Towards methods for robotic systems capable of human-level dexterity in manipulation and locomotion

Alex Simpkins, Ph.D.
TDLC
University of Washington
Problem

- Little evidence that the brain solves all problems one way
- There are people that have faith in a single approach, but what we are interested in here is pragmatic approaches with potential
- How can we create a model of the brain’s approach which can not only be intuitively and easily abstracted, and fits data, but also can be implemented in robotics along with guarantees
- Importance in manufacturing and society in general
Great but what does this have to do with TDLC?

- Education impact
  - Automated teaching and assessment tools
  - One size fits all vs. student-centered approaches

- These are dynamic problems, involving learning and time – ‘TDLC’

- Parallel problems in education as manufacturing
Solution method

- Create a modular approach which can remain stable for a variety of structures
  - Create self-stable sub-solutions, leading to a BIBO*-type stable system (strictness)

- Real-time capability
  - Hierarchical
  - Changing morphology

*BIBO – ‘Bounded input bounded output’ is stability of linear systems and signals, where inputs that are bounded result in outputs that are bounded. Note BIBO-type here is in reference to the notion of stability for the Class of system used, be it linear or nonlinear (i.e. create solutions with as strong a stability proof as possible)
Background – control and automation

- Automatic control systems and mechatronics
- 1970’s – industrial robotics
- Gluing, welding, cutting, etc
- Pre-programmed, independent axes, stiff systems, well-defined tasks
Modern revolution

- Embedded systems
- Powerful PC’s
- Better sensors and perceptual processing
- Better control theory
A coming shift in the tide

- Rising costs in manufacturing – outsourcing issues
- Aging population
- Medical care
- Robots in society
A theory for theories, a method for methods

Basic tenants for success

- **Stability** - Each component should be stable
- **Simplicity** - Each component should be as simple/fast as possible
- **Independence** - Only information/states are shared
- **Flexibility** - Bandwidth and frequencies can be variable
A theory, a method

- Break up single complex problem into multiple more tractable problems
- Hierarchical – allows different strategies at each stage, more freedom
- Behaviors can be encoded at various levels
Manipulation
High level dynamics

\[ \sum \begin{bmatrix} F_{x_0} \\ F_{y_0} \\ M_0 \end{bmatrix} = \begin{bmatrix} W & 0 & 0 \\ 0 & W & 0 \\ -d_y(\alpha) & d_x(\alpha) & 0 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ 0 \end{bmatrix} - \begin{bmatrix} m_o & 0 & 0 \\ 0 & m_o & 0 \\ 0 & 0 & J \end{bmatrix} \begin{bmatrix} \ddot{x}_o \\ \ddot{y}_o \\ \ddot{\theta}_o \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \nu \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{\theta}_o \end{bmatrix} \]

Solve for acceleration

\[ \begin{bmatrix} \ddot{x}_o \\ \ddot{y}_o \\ \ddot{\theta}_o \end{bmatrix} = \begin{bmatrix} m_o & 0 & 0 \\ 0 & m_o & 0 \\ 0 & 0 & J \end{bmatrix}^{-1} \begin{bmatrix} F_{x_0} \\ F_{y_0} \\ M_0 \end{bmatrix} - \nu \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{\theta}_o \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \]
Collect the states into one state vector $q$

$$q = [x \ y \ \theta \ \dot{x} \ \dot{y} \ \dot{\theta}]^T$$

Rearrange the equations and define matrices, as below, resulting in the standard state space form, where $u$ is the input force from the manipulators

$$A = \begin{bmatrix}
0_{3x3} & I_{3x3} \\
0_{3x3} & 0_{3x3}
\end{bmatrix}, \quad M = \begin{bmatrix}
m & 0 & 0 \\
0 & m & 0 \\
0 & 0 & J
\end{bmatrix}, \quad B = \begin{bmatrix}
0 & 0 & 0 \\
0 & M^{-1} & 0 \\
0 & F_x & F_y & F_o
\end{bmatrix}$$

$$\dot{q} = Aq + Bu$$

We can directly measure the state here, so $C$ is the identity matrix

$$p = Cq$$

*Note: superscript $T$ represents matrix transpose here*
Now we can solve this in any desired fashion

- Simple, linear dynamics (can be replaced)
- Control can be simple or complex
  - LQR – given linear system with quadratic costs, we can exactly solve for a controller which is the optimal choice to make at each state given the problem description
    
    $$J = \int_{0}^{\infty} \left(x^T Q x + u^T R u \right) dt ,$$

    - By minimizing the above equation we get a set of gains $K$ which map state into actions
      
      $$u = -K x.$$
Point mass dynamics

- Balance of forces on the point mass
  - \( b \) represents force we apply to manipulator by motors as mapped through the structure (kinematic map from torque to Cartesian space at endpoint)

- Rearrange to get acceleration

- Add constraint of no slip when in contact, then acceleration of object is related to that of the point mass (will help us later)

\[
\sum F = b_i - m_i g - f_i = m_i \ddot{q}_i, \\
\ddot{q}_i = \frac{1}{m_i} (b_i - f_i) - g.
\]
• First substitute the acceleration relation into the manipulator dynamics to give us acceleration:

\[ f_i + m_i \ddot{a}_o + m_i \dddot{\theta}_o \dot{d}(\alpha_i) = b_i + m_i g \]

• Augment object dynamics with manipulator dynamics, and collect terms to get a nice classic matrix problem which can be solved quickly for \( f \):

\[
N = \begin{bmatrix}
1 & 0 & -m_o & 0 & 0 \\
0 & 1 & 0 & -m_o & 0 \\
0 & 0 & 0 & 0 & J_o \\
I_{m \times m} & 0 & m_i & 0 & m_i d_x(\alpha_i) \\
0 & I_{m \times m} & 0 & m_i & m_i d_y(\alpha_i)
\end{bmatrix},
\]

\[
f = [f_{x_i}, f_{y_i}, a_{x_o}, a_{y_o}, \ddot{\theta}_o]^T,
\]

\[
b = [0, g, 0, b_{x_i} + m_i g]^T.
\]

• This facilitates real-time execution.

\[ Nf = b \]
Policy space force field function

- Drawing fingers toward an object to grasp it at an appropriate location
- Breaking contact with the object when necessary (near workspace boundary)
- Ensuring stability of object grasp during contact breaking
Virtual force field

Toward object center

\[ F_e = K_e \frac{(x - x_e)\beta}{1 + \|x - x_e\|^2}, \quad \beta = \min\left\{ e^{\frac{1}{\beta(\|x_e - x_e\|)}} \right\}, \]  

Toward workspace center

\[ F_s = K_s n\|x - x_w\|^2 e^{-\sigma(x - x_c)^2} \]

Velocity damping

\[ F_d = -K_d \left( \frac{\dot{x} \cdot (x - x_w) \cdot (x - x_w)}{(x - x_w) \cdot (x - x_w)} \right)(x - x_w). \]

Other shapes than circles

\[ F_e = K_e \frac{\phi(x, x_e)\beta}{1 + \psi(x, x_w)}, \quad \beta = \min\left\{ e^{\frac{1}{\beta(\|x_e - x_w\|)}} \right\}, \]
Contact forces - Convex optimization

Minimize sum of forces and error between ideal force and total applied forces

\[ \zeta \| \sum F \|^2, \quad \beta \| \sum F - F_a \|^2 \]

Subject to the constraints that forces only are directed inward toward the object if in contact and that forces are less than a maximum

\[ \min \quad \varepsilon = F^T G F \quad F \cdot n < 0 \quad F_i < F_{\max}, \quad 0 < i \leq m \]

\[ G = \begin{bmatrix} \zeta I & 0 & 0 & 0 & 0 \\ 0 & \zeta I & 0 & 0 & 0 \\ \beta W & 0 & -\beta & 0 & 0 \\ 0 & \beta W & 0 & -\beta & 0 \\ -\beta dy & \beta dx & 0 & 0 & -\beta \end{bmatrix} \]

W – vector of ones
I – identity matrix
\( \zeta \) and \( \beta \) scaling parameters
F – force of the manipulators
\( F_a \) – goal force
Experiments and results

- Experiment 1: grasping from various initializations
- Experiment 2: tracking known shape, various trajectories
- Experiment 3: Stability and performance
- Experiment 4: cooperation and flexibility
Random initializations, range of motion, 1msec, symplectic euler, real-time solutions

Typical damped 2\textsuperscript{nd} order sys-type behavior, balance human-like, 5-600msec settle
Random accelerations

\[ x_r(k) = \sin[\gamma], \quad \gamma = N_1(0, \Delta_r) \]
\[ y_r(k) = \sin[\varphi], \quad \varphi = N_2(0, \Delta_r) \]
\[ \theta_r(k) = \sin[\phi], \quad \phi = N_3(0, \Delta_r) \]
Tracking

http://casimpkinsjr.radiantdolphinpress.com/files/manipulation2d/2dmanipulation.mp4
Tracking – \( \sin(\text{randn}) \)
Tracking – *steps, ramps, sinusoids*
Eigenvalues in left half of real-imaginary plane are stable
*so eigenvalues provide a means to analyze how the properties of the system’s dynamics are changed when we add a control system, in the linear case

Uncontrolled system

\[ \lambda_o = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T \]

Controlled system

\[ \lambda_c = [-10.5737 \ +10.5737i \ -10.5737 \ +10.5737i \ -10.5737 \ +10.5737i \ -10.5737 \ -10.5737i \ -10.5737 \ +10.5737i \ -10.5737 \ -10.5737i ] \]

Our particular LQR parameters for these control experiments

\[ Q = \begin{bmatrix}
500 & 500 & 0 \\
500 & 0 & 0 \\
0 & 0 & 0 \\
\end{bmatrix}, \quad R = \begin{bmatrix}
0.01 & 0.01 & 0 \\
0.01 & 0.01 & 0 \\
0 & 0 & 0.01 \\
\end{bmatrix} \]
Stability – impulse response

Uncontrolled system

Controlled system

(a) Impulse Response

(b) Impulse Response
Stability – step response

*In both this slide and previous, note the uncontrolled system grows in an unbounded way, while the controlled system responds to the perturbation in a controlled way*
• If we allow ‘telekinesis’ does the system still work just with the high level?

• Yes it does indeed

• How does the high level only compare with the overall manipulator-controlled system?

• Small error, some difference due to dynamics of manipulators, but this is to be expected
Different numbers of the manipulators can solve the problem with no change to the structure of the controller, can be changed in real-time.

Overall force goes down per manipulator with more manipulators.

There is an optimal number for a task – sometimes too much or not enough redundancy affects performance.
A change in hardware methodology

- Modbots and the future
  - Compliant
  - Fast
  - Sensitive
  - Force and position feedback
  - Low level processor
Summary

- Multi-theory approach
- Experimental demonstrations
  - More manipulators not necessarily better
  - Redundancy is important
- Pragmatism
- Design for/with implementation in mind
Thank you!  Questions?

casimpkins@gmail.com

http://casimpkinsjr.radiantdolphinpress.com

References


Simpkins, Kelley, and Todorov (2011), Modular bio-mimetic robots that can interact with the world the way we do. In the Proceedings of the International Conference of Robotics and Automation (ICRA) 2011.

www.ddlc.ucsd.edu


