRATUS project is just one of the many of applications of the control algorithm presented in this paper. In particular, the formulation of the constraints is clear, making it easy to implement. There were a few typos in the mathematical derivation of the constraints, but overall it was not difficult to figure out what the correction was. Aside from these minor errors, implementing these constraints in our system was fairly straightforward, and has given us good results thus far. While the paper described simple sample tasks for surgical tools, the successful implementation of this method in the STRATUS system demonstrates future potential for virtual fixtures in ultrasound imaging systems.

---

[i] M. Li, A. Kapoor, and R. H. Taylor, “A constrained optimization approach to virtual fixtures,” in IROS, 2005, pp. 1408–1413.