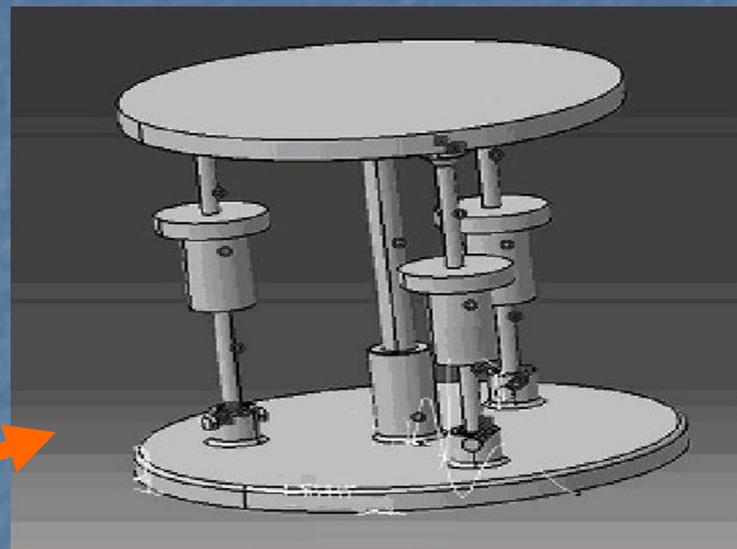
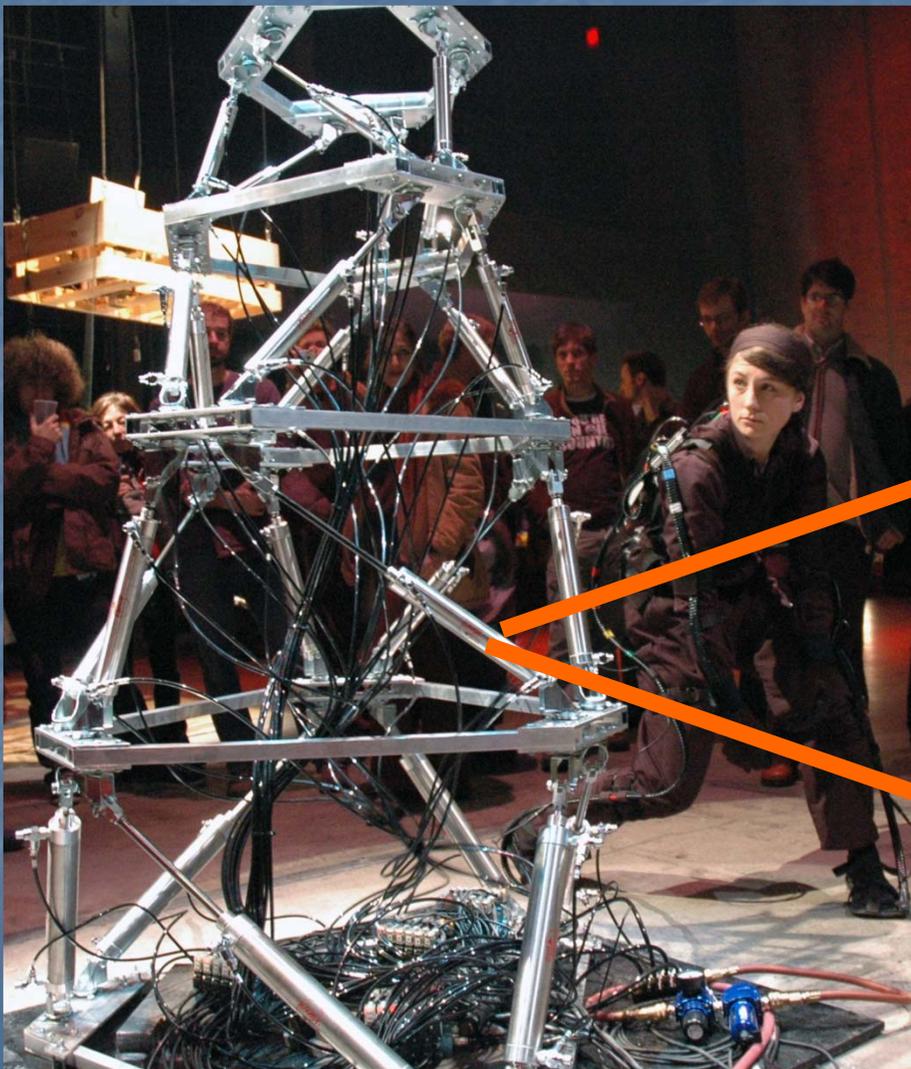


ОПТИМАЛЬНОЕ ПРОЕКТИРОВАНИЕ ДВИЖЕНИЯ СЕКЦИИ МАНИПУЛЯТОРА ТИПА ХОБОТ

*Антонов Д.А., Литун Т.О.
д.ф.-м.н., профессор Карпенко А.П.,
к.т.н., доцент Каганов Ю.Т.*

Примеры рассматриваемых манипуляторов



Постановка задачи

Координаты крепления
 i -го
шарнира к подвижной
платформе –
варьируемые
параметры

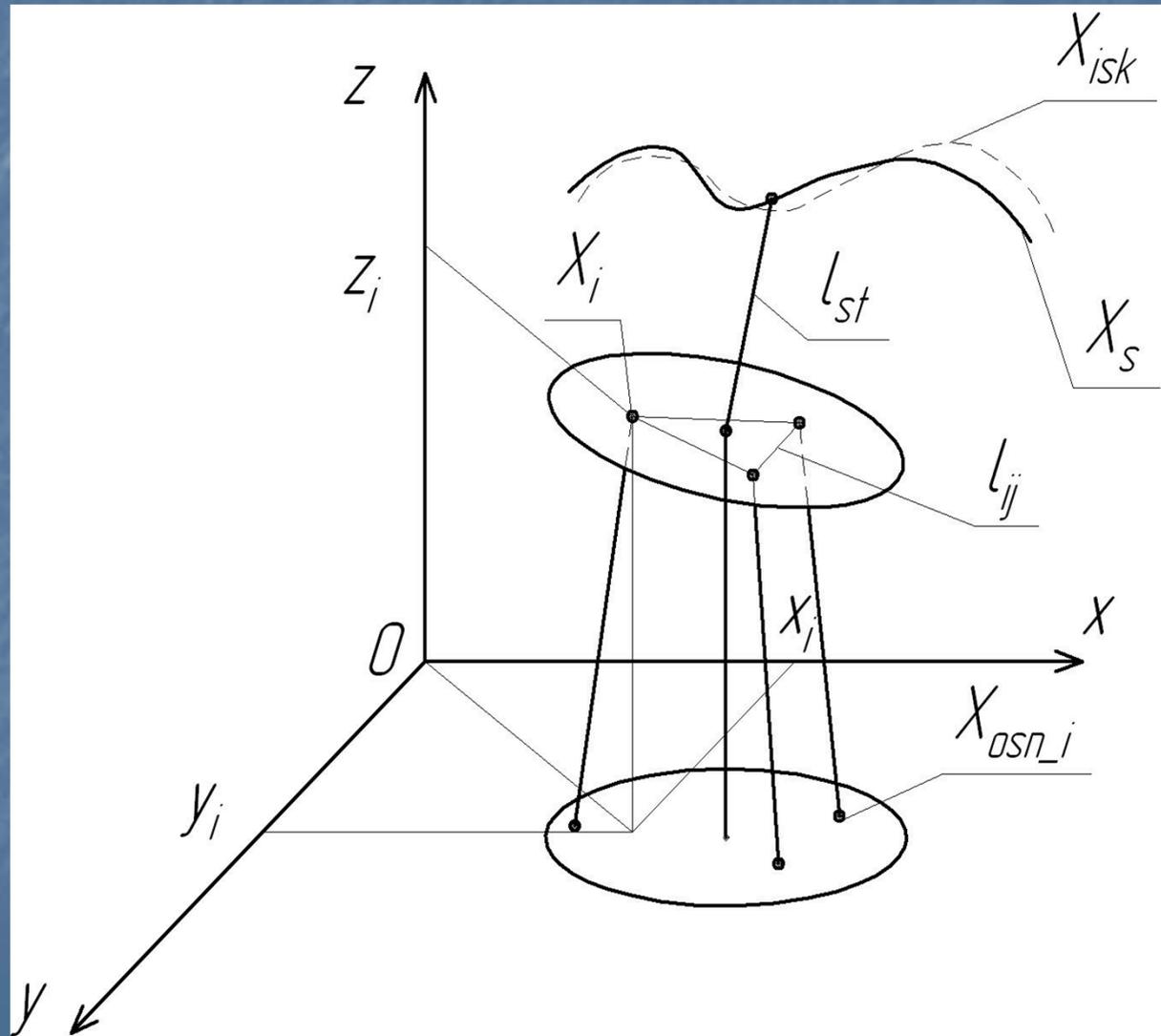
$$X_i = x_i, y_i, z_i;$$

$$i=1, 2, 3$$

Координаты
крепления i -го
шарнира к
неподвижной
платформе

$$x_{osn_i}, y_{osn_i}, z_{osn_i};$$

$$i=1, 2, 3$$



Постановка задачи

Расстояния между шарнирами на подвижной платформе

$$l_{1,2}, l_{2,3}, l_{1,3}$$

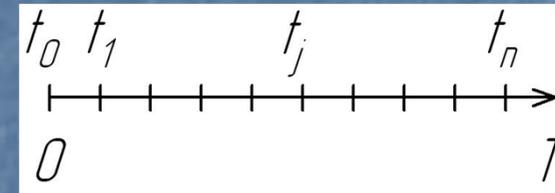
Длина схвата манипулятора l_{st}

Вектор нормали к плоскости подвижной платформы

$$V_n = (x_n, y_n, z_n)$$

Искомая и реальная(расчетная) траектория

$$X_{isk}(t_j), X_s(t_j)$$



Критерий оптимальности – евклидова норма отклонения реальной траектории схвата манипулятора от желаемой

$$W(X(t_j)) = \| X_{isk}(t_j) - X_s(t_j) \|^2; \quad X(t_j) \in D_x(t_j)$$

$$D_x = \{X(t_j) \mid c_i(X(t_j)) < 0, c_{eq_k}(X(t_j)) = 0, i \in [1:8], k \in [1:6]\}$$

Определение вектора нормали

Вектор нормали к подвижной платформе совместно с координатами её центра определяет положение рабочей точки манипулятора

Проекции нормали на оси неподвижной системы координат

$$x_n = (y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1)$$

$$y_n = -(x_2 - x_1)(z_3 - z_1) - (z_2 - z_1)(x_3 - x_1)$$

$$z_n = (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)$$

Направляющие косинусы

$$\cos(\varphi_x) = \frac{x_n}{\|V_n\|}$$

$$\cos(\varphi_y) = \frac{y_n}{\|V_n\|}$$

$$\cos(\varphi_z) = \frac{z_n}{\|V_n\|}$$

Определение координат схвата манипулятора

Положение системы координат подвижной платформы относительно базы задается матрицей перемещения

$$T_a = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & z_P \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

С учётом длины схвата манипулятора l_{st} координаты его рабочей точки в неподвижной системе координат, связанной с базой, равны

$$X_s = T_a l_{st} \left(\cos(\varphi_x) \quad \cos(\varphi_y) \quad \cos(\varphi_z) \right)^T$$

Функциональные и конструктивные ограничения

Фиксация шарниров на платформе описывается системой из шести уравнений вида:

$$seq_i = ((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 - l_{i,j}^2) = 0$$
$$i \in [1:4], j \in [1:4], i \neq j$$

Каждый актуатор может изменять свою длину в диапазоне

от P_{\min_i} до P_{\max_i}

Ограничение на длину актуатора в нормированном виде

$$c_i^r = \frac{\sqrt{(x_i - x_{osn_i})^2 + (y_i - y_{osn_i})^2 + (z_i - z_{osn_i})^2} - P_{\max_i}}{P_{\max_i} - P_{\min_i}} \quad i \in [1:4]$$

$$c_i^l = \frac{P_{\min_i} - \sqrt{(x_i - x_{osn_i})^2 + (y_i - y_{osn_i})^2 + (z_i - z_{osn_i})^2}}{P_{\max_i} - P_{\min_i}}$$

Задача безусловной оптимизации

Чтобы свести исходную условную задачу к безусловной используем метод штрафных функций, где штрафы можно записать с помощью непрерывной аппроксимации функции Хевисайда

$$H_i^c = \frac{1}{1 + \exp(-2kc_i)} \quad i \in [1:8]$$

Штрафная функция имеет следующий вид:

$$\begin{aligned} Q_i &= H_i \text{abs}(c_i) & i \in [1:8] \\ Q_j &= H_j \text{abs}(ceq_j) & j \in [1:12] \end{aligned} \quad Q(X) = \sum_i Q_i(X) + \sum_j Q_j(X)$$

Тогда целевая функция для задачи безусловной оптимизации будет иметь вид:

$$\Phi(X) = Q(X) + W(X)$$

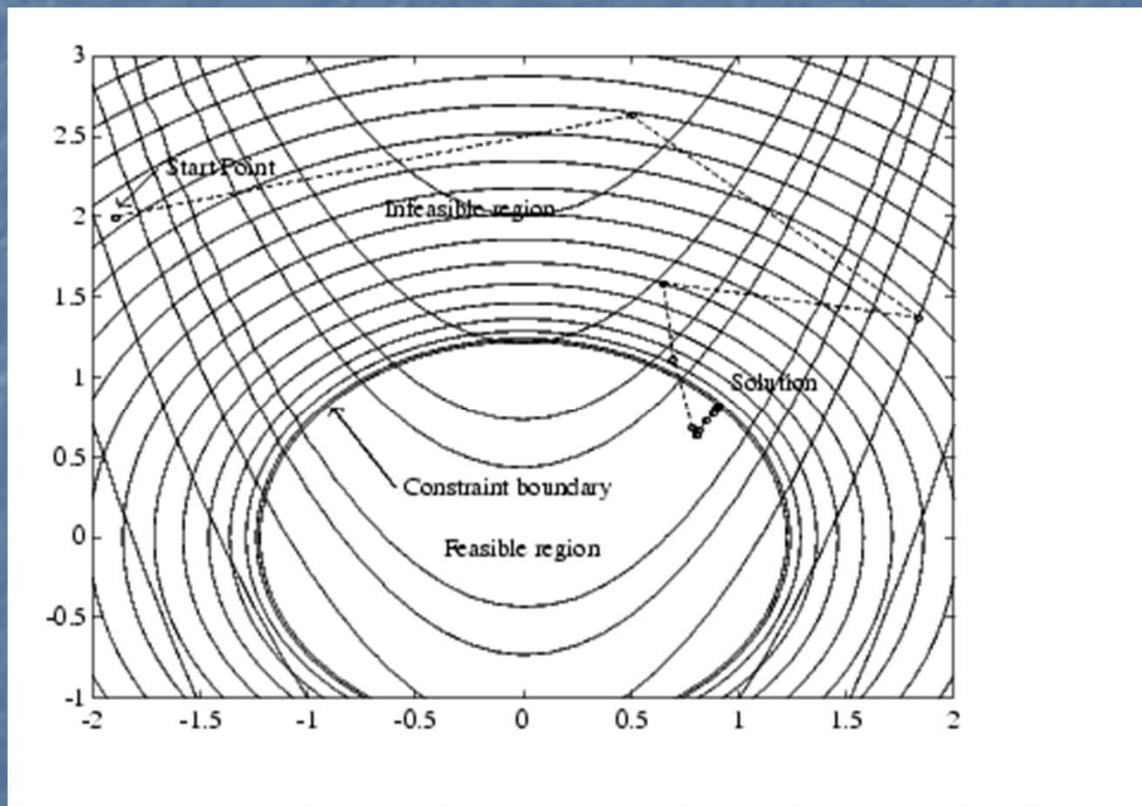
Конечным результатом решения задачи как условной, так и безусловной оптимизации является набор координат шарниров, соединяющих подвижную платформу с актуаторами.

Методы оптимизации

- *последовательное квадратичное программирование SQP*
- *квази-ньютоновский метод с модификацией BFGS*
- *метод Нелдера-Мида*

Последовательное квадратичное программирование SQP

В данном методе на каждой итерации решается подзадача Квадратичного Программирования (QP). Вид матрицы Гессе для функции Лагранжа обновляется на каждой итерации с помощью формулы BFGS.



Краткое описание метода SQP

На каждой основной итерации метода осуществляется аппроксимация гессиана для функций Лагранжа при помощи квази-ньютоновского модифицированного метода.

Основная идея постановки подзадачи QR заключается в квадратичной аппроксимации функции Лагранжа.

$$L(x, \lambda) = F(x) + \sum_{i=1}^m \lambda_i g_i(x)$$

Подзадача квадратичного программирования (QP)

$$\min_x \left(\frac{1}{2} x^T Hx + c^T x + b \right)$$

$$\nabla g_i(x_k)^T d + g_i(x_k) = 0$$

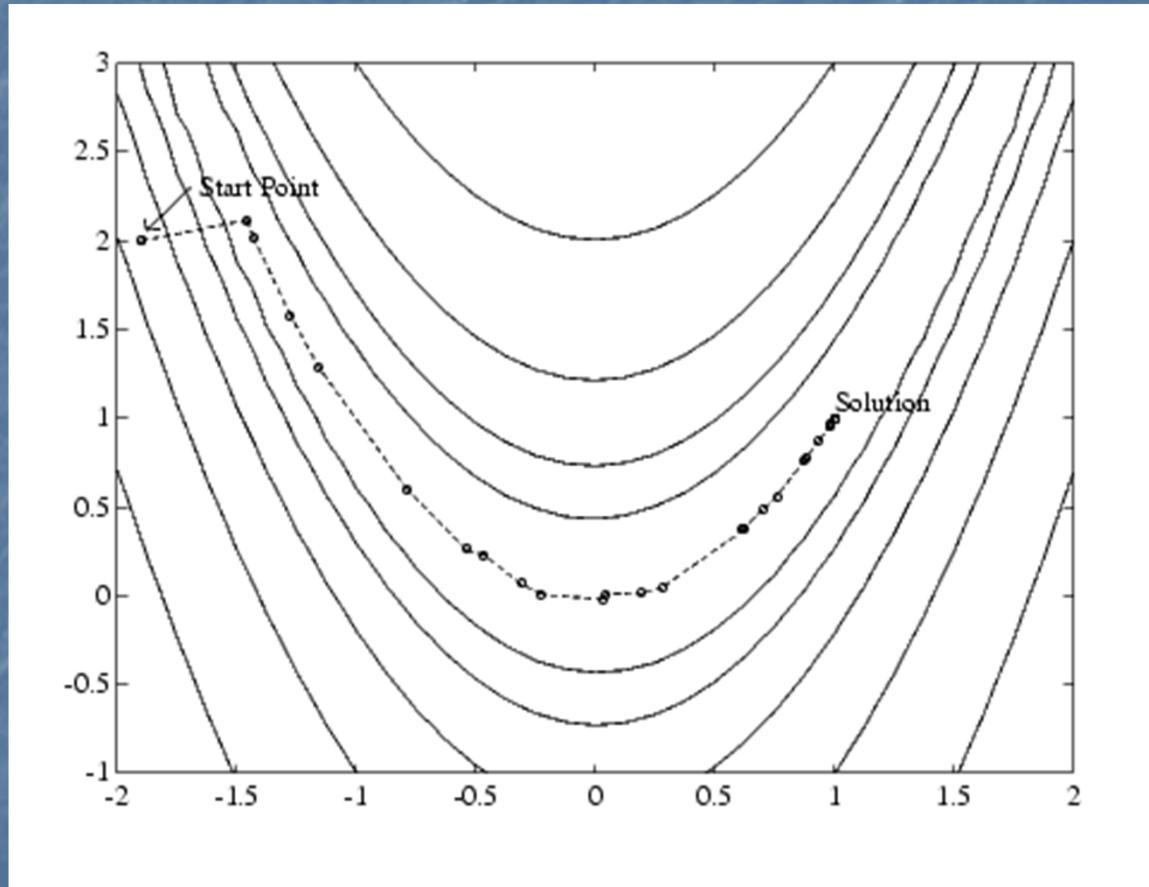
$$\nabla g_j(x_k)^T d + g_j(x_k) < 0$$

Где решение находится по итерационной формуле

$$x_{k+1} = x_k + \alpha_k d_k$$

Квази-ньютоновский метод с модификацией BFGS

Квази-ньютоновский метод со смешанной процедурой квадратичного и кубического поиска. Данный квази-ньютоновский метод использует формулу BFGS для модернизации приближенного значения матрицы Гессе.



Краткое описание метода BFGS

На каждой итерации метода решается задача локальной оптимизации вида:

$$\min_x \left(\frac{1}{2} x^T H x + c^T x + b \right)$$

Где H - симметричная и положительно определенная матрица второго порядка частных и смешанных производных (матрица Гессе, или Гессиан), c - постоянный вектор, b - константа.

Оптимальное решение приведенной задачи соответствует нулевым значениям первых производных, то есть

$$\nabla f(x^*) = Hx^* + c$$

Итерационная формула BFGS для модификации Гессиана

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k}$$

где $s_k = x_{k+1} - x_k$

$$q_k = \nabla f(x_{k+1}) - \nabla f(x_k)$$

Интерфейс ввода данных

INPUT_DATA

Pilot Pin #1

Length - min: max:

Origin Coordinates - End Coordinates -

X: X:

Y: Y:

Z: Z:

Pilot Pin #2

Length - min: max:

Origin Coordinates - End Coordinates -

X: X:

Y: Y:

Z: Z:

Pilot Pin #3

Length - min: max:

Origin Coordinates - End Coordinates -

X: X:

Y: Y:

Z: Z:

Core Pin

Length - min: max:

Motion Path

Set from File

source:

Set Manually

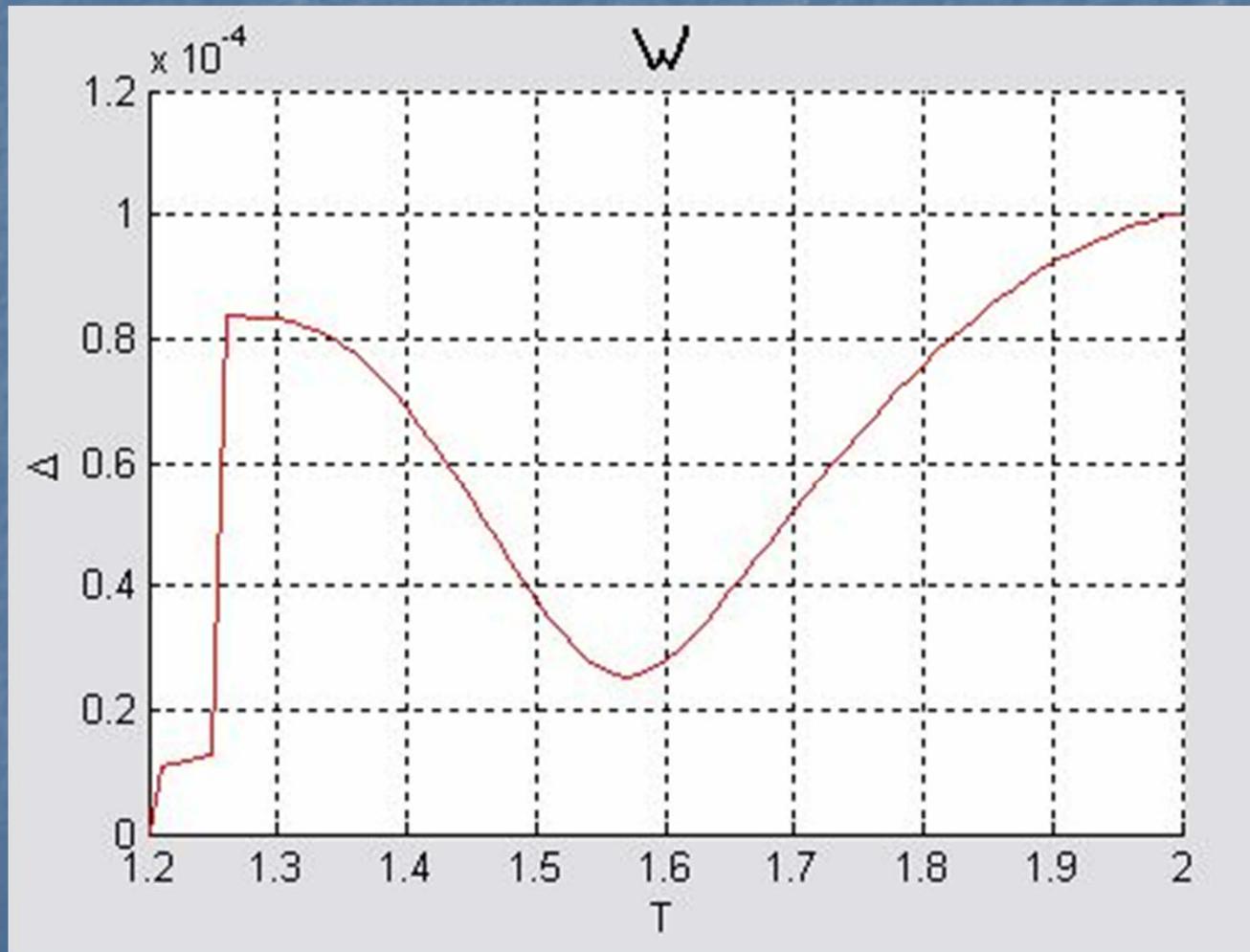
X(t)=

Y(t)=

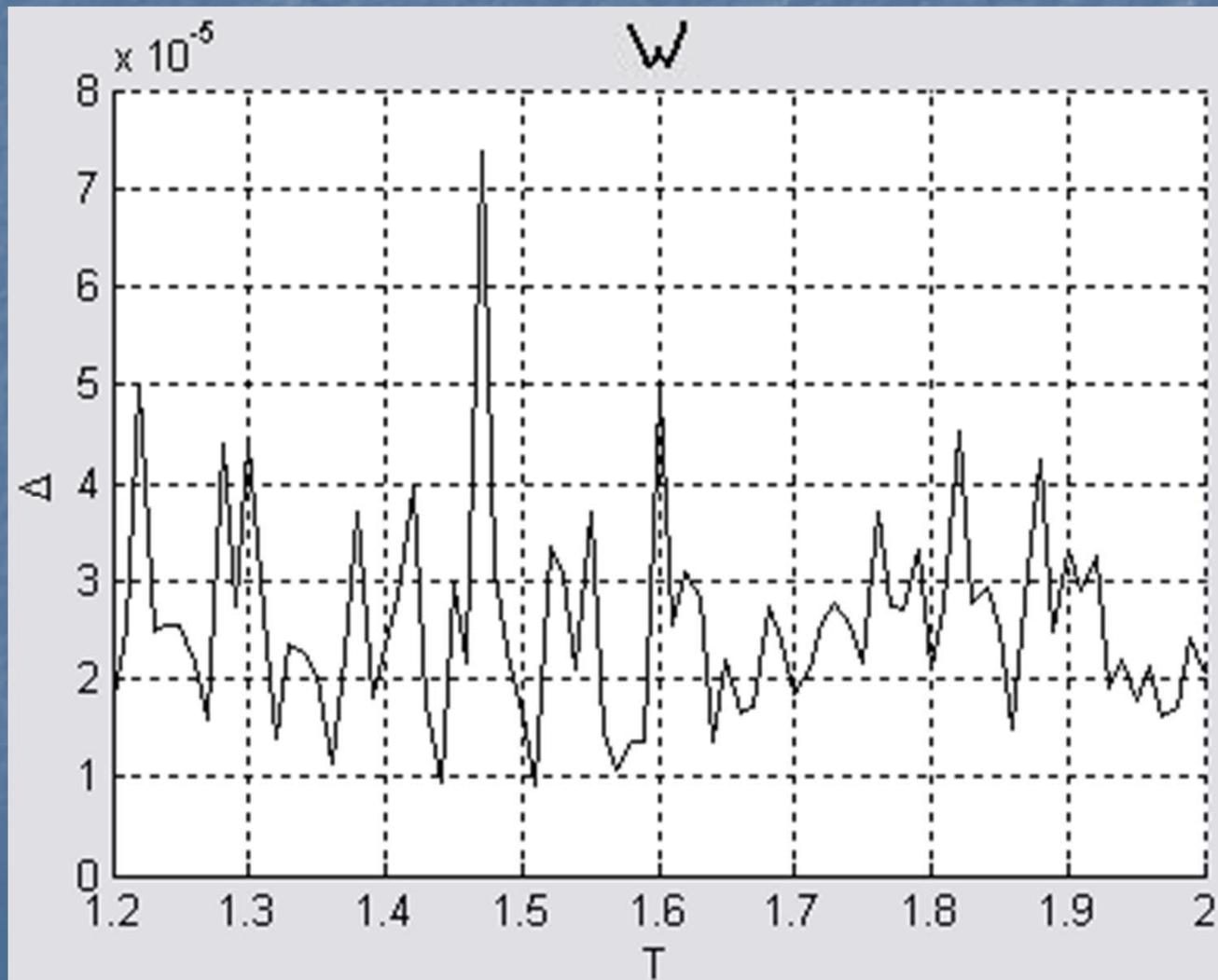
Z(t)=

t:
min step max

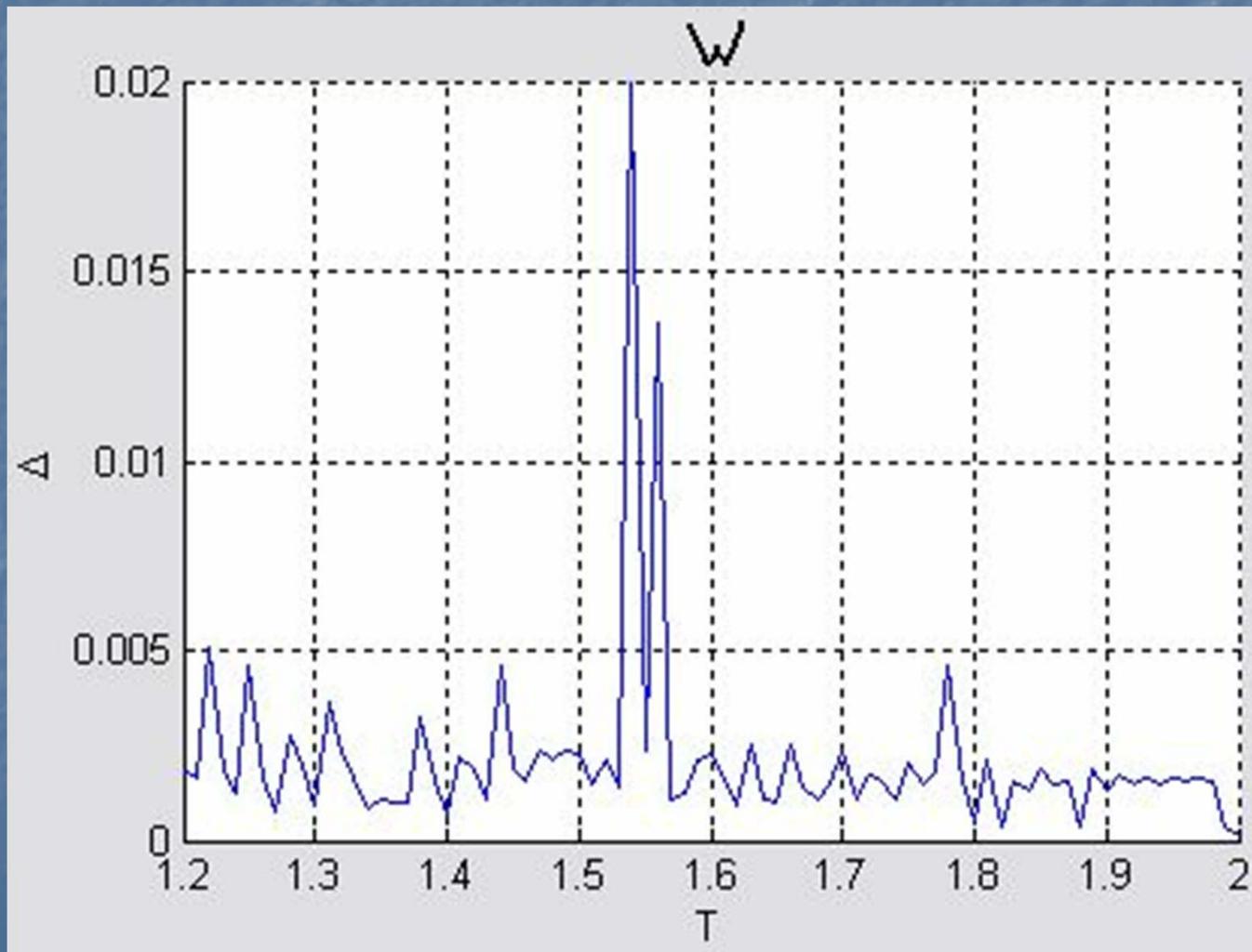
Отклонение схвата манипулятора от заданного
положения (метод SQP).



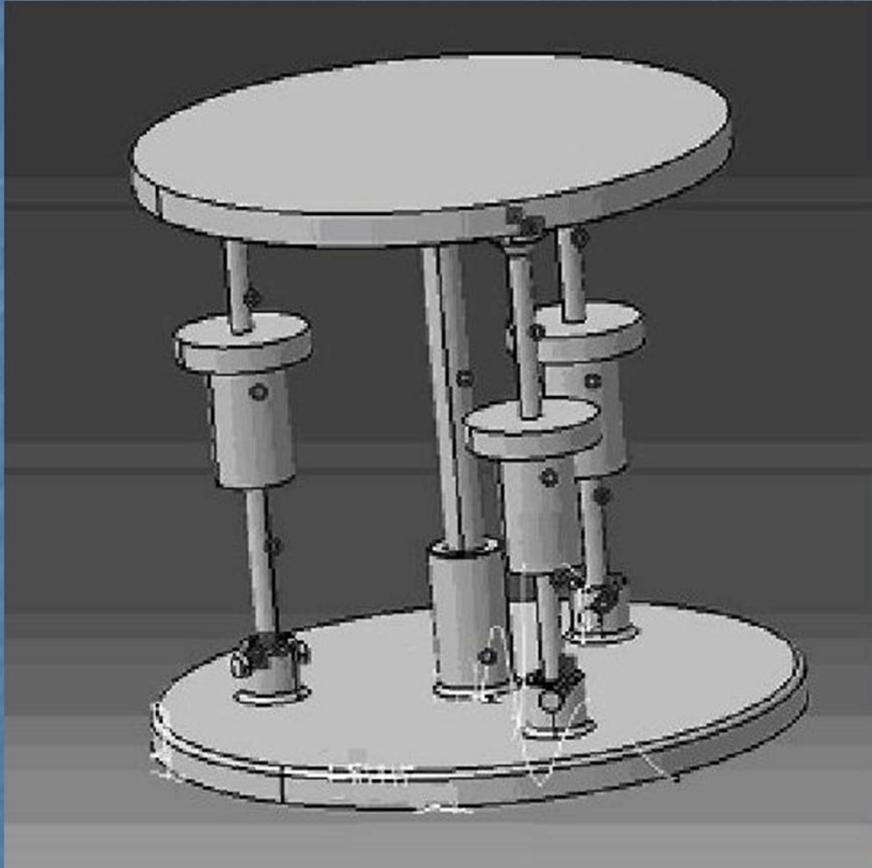
Отклонение схвата манипулятора от заданного
положения (метод BFGS).



Отклонение схвата манипулятора от заданного положения (метод Nelder-Mead).



Заключение



Среди рассмотренных методов наиболее точным можно назвать квазиньютоновский метод BFGS. Практически таким же эффективным оказывается метод SQP, менее точный, но более производительный.

Данный простой пример позволил отработать подход для поиска оптимального управления многосекционным манипулятором типа хобот.