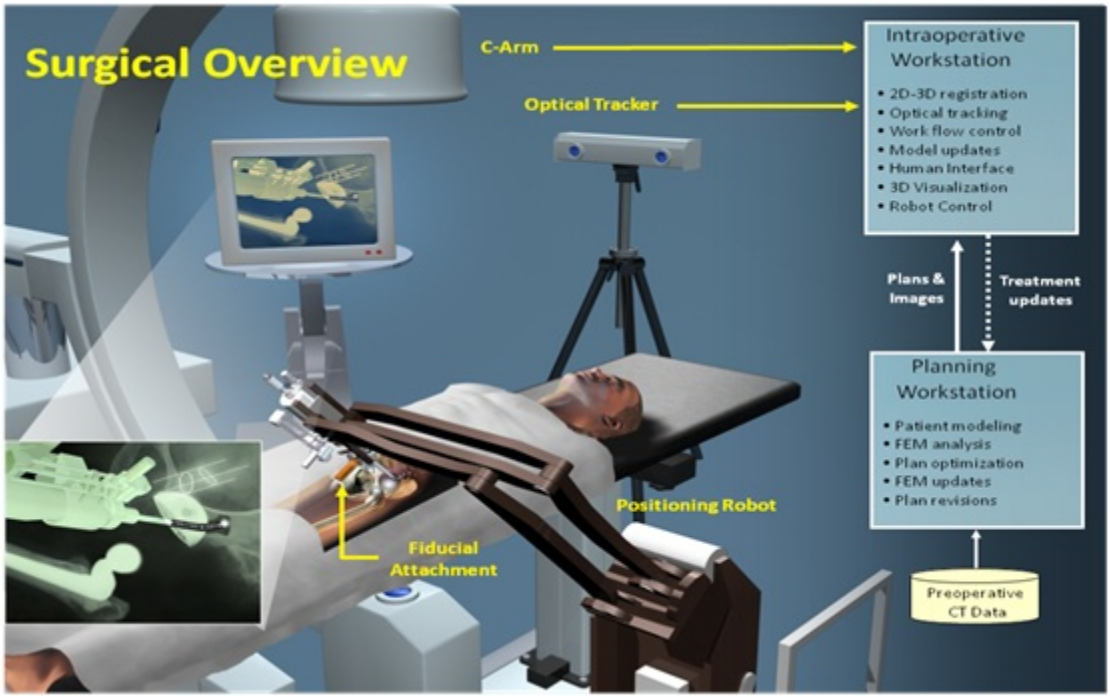
Position Control of BIGSS Lab Snake for Total Hip Arthroplasty (THA) Surgery

600.446 Computer Integrated Surgery II

Paper Summary



Project 6:

Farshid Alambeigi

Mentors:

Dr. Mehran Armand, Ryan Murphy, Dr. Russell Taylor

Spring 2014

In this report I want to briefly describe 3 papers which I have used them in my work. These papers are:

1. Ankur Kapoor, Ming Li and Russell H. Taylor, "*Constrained Control for Surgical Assistant Robots*", Proceedings of the 2006 IEEE International Conference on Robotics and Automation, Orlando, Florida - May 2006

2. Ankur Kapoor , Nabil Simaan , Russell H. Taylor, "*Suturing in Confined Spaces: Constrained Motion Control of a Hybrid 8-DoF Robot*" , Proceedings of the 12th International Conference on Advanced Robotics, 2005

3. Ryan J. Murphy, Yoshito Otake, Russell H. Taylor, and Mehran Armand, "*Predicting Kinematic Configuration from String Length for a Snake-like Manipulator Not Exhibiting Constant Curvature Bending*", submitted in Intelligent Robots and Systems (IROS), 2014.

First paper presents an approach to implement virtual fixtures for surgical robot assistants. This paper uses a weighted, multi-objective constrained optimization framework to formalize a library of virtual fixtures for task primitives.

Second paper uses the first paper as a method to control a tele-robotic system for minimally invasive surgery of the throat and upper airways. This system has been combined from a six DoF robot and a two DoF snake-like unit.

Third paper introduces an experimental approach to define a kinematic model for continuum robots. This two-step model predicts the tip position and kinematic configuration directly from string length with no assumptions regarding constant curvature bending.

I have chosen these papers because in my project I have used the optimization approach and virtual fixture library of first paper. Also, I have used the general approach of second paper because of its similarity to my work. Finally I have used the kinematic model of third paper to implement the optimization algorithm.

For each paper I briefly explain the abstract, method and results of each paper.

1. **Ankur Kapoor, Ming Li and Russell H. Taylor, "*Constrained Control for Surgical Assistant Robots*", Proceedings of the 2006 IEEE International Conference on Robotics and Automation, Orlando, Florida - May 2006**

Abstract and motivation:

This paper introduces weighted constrained optimization framework to formalize a library of "virtual fixtures" for surgical robot assistants. The virtual fixtures can be customized for a particular surgical task by combining one or more objectives and constraints assigned to single or multiple task frames. Also this paper investigates the effect of approximating nonlinear constraints with linear ones. Moreover, it applies the "soft" virtual fixture to have some resistance inside safety regions and no resistance in preferred regions.
Often in surgical operations where a robot is used, we want to impose some restrictions on the robot motion, e.g. prevent the tooltip from entering some undesired region or confine the tool shaft to pass through some fixed point in space (incision point etc.). In this case, the goal is to obey the constraints and at the same time we want the tooltip to follow the motions of the master robot as close as possible. This can be formed as some constrained optimization problem where the objective function that should be minimized is the difference between the desired and robot tip motions.

Method:

This paper divides control algorithms to these steps:

Calculate actual position of the robot using forward kinematics:

$$Actual\ Position = forward\ kinematics\ (\theta_{Old})$$

1. Calculate desired incremental motion in Cartesian space Δ_{pos} :

   $$\Delta_{pos} = Actual\ Position - Desired\ Position$$

2. Consider Δt as a small time increment and using below linear relations to approximate the incremental motion in Cartesian space Δx : $\dot{x} = J \cdot \dot{q} \rightarrow \frac{\Delta x}{\Delta t} = J \cdot \frac{\Delta q}{\Delta t} \rightarrow \Delta x = J \cdot \Delta q$

3. Solve this constrained optimization problem using known numerical methods which minimize the two-norm of error between desired and actual incremental motions with minimum joint motions:

   $$\Delta q = \arg\min_{\Delta q}(\left\|\Delta x - \Delta_{pos}\right\|_2^2 + \left\|w.\Delta q\right\|_2^2); \Delta q \in \mathbb{R}^{n\times 1}$$
   $$s.t.\ \ A.\Delta x \le b$$

   Where Δq is desired incremental motions of the n-DOF of the robot, w is a diagonal matrix for weights. A and b matrices define the constraints of the problem. We should mention that some of the constraints in this problem can be nonlinear.

4. Calculate new vector q which defines n-DOF variables: $q_{New} = q_{Old} + \Delta q$

Therefore For solving this problem we need:

- ✓ Forward Kinematics of robot
- ✓ Jacobian matrix of robot
- ✓ Finding the A and b Matrices which are defining our constraints (RCM constraint+ Limitation on cable length and joint angles)

Defining matrix A and b for constrained optimization problem:

For determining matrix A and b this paper defines virtual fixtures for five task primitives. It considers the robot task frame as a purely kinematic Cartesian device with the tool position $x_p \in \mathbb{R}^3$ and the tool orientation given by unit vector $l_t \in \mathbb{R}^3$.
The names and descriptions of these task primitives are:

1) **Stay on a point:** Keep the tool position x_p on the reference position x_0.

2) **Maintain a direction**: Keep the tool orientation l_t aligned with the reference direction l_r.

3) **Move along a line:** Keep the tool position x_p on line L which has the direction l_r and passes through point x_0. At the same time, the tool should move along L proportional to the users input τ.

4) **Rotate around a line:** Keep the tool orientation l_t perpendicular to line L which has the direction l_r and passes through point x_0. At the same time, the tool should rotate around L proportional to the users input τ.

Task Primitives	Nominal Error Term	Constraints
Stay on a point	$\delta_p = x_p - x_0$	$\frac{1}{2}\|\delta_p\|_2^2 \leq \epsilon_m$
Maintain a direction	$\delta_r = \hat{l}_t \times \hat{l}_r$	$\frac{1}{2}\|\delta_r\|_2^2 \leq \epsilon_m$
Move along a line	$\delta_p = R_p^T[x_p - x_0 - [(x_p - x_0)\cdot \hat{l}_r]\hat{l}_r]$	$\|\delta_p(1)^2 + \delta_p(2)^2\|_2^2 \leq \epsilon_m$
Rotate around a line	$\delta_r = R_r^T \times [\frac{\hat{l}_t - \hat{l}_r(\hat{l}_t \cdot \hat{l}_r)}{\|\hat{l}_t - \hat{l}_r(\hat{l}_t \cdot \hat{l}_r)\|_2} \times \hat{l}_t]$	$\|\delta_r(1)^2 + \delta_r(2)^2\|_2 \leq \epsilon_m$
Stay above a plane	$\hat{d}^T \cdot \frac{x_p - x_0}{\|x_p - x_0\|_2}$	$\hat{d}^T \cdot (x_p - x_0) \geq \epsilon_m$

Table 1. The nominal error terms and constraints for five primitive tasks[1].

The nominal error terms and constraints for five tasks are summarized in Table 1. In this table, R_p and R_r are rotation matrices that would transform a plane perpendicular to $\boldsymbol{l}_p$ and $\boldsymbol{l}_r$ respectively to world coordinates. The maximum allowed error is denoted by ε_m which is a small positive number. $\boldsymbol{\delta}(i)$ represents the i^{th} component of vector $\boldsymbol{\delta}$.

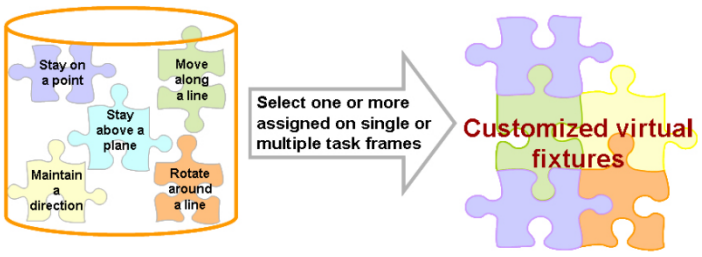


Figure 1, Customized virtual fixture[1]

According to Figure 1, using this library each customized virtual fixtures for complicated surgical tasks can be treated as the combination of one or more objects assigned on single or multiple task frames.

Investigating the effect of linearizing nonlinear constraints:

The constraints for the task primitives are often nonlinear; nevertheless we can use linear expressions to approximate the constraints. Solving linearly constrained least squares problems can take less computation time and computation for a linear constrained quadratic optimization problem is efficient and robust. This paper uses a set of hyperplanes to bound a polyhedron to approximate a geometric constraint region like spherical error tolerance region. It is obvious that as the number of the hyperplanes increases, the volume of the polyhedron reduces and the polyhedron approaches the inscribed sphere.

Figure 2 shows the relation of polyhedron defined by $Ax \leq b$ with different numbers of hyperplanes and the specified spherical error tolerance region.

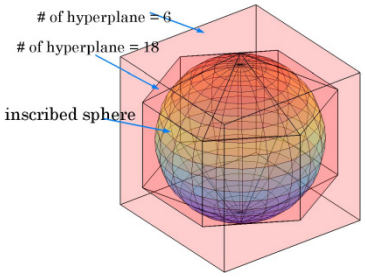


Figure 2, the polyhedron determined by $Ax \leq \boldsymbol{b}$ with different numbers of hyperplanes. The inscribed sphere defines the ideal error tolerance region[1].

Some experiments have been done to compare this approximation (Figure 3). Results show that there is a trade-off between accuracy and speed between linear and nonlinear constraints. One can choose to use a linear approximation with fewer numbers of hyperplanes if accuracy is not a concern, whereas nonlinear gives better accuracy..

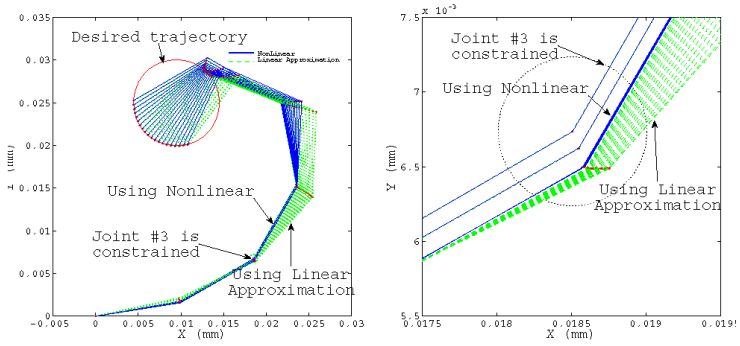


Figure 3, (a) A example trajectory of a planar 6 link robot for showing the difference between linear approximation and nonlinear constraints(b) Close-up of part (a)showing difference between linear and nonlinear approximation[1.]

2. **Ankur Kapoor , Nabil Simaan , Russell H. Taylor, "*Suturing in Confined Spaces: Constrained Motion Control of a Hybrid 8-DoF Robot*" , Proceedings of the 12th International Conference on Advanced Robotics, 2005**

Abstract and motivation:

This paper presents the kinematic modeling and high level control of a hybrid 8 DoF robots used for dexterous applications such as suturing for minimally invasive surgery of the throat and upper airways in confined spaces. The high level control is based on a linearized multi-objective constrained optimization method that described in paper 1.

Method:

For this work, they have attached a snake-like unit (SLU) to a modified version of the LARS - a 6-DoF robot developed at IBM. The robots are shown in Figure 4. Lars robot is composed from a 3 DoF X-Y-Z stage that is serially attached to a "Remote Center of Motion" (RCM) mechanism. This mechanism is designed to rotate the tool tip around a fixed point in space. They have introduced the kinematic model of coupled robot such that they could be able to use the optimization approach.
According to paper 1, for solving the optimization problem forward kinematics and Jacobian matrix of coupled robots are needed. Also they needed to define suturing task as matrix A and b.

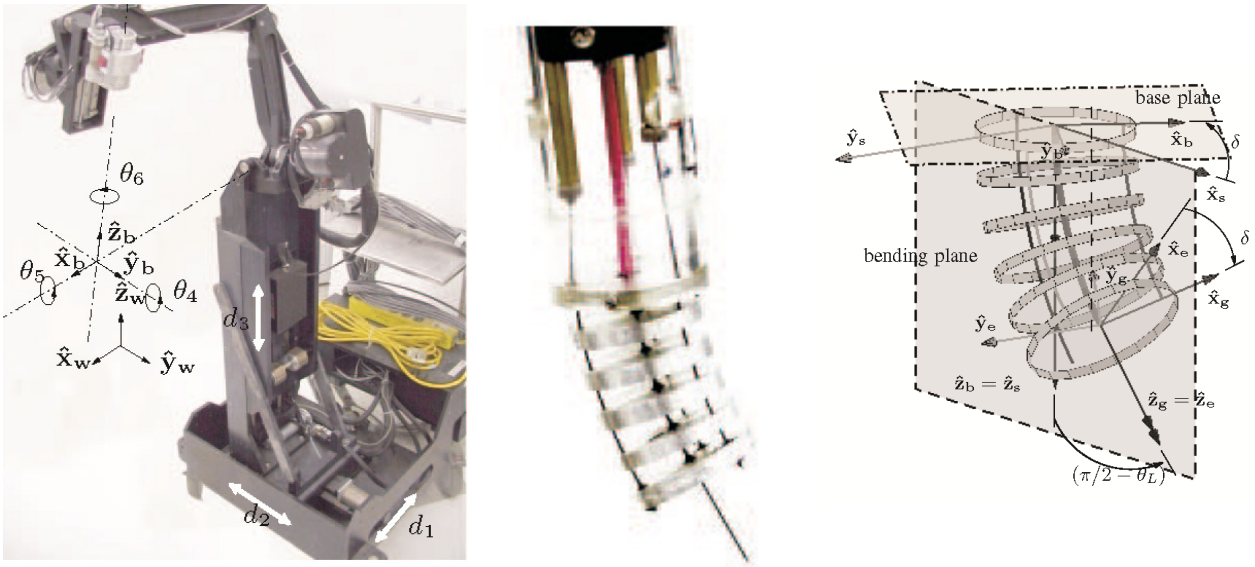


Figure 4, Lars robot(Left) and Snae Like Unit (SLU)[2]

a) Kinematic Model

The generalized twist of the LARS is related to the joint velocities of it according to:

$$\dot{x}^{W}_{slu\ base} = J_{Lars} \cdot \dot{q}_{Lars}$$

Also they have used this equation for determining Jacobian matrix of SLU:

$$V^{w}_{SLU} = V^{w}_{SLU\,base} + R^{w}_{SLU} . V^{SLUbase}_{SLU\,tip} + \omega^{w}_{SLU\,base} \times R^{w}_{SLU\,base} . p^{SLU\,base}_{SLU\,tip}$$

$$\omega^{w}_{SLUtip} = \omega^{w}_{SLU\,base} + R^{w}_{SLU} . \omega^{SLUbase}_{SLU\,tip}$$

Where $R^{w}_{SLU\,base}$ is the rotation matrix from snake base to world coordinate, and $p^{Snakebase}_{Snaketip}$ is the position of SLU related to its base. V^{w}_{SLUtip} is the tip velocity of the SLU in world coordinate.

Therefore, the kinematics of hybrid system consisting of the 6-DoF LARS and 2-DoF SLU can be described using 8 independent variables. For convenience they have defined an augmented state vector s ∈ R8 as composed of the joint variable of LARS and two angles describing the configuration of the SLU, that is:

$$J_{combined} = \begin{bmatrix} J_{Lars} & J_{SLU} \end{bmatrix}; J_{Lars} \in \mathbb{R}^{6\times 6}, J_{Snake} \in \mathbb{R}^{6\times 2}$$
$$\dot{x}^{W}_{SLU} = J_{combined} \cdot \dot{s};\ J_{combined} \in \mathbb{R}^{6\times 8}, \dot{s} \in \mathbb{R}^{8\times 1}$$

b) <u>Defining Constraints and Objective Functions:</u>

1) *<u>Minimizing tissue tear</u>*: To ensure pure rotation about the center of the suture, They rotate the gripper such that its angular velocity vector is perpendicular to the suture plane and the center point of the suture is constrained to lie within a small sphere of radius ε_g . This translates into the constrained optimization problem given by:

$$\arg\min_{\dot{x}}(\left\| W_g(\dot{x}^{w}_{g} - \dot{x}^{w}_{gdesired}) \right\|_2$$
$$s.t.\ \left\| p^{w}_{g} - p^{w}_{gstart} \right\| and\ \dot{x}^{w}_{gdesired} = (0,0,0,0,0,\omega_d)^t$$

Where in this equation w_d is the desired angular speed about the center of suture and Wg is a diagonal matrix of weights.

2) *<u>Avoiding joint speed limits</u>:* for considering limits on the joint velocities of SLU they have considered this constraint:

$$\arg\min_{\dot{x}}(\left\| W_g(\dot{x}^{w}_{g} - \dot{x}^{w}_{gdesired}) \right\|_2$$
$$s.t.\ H_t \cdot \dot{q}_{SLU} \geq h_t; H_t = (\mathrm{I}, -\mathrm{I})^t \in \mathbb{R}^{6\times 3}$$
$$and\ h_t = (\dot{q}_{SLU\,lo}, -\dot{q}_{SLU\,up})^t \in \mathbb{R}^6$$

where $\dot{q}_{SLU\,lo}\ and\ \dot{q}_{SLU\,up}$ are lower and upper limits for joint velocities attainable by the SLU secondary backbones respectively.

3) *<u>Avoiding joint limits</u>:* To ensure that the motion is within the workspace of the system given by $s_{lo}\ and\ s_{up}$ with minimum extraneous motion of the system they have added an objective function with weight W_s. Therefore this constraint and objective can be written as:

$$\arg\min_{\dot{s}}(\left\| W_s \cdot \dot{s} \right\|_2$$
$$s.t.\ H_s \cdot \dot{s} \geq h_s; H_s = (\mathrm{I}, -\mathrm{I})^t \in \mathbb{R}^{16\times 8}$$
$$and\ h_s = (s_L / \Delta\mathrm{t}, s_U / \Delta\mathrm{t})^t \in \mathbb{R}^{16}$$

After defining all of these constraints and objective functions they have summarized them as:

$$\arg\min_{\dot{s}}\left(\left\|\begin{bmatrix} W_g & 0 \\ 0 & W_s \end{bmatrix}_s \left(\begin{bmatrix} J_{SLU} \\ I \end{bmatrix}\cdot\dot{s} - \begin{bmatrix} \dot{x}^{w}_{gdesired} \\ 0 \end{bmatrix}\right)\right\|_2\right)$$

$$s.t.\begin{bmatrix} H_g & 0 & 0 \\ 0 & H_t & 0 \\ 0 & 0 & H_s \end{bmatrix}\begin{bmatrix} J_{SLU} \\ \begin{bmatrix} 0 & J_{J_{SLU}} \end{bmatrix} \\ I \end{bmatrix}\dot{s} \geq \begin{bmatrix} h_g \\ h_t \\ h_s \end{bmatrix}$$

Simulation and Results:

After defining the optimization problem they have simulated their work. Figure 5 shows the simulations comparing suturing using the SLU verses a rigid tool to hold the suture. These results clearly indicate the importance of maintaining tool tip dexterity to avoid large motions in the proximal end of the tools. This is a crucial requirement for suturing in confined spaces, such as MIS of the throat.

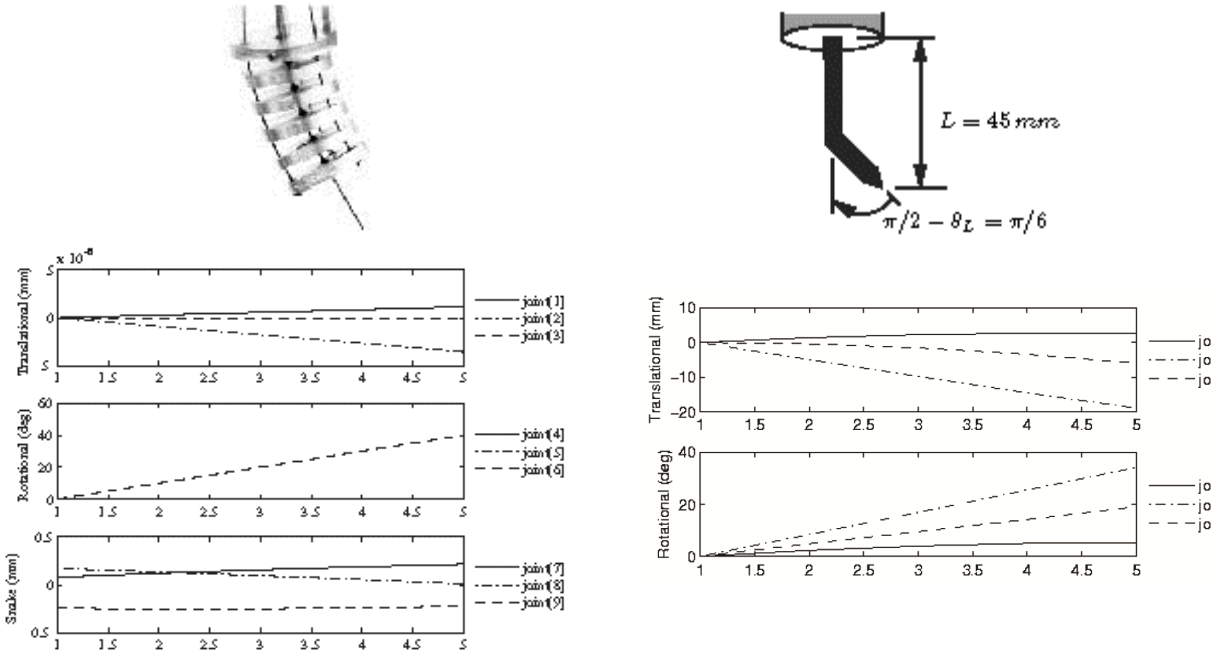


Figure 5 Lars robot(Left) and Snae Like Unit (SLU)[2]

These simulations have been validated by experiments based on encoder readings and the forward kinematic model of the hybrid robot using a tracking camera that determined the efficacy of the high-level controller.

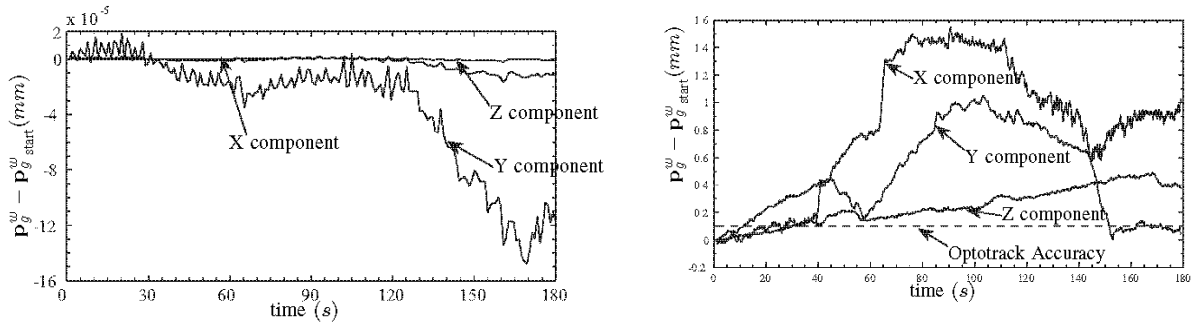


Figure 6, X, Y, and Z components using motor encoder readings(Left), and measured by Optotrak.(Right)[2].

3. **Ryan J. Murphy, Yoshito Otake, Russell H. Taylor, and Mehran Armand, "*Predicting Kinematic Configuration from String Length for a Snake-like Manipulator Not Exhibiting Constant Curvature Bending*", submitted in Intelligent Robots and Systems (IROS), 2014.**

Abstract:

This paper uses BIGSS lab snake and tries to find its kinematics model using a series of experiments. This snake does not have a constant curvature and therefore general models of continuum robots do not work for it. They have presented a model developed from a training data set to predict the full manipulator configuration from string length. The two-step kinematic model first predicts the tip position p_{eef} from string length using a 7th order Bezier polynomial to predict x position end-effector and a sum of three sinusoids to predict z position. The second step pins the manipulator tip at the predicted position and runs an energy minimization to estimate the kinematic configuration. This model does not assume constant curvature and was validated through experimental test data.

MANIPULATOR OVERVIEW:

APL has designed a cable driven Snake –like Dexterous Manipulator (SDM) to achieve the osteolytic lesions behind the well-fixed cup during revision surgery without the removal of implants (less-invasive procedure) which has these specifications (Figure 7):

- Planar manipulator composed of superelastic nitinol
- 4mm open lumen for inserting different tools in the SDM
- Designed to fit in a hip implant (6mm OD)
- Independent solid stainless steel cables to actuate the manipulator

Figure 7 Specifications of BIGSS Lab SDM [3].

EXPERIMENTAL SETUP:

Paper has performed a series of experiments bending the manipulator freely without obstruction in air. Each test limited the tension measured by the load cells to 22.2N, defined a specific motor voltage, and set the frequency of data collection.
An acrylic stand housed the manipulator actuation unit and centered a camera above the manipulator (Figure8) and prior to each test, a motor calibration procedure defined the zero cable position.
Then, an automated, piecewise-rigid 2D/3D registration technique defined the kinematic configuration of the manipulator from each static image (Fig. 5). This technique has maximized the similarity between a manipulator projection and the recorded image using gradient information.

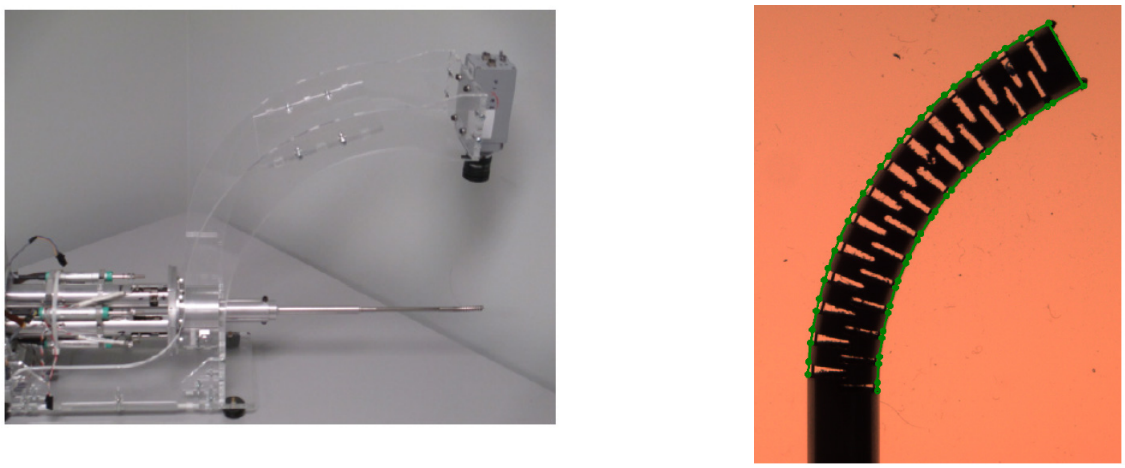


Figure 8, Test setup for investigating manipulator behavior[3].

KINEMATIC PREDICTION:

Using output of experimental tests the relation between cable length (l) and tip position (p) of the snake has been derived by curve fitting technique. They have used a nonlinear least-squares optimization to fit a linear combination of Bernstein basis polynomials (a B´ezier polynomial) to the data for determining x position. Then, they have defined z position as sum of sinusoids using a nonlinear least-squares optimization. Which this position is a function of x position:

$$p_x = f_1(\hat{l}) = B_n(\hat{l})$$
$$p_y = 0$$
$$p_z = f_2(p_x) = \sum_{i=1}^{3} a_i \sin(b_i.p_x + c_i)$$
$$p = \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} = \begin{pmatrix} f_1(\hat{l}) \\ 0 \\ f_2(f_1(\hat{l})) \end{pmatrix}$$

where p_x, p_y, and p_z are components of the SDM position, $\hat{l}$ is the normalized string length and B_n is an n^{th} order Bernstein polynomial. According to these equations p_z has been defined based on p_x and is equal to sum of three sinusoids. Also coefficients a_i, b_i, and c_i represent the fit parameters for the i^{th} sinusoid.

after predicting the position, the second step pins the manipulator tip at thised position and runs an energy minimization to estimate the kinematic configuration.

They have modeled ith pin joint as a torsional spring with some unknown spring constant ki. With assuming that the manipulator will settle in the least-energy state for a specific tip position they have solved this problem for finding snake configuration:

$$\tilde{\theta} = \arg\min \sum_{i=1}^{27} \theta_i^2$$
$$s.t. 0 = \left\| \tilde{p}_{eef} - p_{eef} \right\| and \left| \tilde{\theta}_i \right| \le 15^o$$

Where $\tilde{p}_{eef}$ is the tip position of the snake using forward kinematics model of the snake and $\tilde{\theta}$.

Simulation and results:

For validating their model they have done a series of error analysis. The forward kinematics from the predicted configuration $\tilde{\theta}$ were compared to the ground truth derived from the overhead images. We computed the magnitude error e_i for each predicted point $\tilde{p}_i$ as:

$$e_i = \left\| p_i - \tilde{p}_i \right\|$$

where p_i is the ground truth measured from the overhead image. According to Figure 9 Over 68% of the predictions resulted in a maximum error less than 1:25mm . This is approximately onefourth of the manipulator diameter. The largest errors occurred at the tip of the manipulator in high-bend configurations where the tip prediction performs poorly.

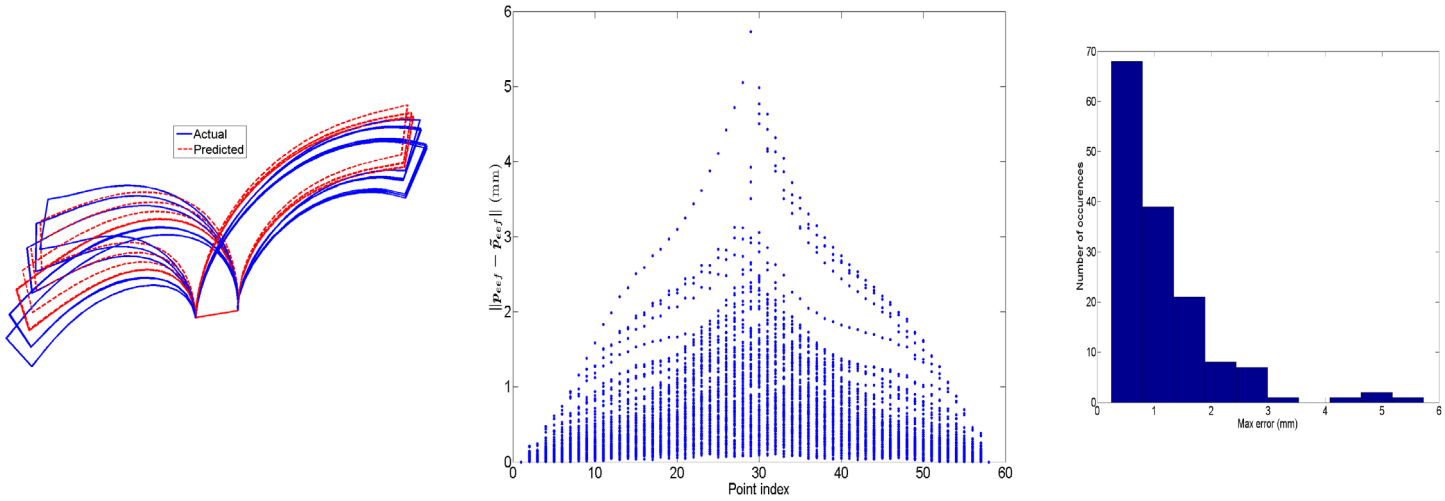


Figure 9. comparison between real and predicted configuration(Left), Errors along manipulator(Middle), Histogram of maximum Error (Right)[3].