Getting things done: Intelligent control of action in biological and artificial systems

Emanuel Todorov

Cognitive Science Department
University of California San Diego

Why do we have a brain?

• the purpose of the brain is to produce adaptable and complex movements

• movement is the only way we have
  – of interacting with the world
  – of communicating (speech, gestures, writing)

• sensory, memory and cognitive processes have evolved to guide movement

TREES DO NOT NEED BRAINS, BECAUSE THEY DO NOT HAVE MUSCLES
Levels of control

High-level vs. Low-level

Internal models: ability to (learn to) predict

External stimulus vs. Self-generated stimulus

Normal vs. Perturbed vs. Adapted vs. Aftereffect
The difference between robotic and human movement

Example of robotic manipulation

Example of human manipulation

Variability: a signature of complex computations
Task-relevant control via synergies

General Idea:
High-level controls have predictable effects on task-relevant parameters, despite variability in individual degrees of freedom

Grasping: ~ 4 synergies
Complex manipulation: ~ 12 synergies

Optimal control models

Ingredients of optimal control modeling:
- Model of the sensory-motor periphery
- Definition of the performance criterion
- Method for designing the optimal control law
Optimal feedback controllers in redundant tasks use *task-specific sensory-motor* synergies.

Open-loop control: Locomotion

(Anderson and Pandy, 2001)

**Biomechanical model:**
23 joints, 54 muscles

**Optimization:**
minimize energy under constraints on initial and final pose

10000 hours CPU time!?

**Joint angles:** model vs. data

**Ground reaction forces:** model vs. data
Open-loop control: Eye and arm movements  
(Harris and Wolpert, 1998)

Simplified biomechanical models, that include control-dependent noise

Minimize endpoint variance  
(measured over some interval)

Eye movements: data vs. model

Arm movements: data vs. model

Learning to maximize rewards

Define control system with (many) adjustable parameters

Define a scalar reward function, that objectively measures how well the controller is doing

Numerically tweak parameters so as to increase reward