Summary: main conceptual points

Gregor Schöner, INI, RUB

Autonomous robotics

- industrial robots are programmed in a fixed and detailed way based on information about the world available to the programmer
- *auto-nomos*: giving laws to oneself:
- => autonomous robots generate behavior based on sensory information obtained from their own on-board sensors

Autonomous robotics

- autonomy as an programming *interface*:
- give goals to a robot at a high level, using human language and gesture in a shared environment…
- the autonomous robot then deals with the details of how to achieve those goals…

[Ioannis Iossifidis at the INI]

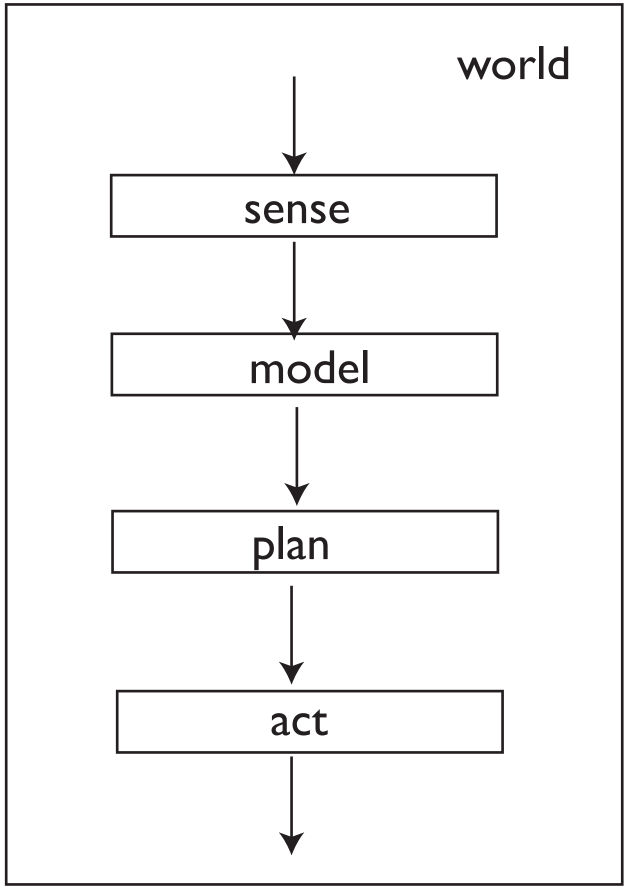
Autonomous robotics as a “playground” of research

- highly interdisciplinary field

- sensing
- perception
- modeling
- AI/planning
- mechanics
- control
- compliance
- embedded computing
- communication / data security
- energetics
- user interfaces
- safety
- ethics

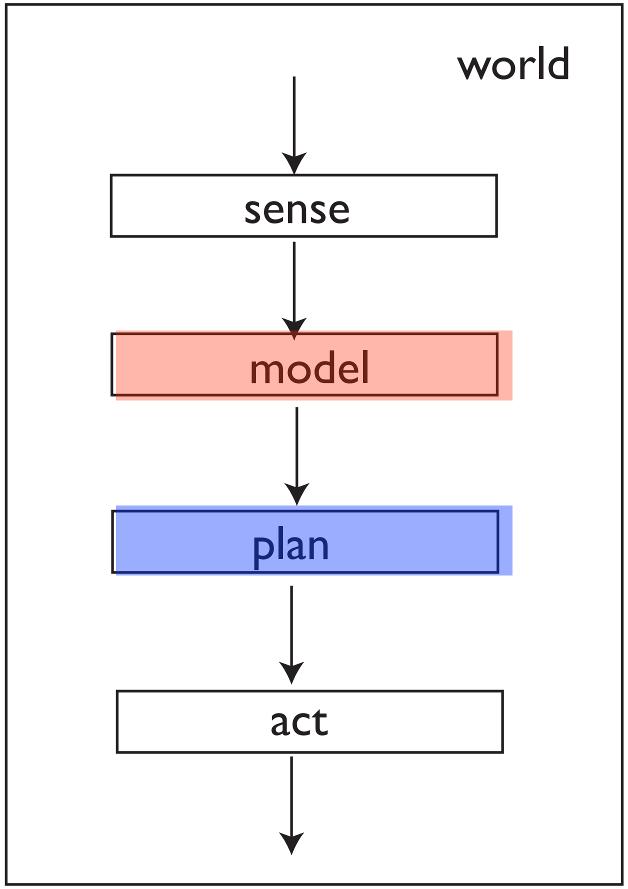
Sense-plan-act

- The classical vision dates back to the 1950's…
- separates the problem into modular functions that follow the sensory stream from sensors to motors



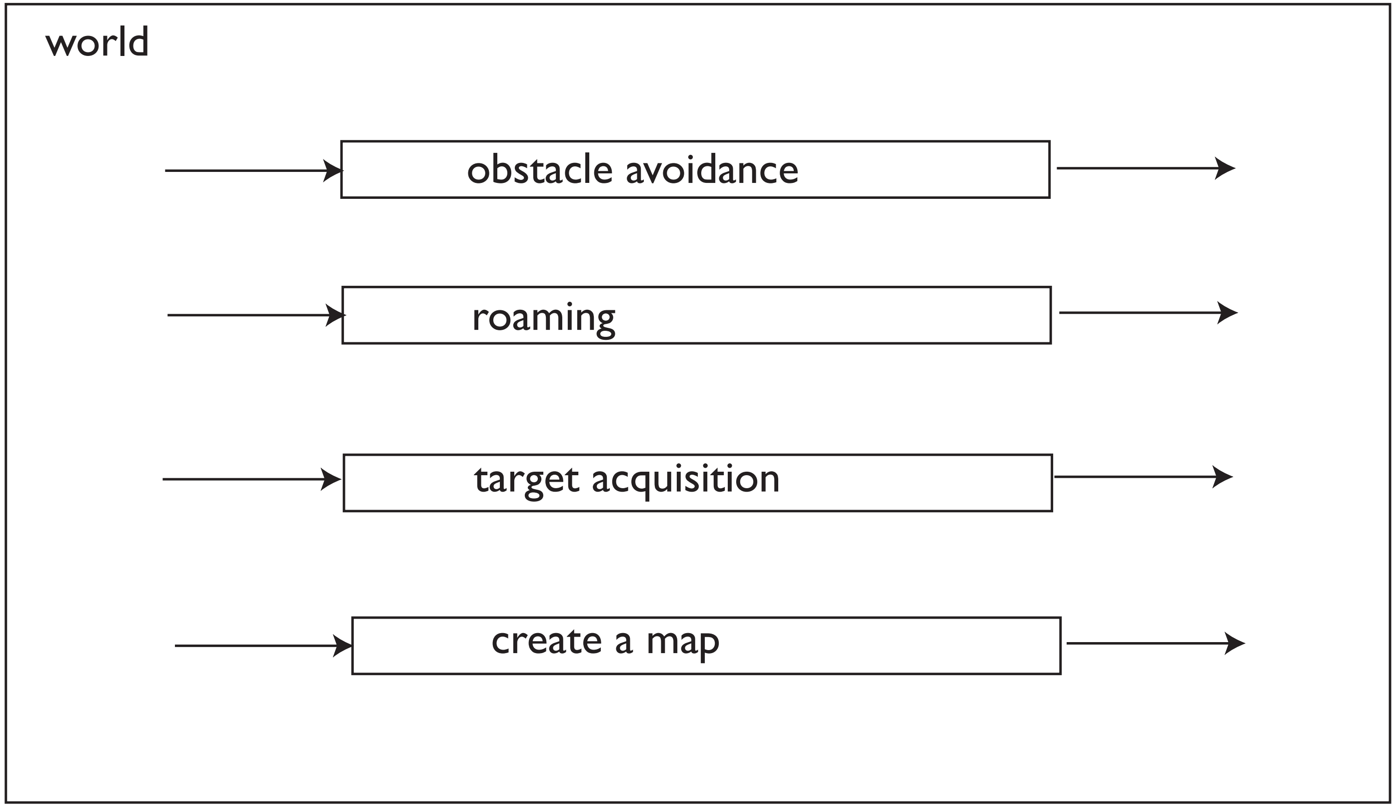
Sense-plan-act

- “intelligence” comes from two core functions
- modeling the world, which entails perception and map building
- planning action, which entails generating sequences of actions that lead to a goal



Behavior-based

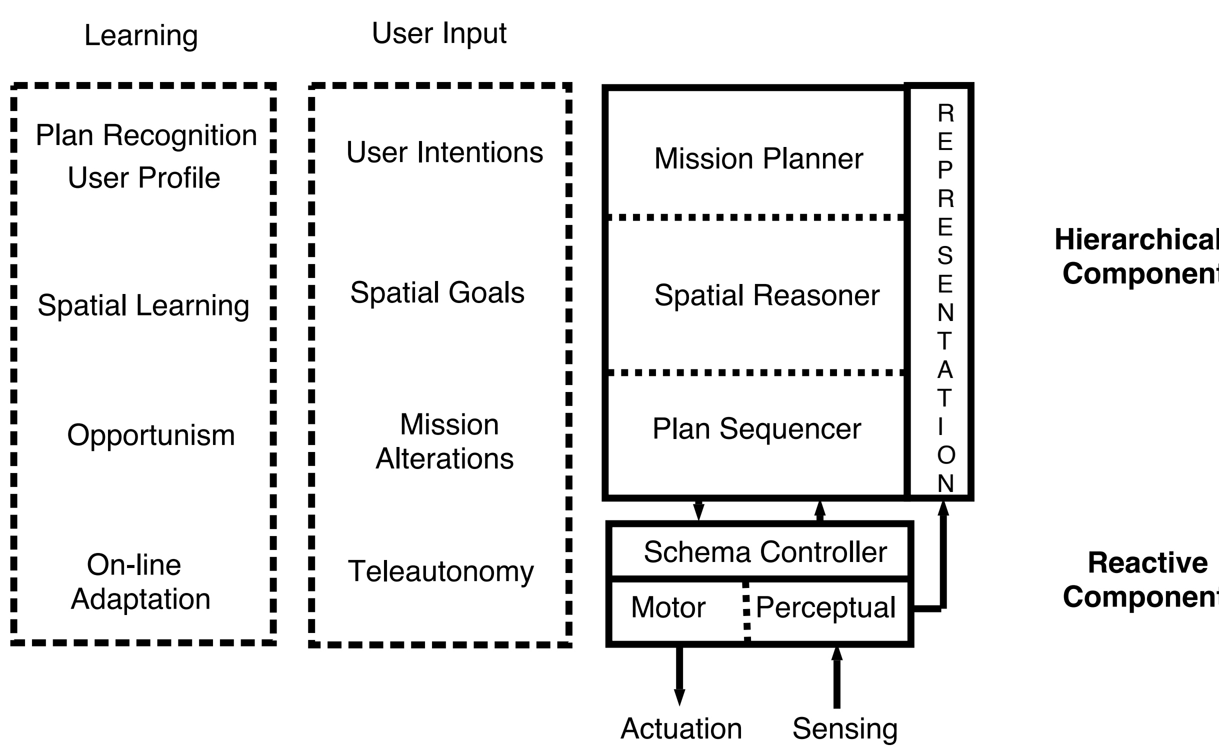
- minimize the difficulty of generating world models by having special purpose perception
- integrate planning and control in individual behaviors



[Brooks and others, 80's]

Hybrid architectures

- use behavior-based ideas as a “reactive layer”
- use sense-plan-act ideas at a ``higher” level of goal-oriented planning



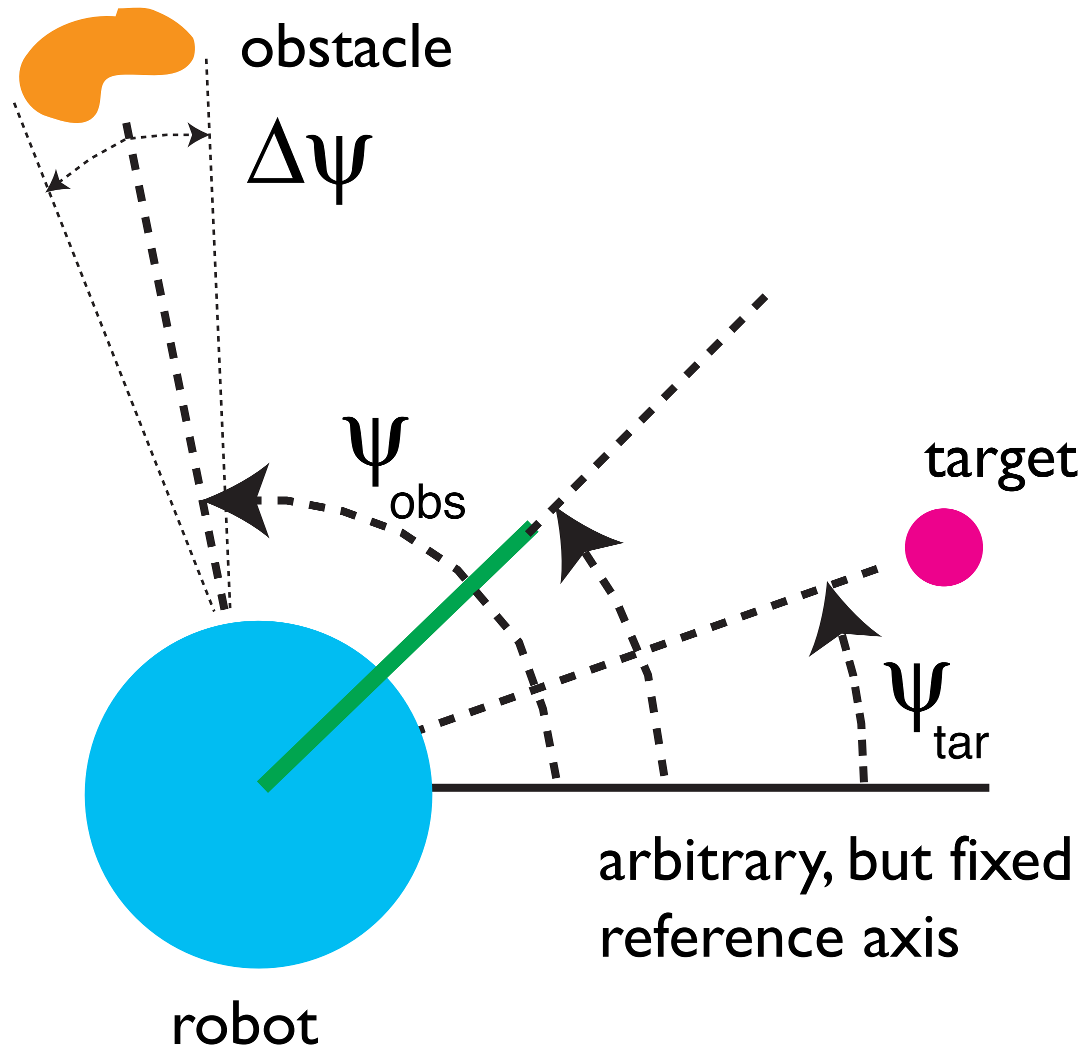
[Arkin, Balch, 1997]

Basic ideas of attractor dynamics approach

- behavioral variables
- time courses from dynamical system: attractors
- tracking attractors
- bifurcations for flexibility

Behavioral variables: example

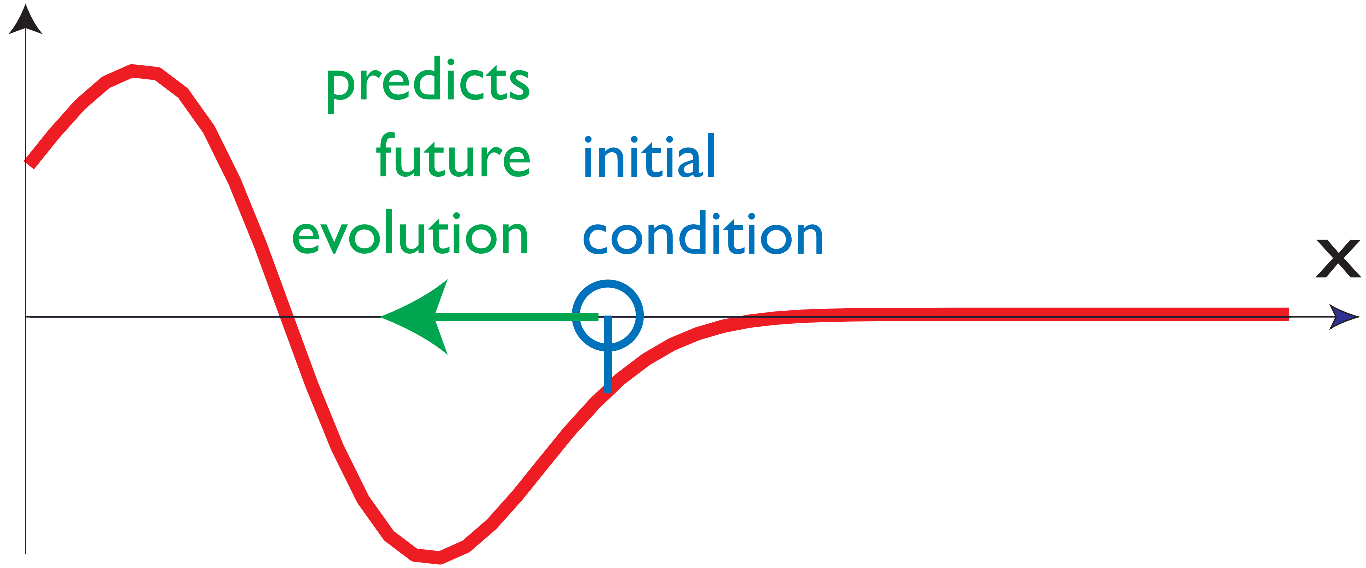
- vehicle moving in 2D: heading direction
- constraints: obstacle avoidance and target acquisition



Dynamical system

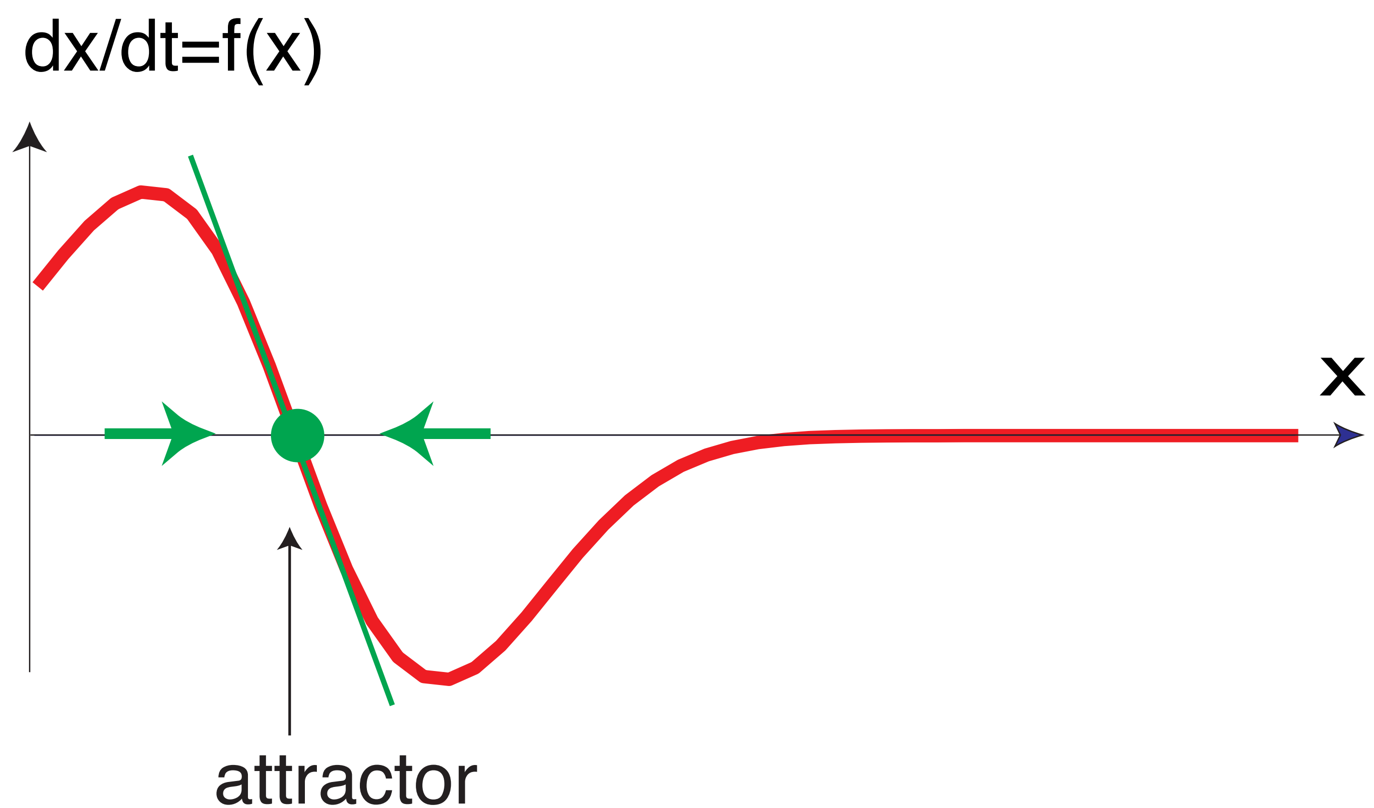
- present determines the future

$dx/dt=f(x)$



Dynamical systems

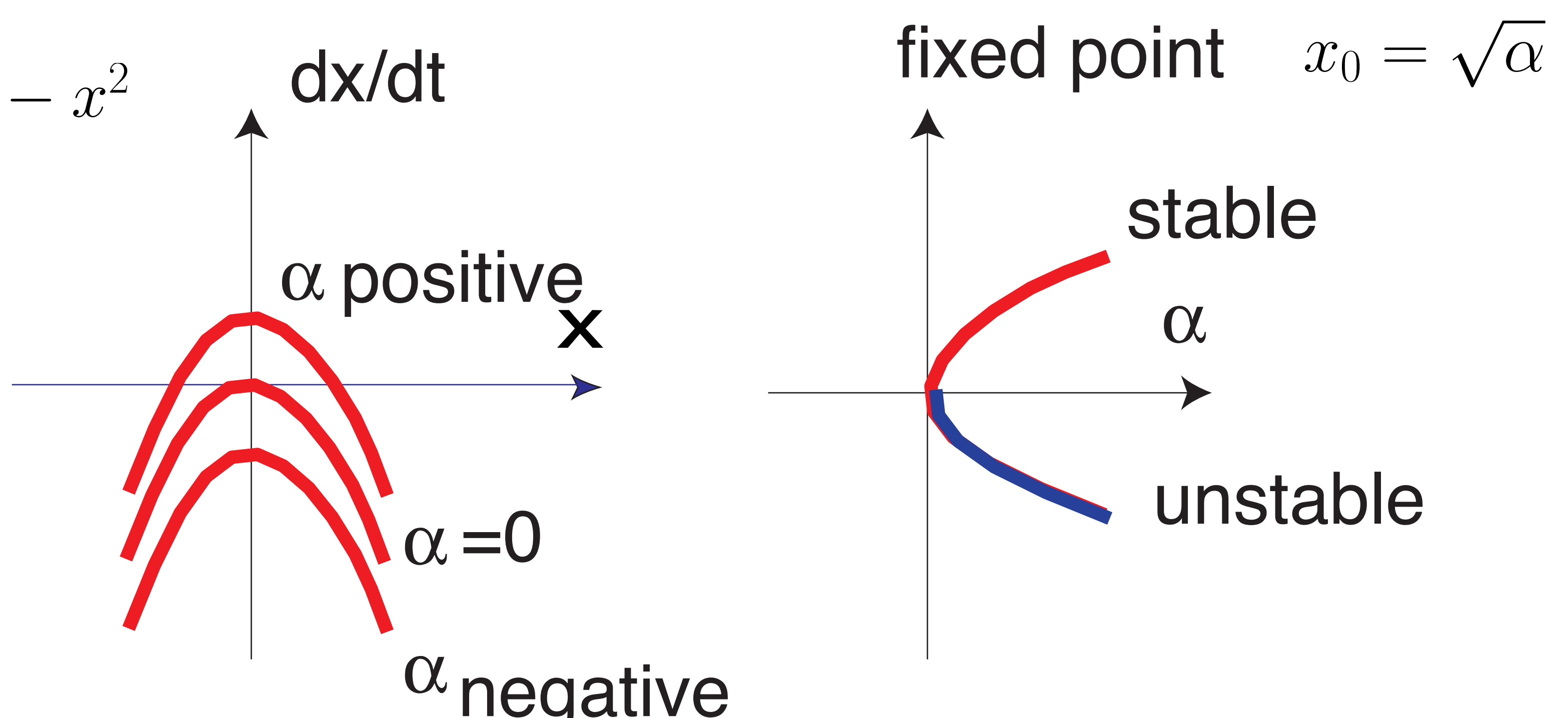
- fixed point = constant solution
- neighboring initial conditions converge = attractor



Bifurcations are instabilities

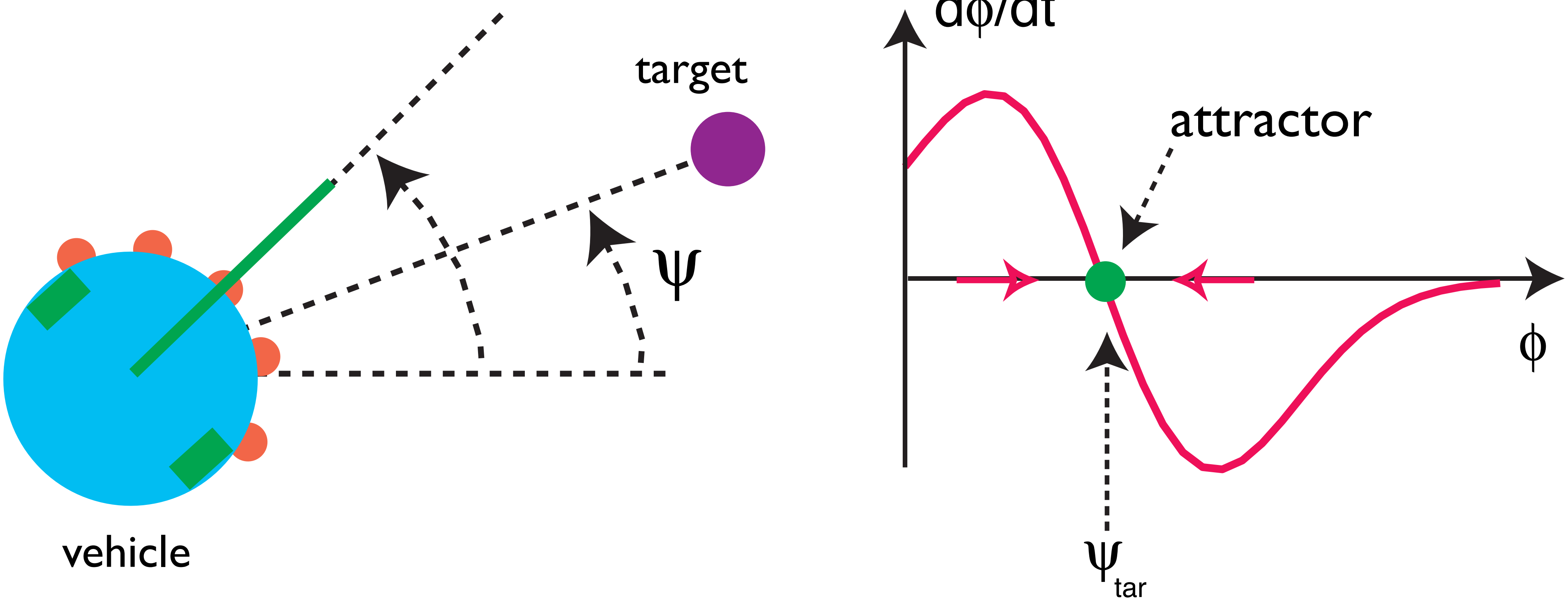
- In families of dynamical systems, which depend (smoothly) on parameters, the solutions change qualitatively at bifurcations
- at which fixed points change stability

$$\dot{x} = \alpha - x^2$$



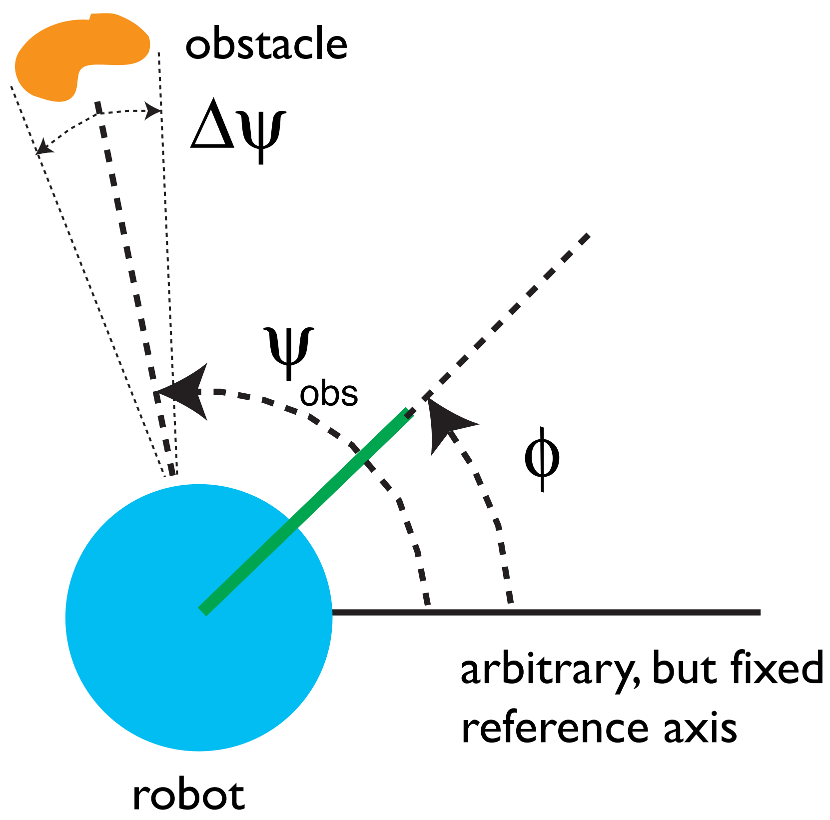
Behavioral dynamics

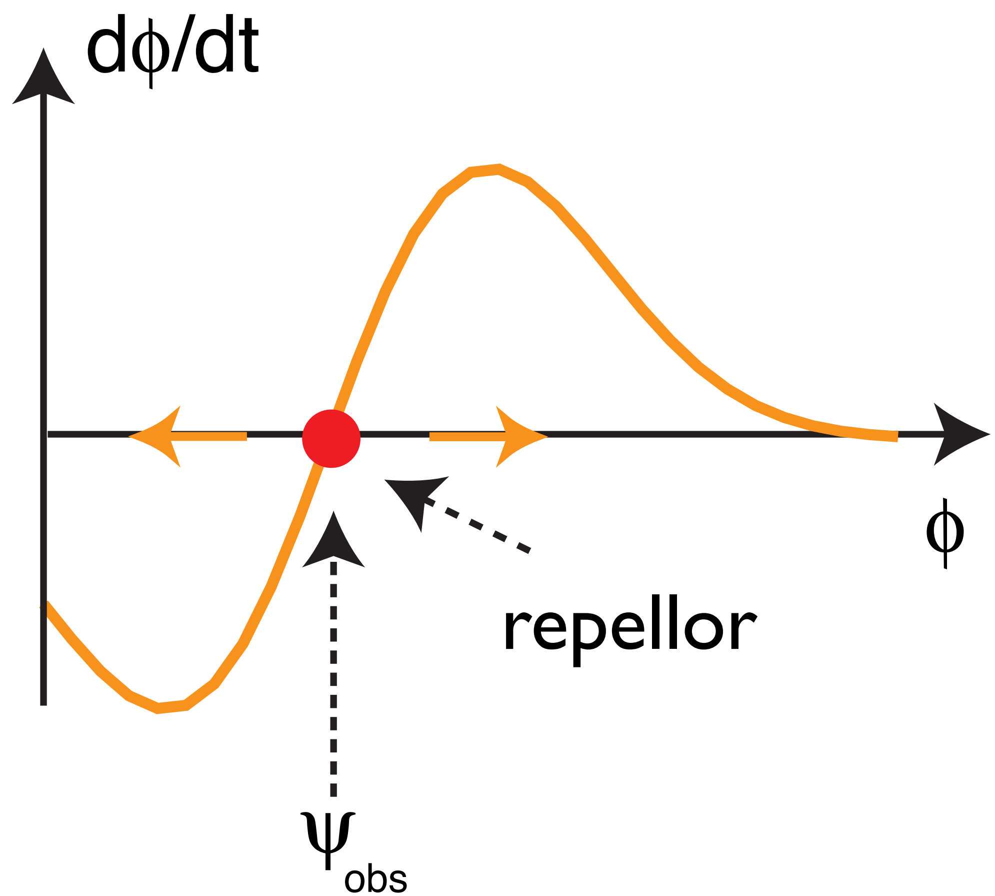
- target acquisition



Behavioral dynamics

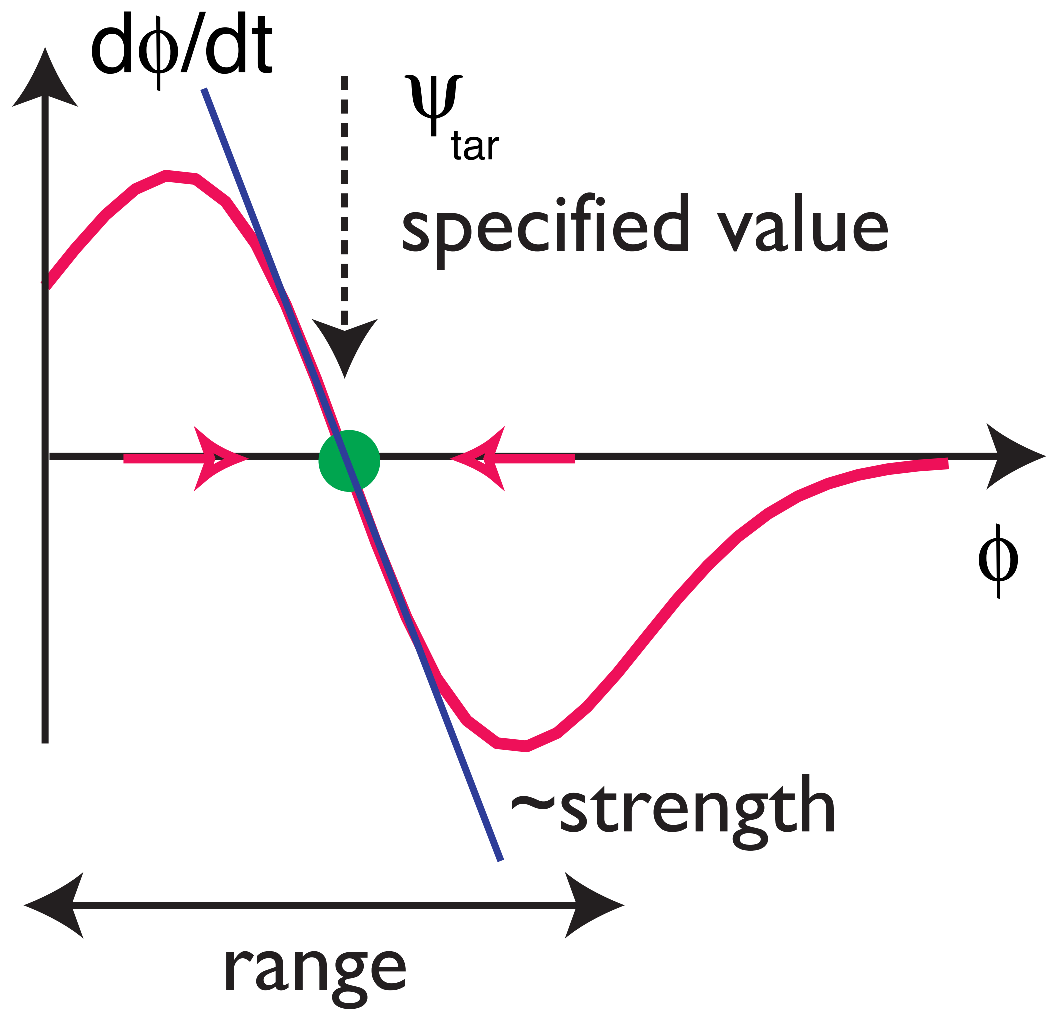
- obstacle avoidance





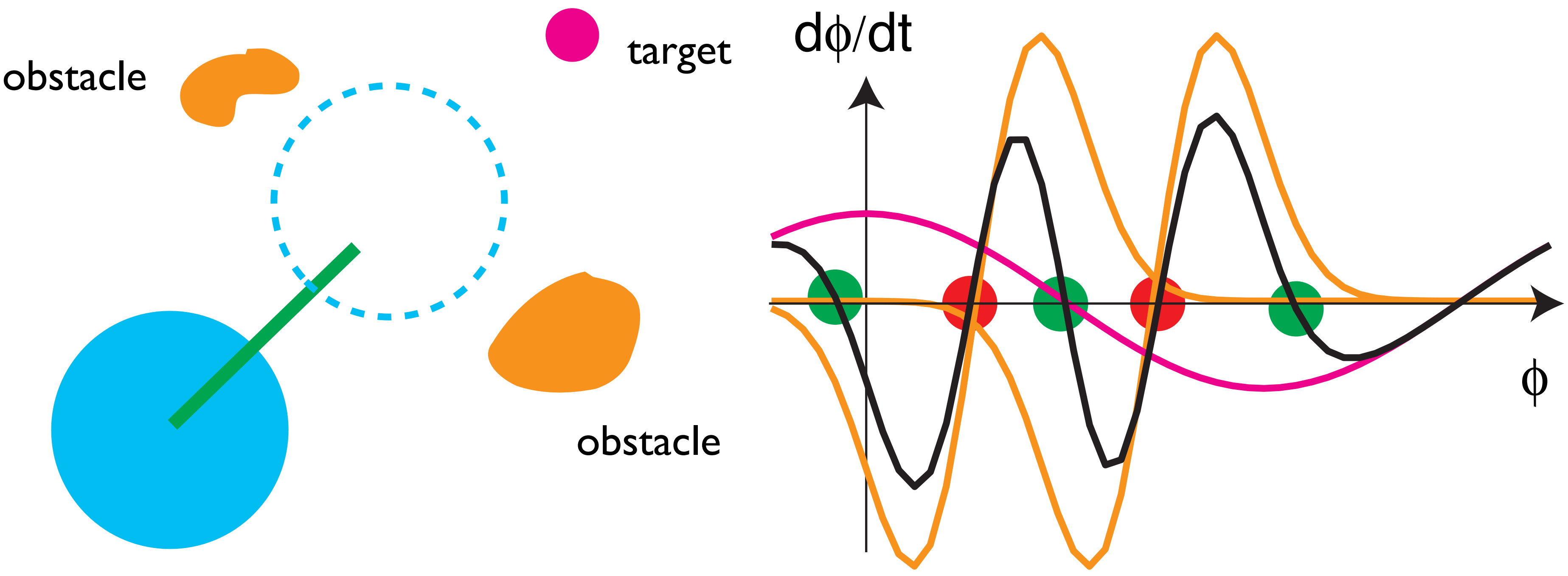
Behavioral dynamics

- each contribution is a “force-let” with
  - specified value
  - strength
  - range



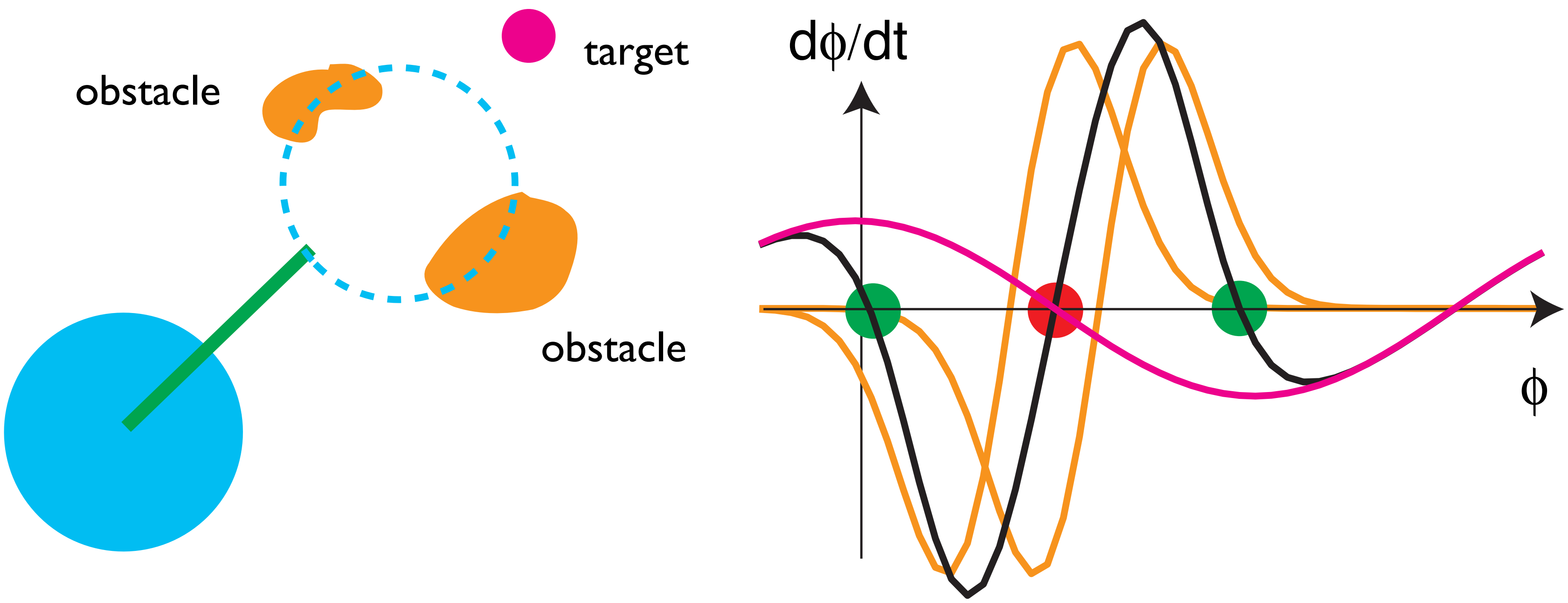
Behavioral dynamics: bifurcations

- constraints not in conflict



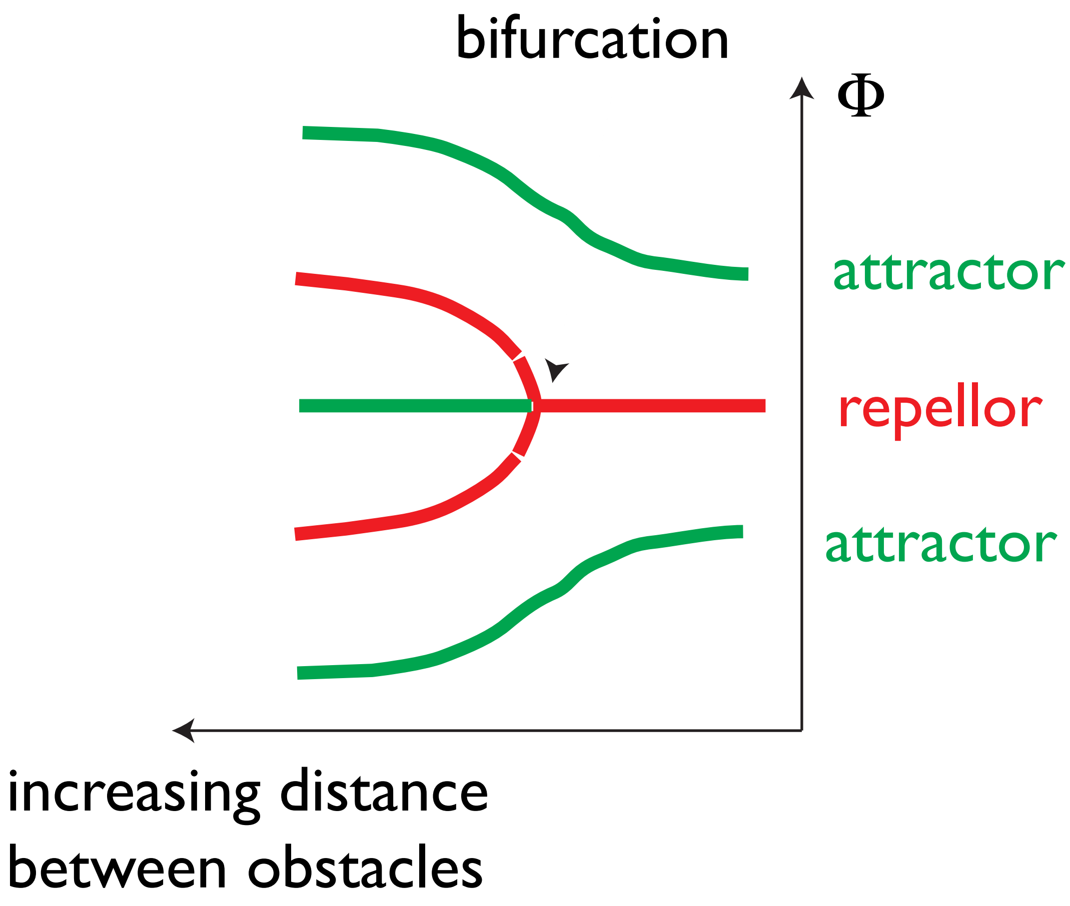
Behavioral dynamics

- constraints in conflict

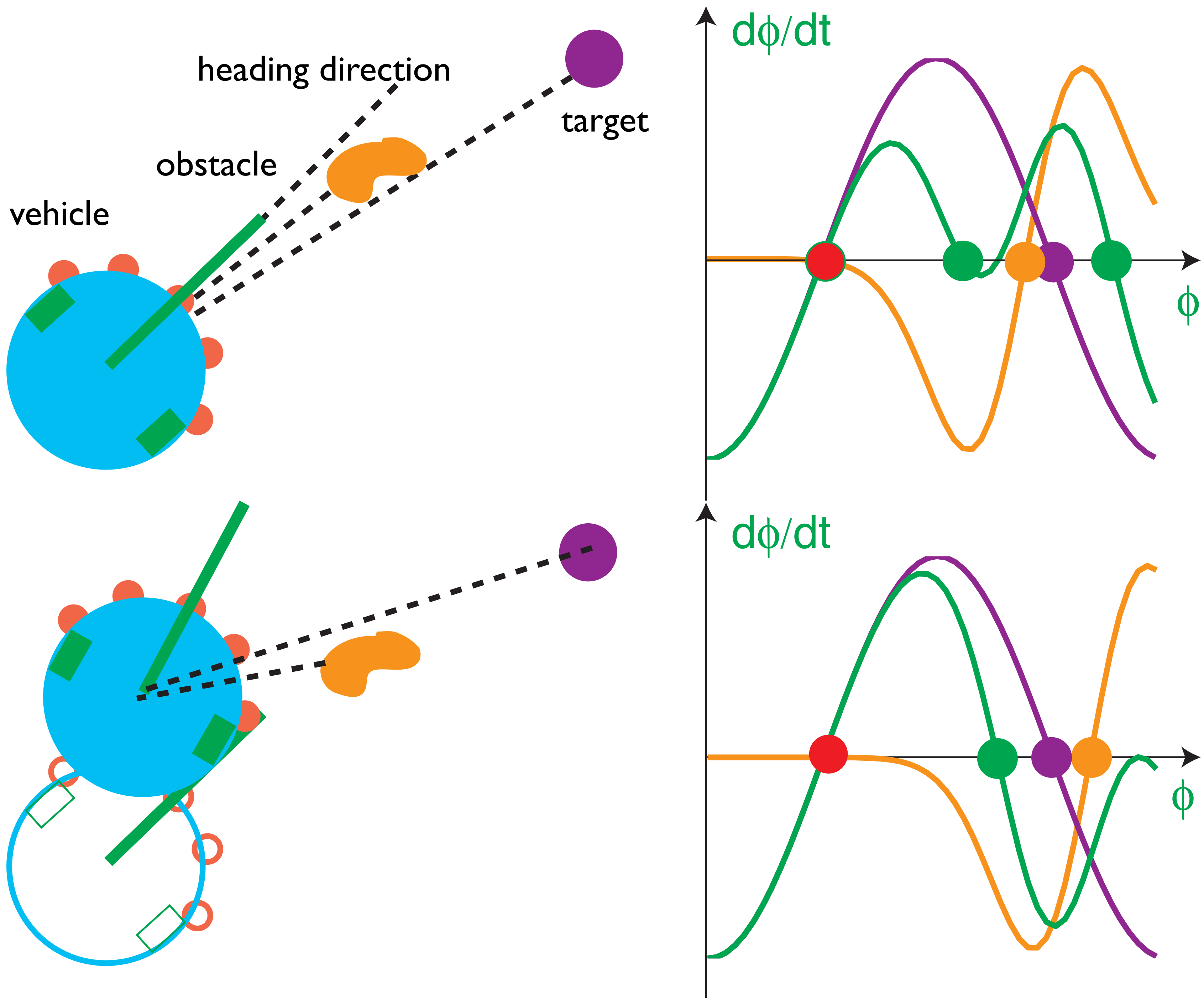


Behavioral dynamics

- transition from "constraints not in conflict" to "constraints in conflict" is a bifurcation

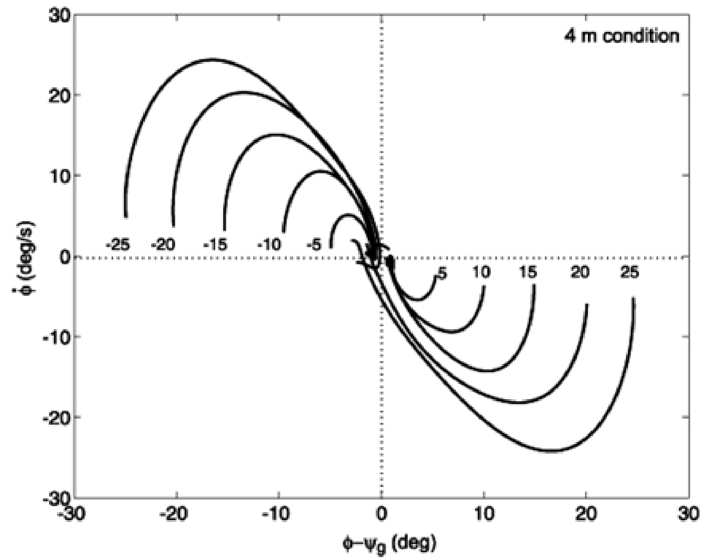


In a stable state at all times



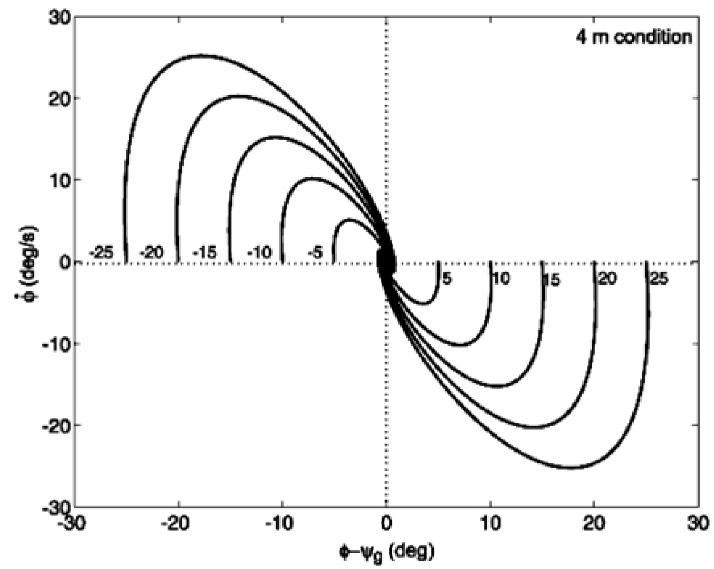
model-experiment match: goal

experiment

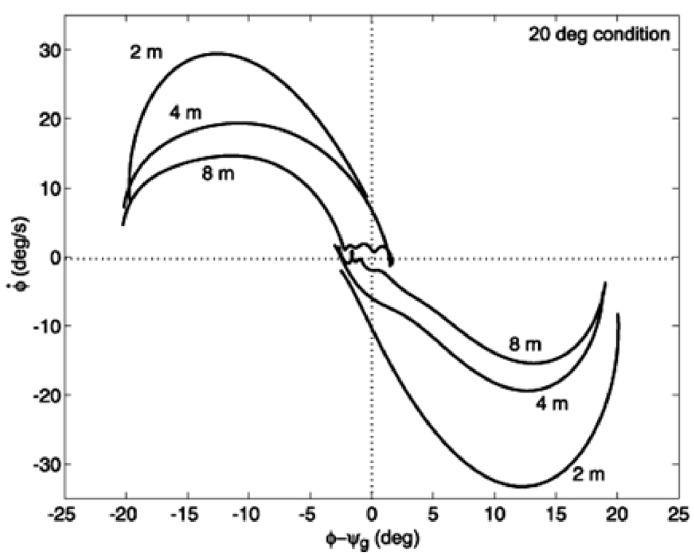


(a)

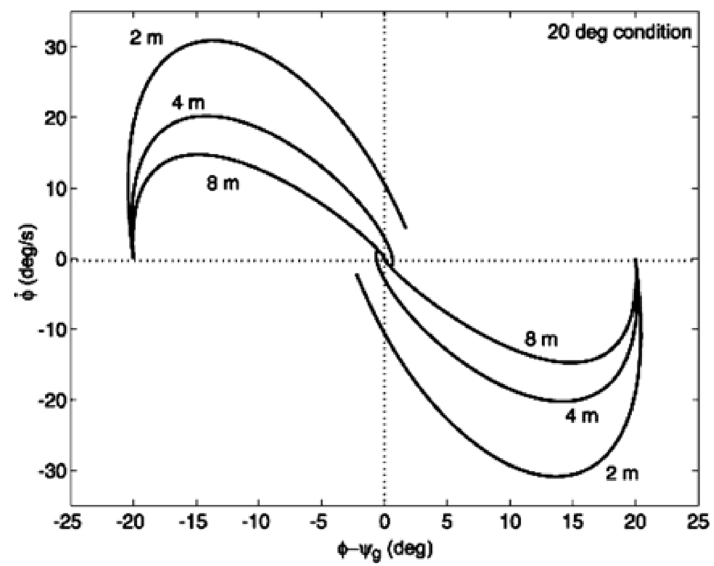
model



(a)



(b)

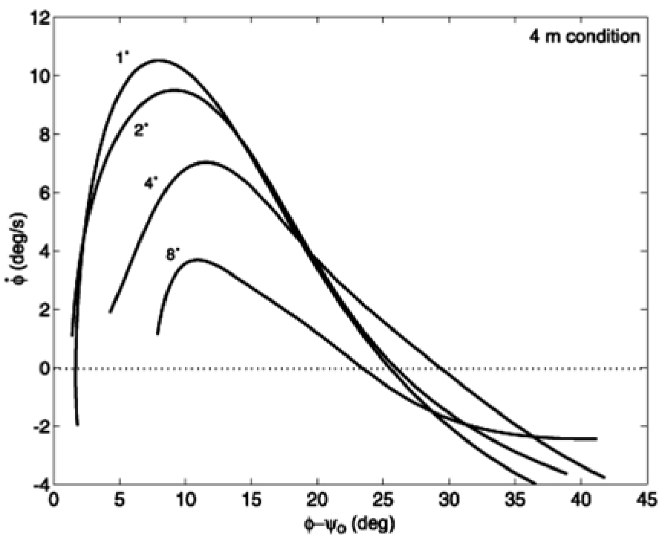


(b)

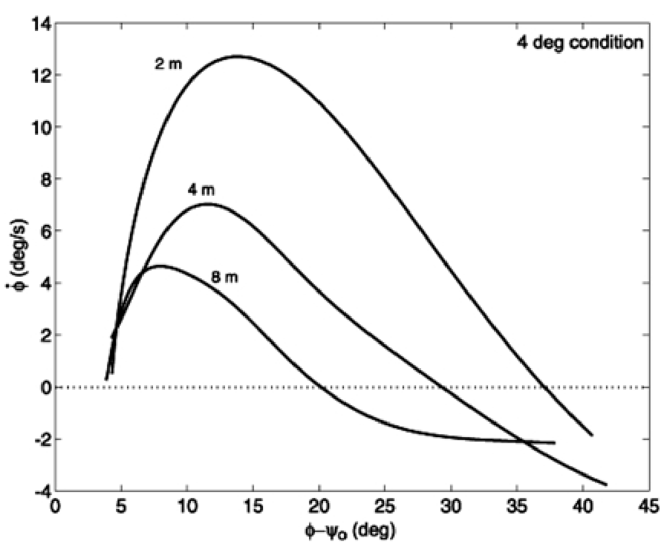
[Fajen et al. 2003]

model-experiment match: obstacle

experiment

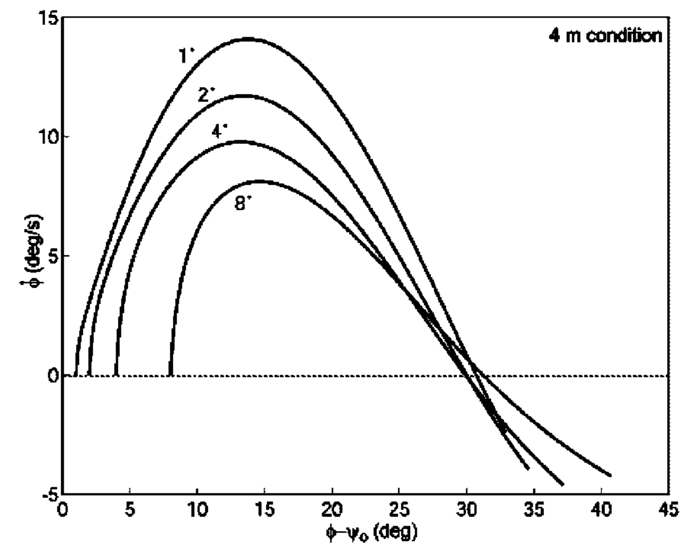


(a)

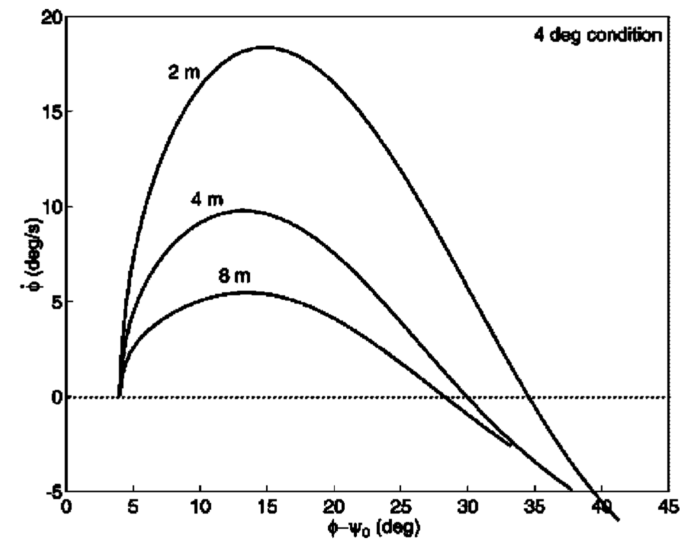


(b)

model



(a)



(b)

[Fajen et al. 2003]

2nd order attractor dynamics to explain human navigation

2

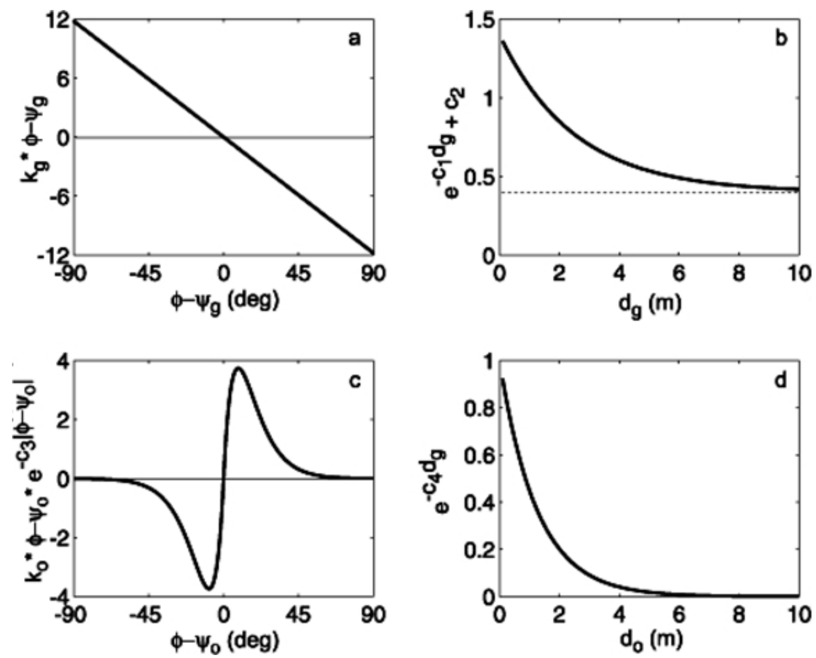
inertial term

damping term

attractor goal heading

$$\ddot{\phi} = -b\dot{\phi} - k_g(\phi - \psi_g)(e^{-c_1 d_g} + c_2) + k_o(\phi - \psi_o)\left(e^{-c_3|\phi - \psi_o|}\right)(e^{-c_4 d_o})$$

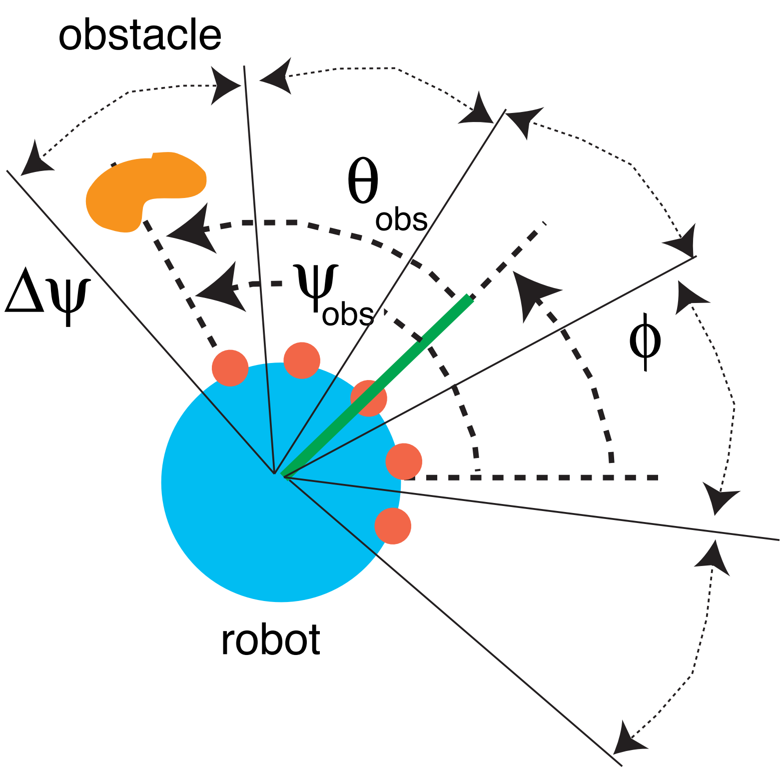
repellor obstacle heading

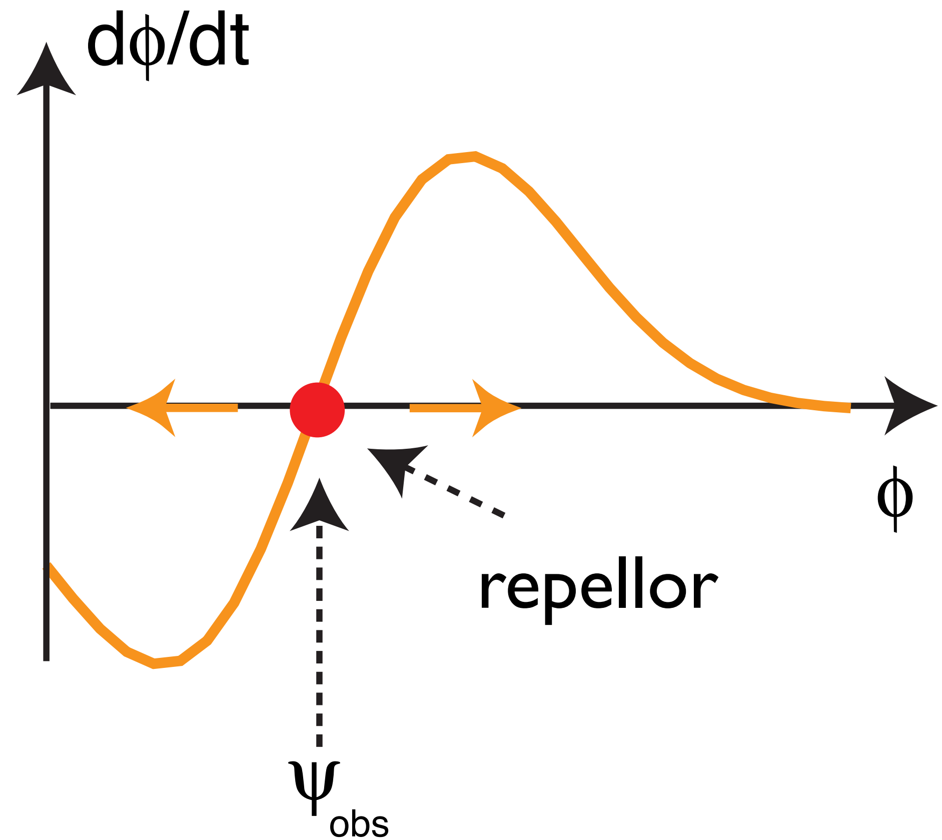


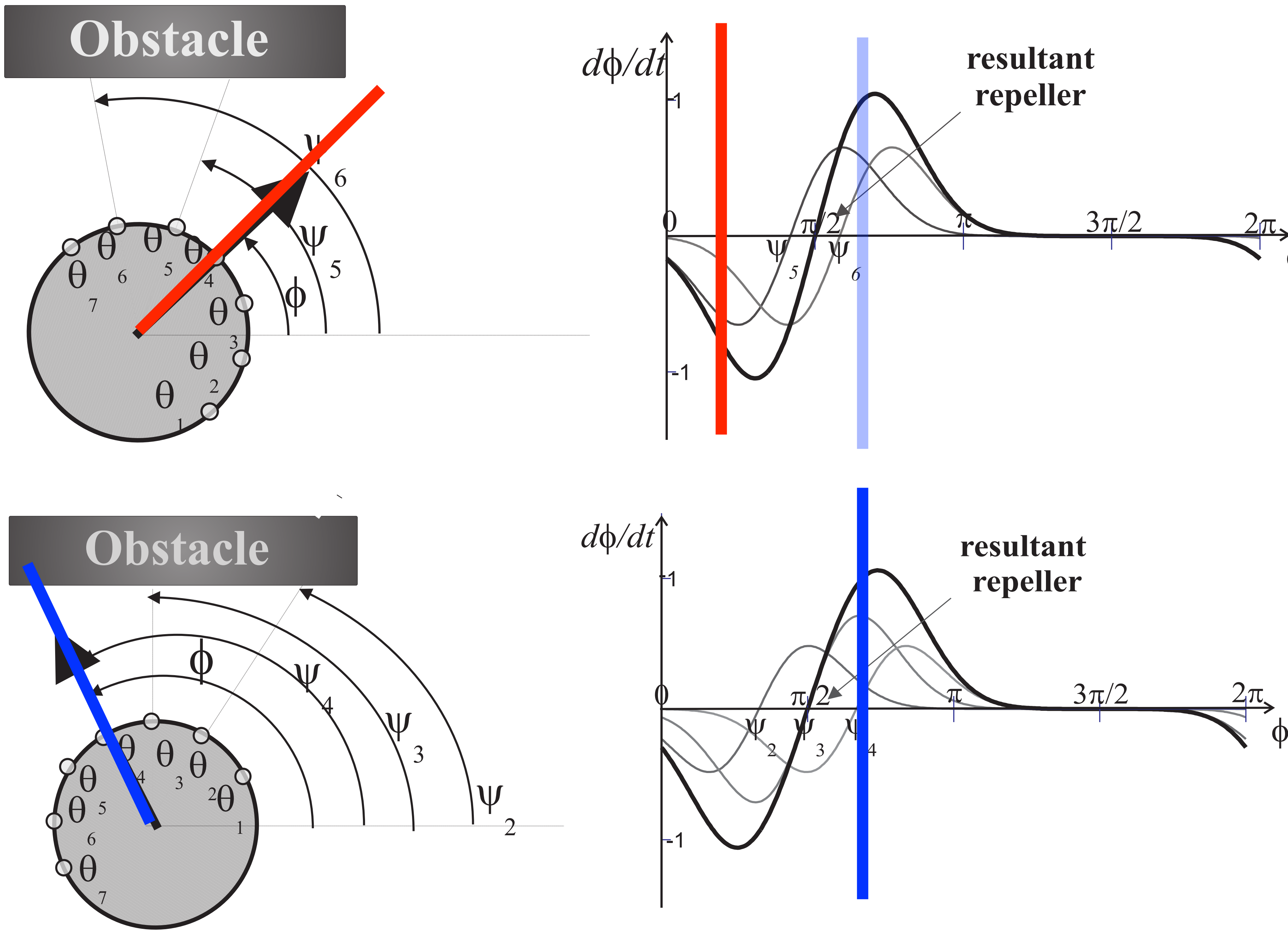
[Fajen et al. 2003]

Obstacle avoidance: sub-symbolic

- obstacles need not be segmented
- do not care if obstacles are one or multiple: avoid them anyway…





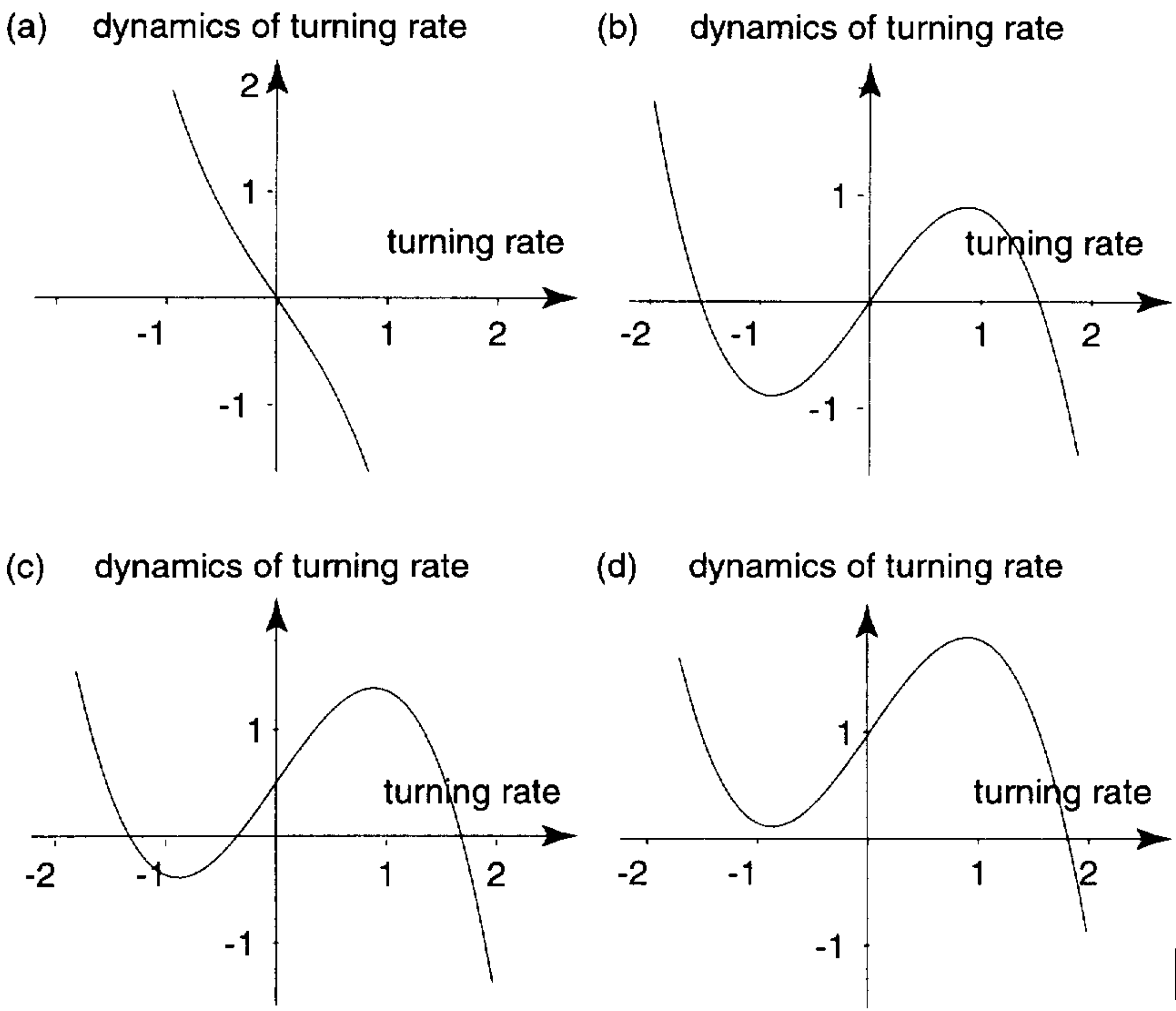


=> dynamics invariant!

[from: Bicho, Jokeit, Schöner]

Alternative 2nd oder approach

$$\dot{\omega} = (\alpha + \frac{1}{2}\pi)c_{\text{obs}}F_{\text{obs}} + \alpha\omega - \gamma\omega^3$$



[Bicho, Schöner, 97]

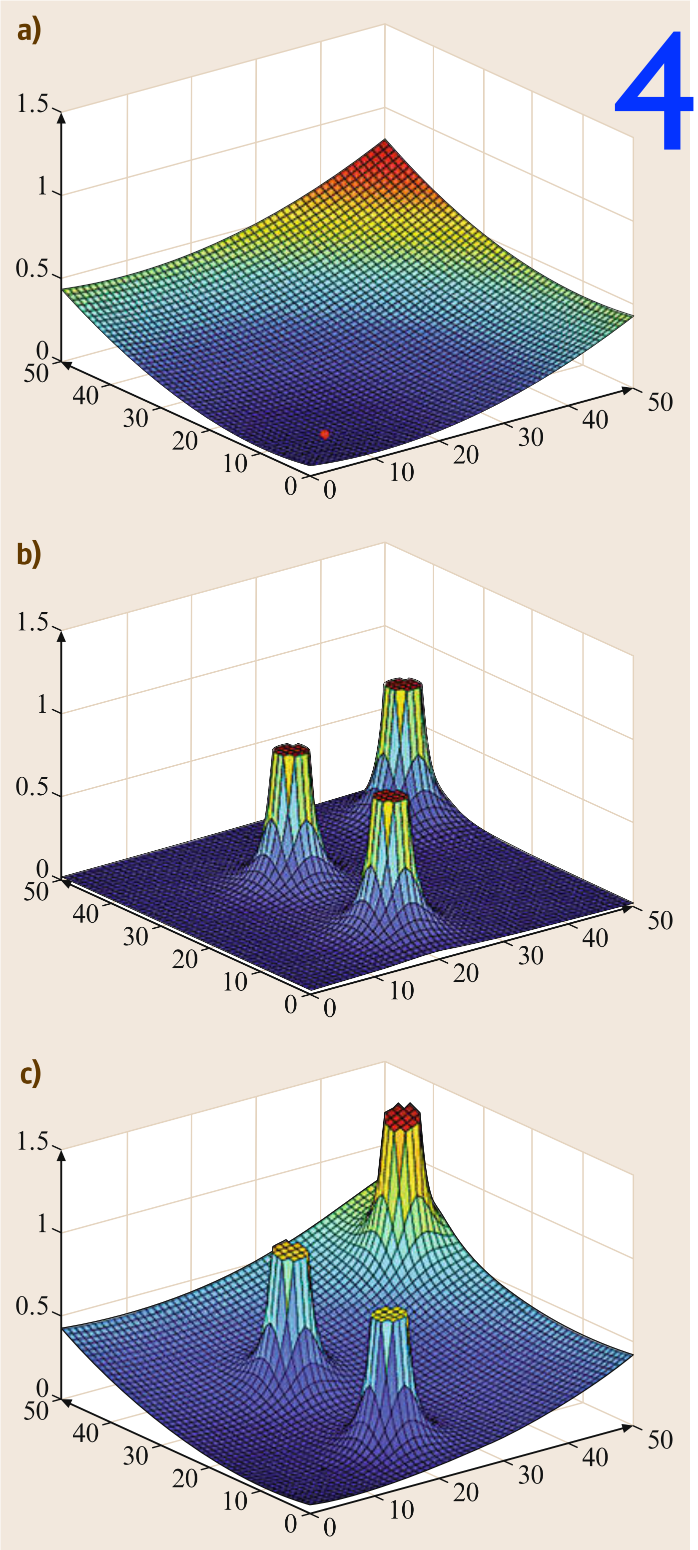
Potential field approach

target component

obstacle component

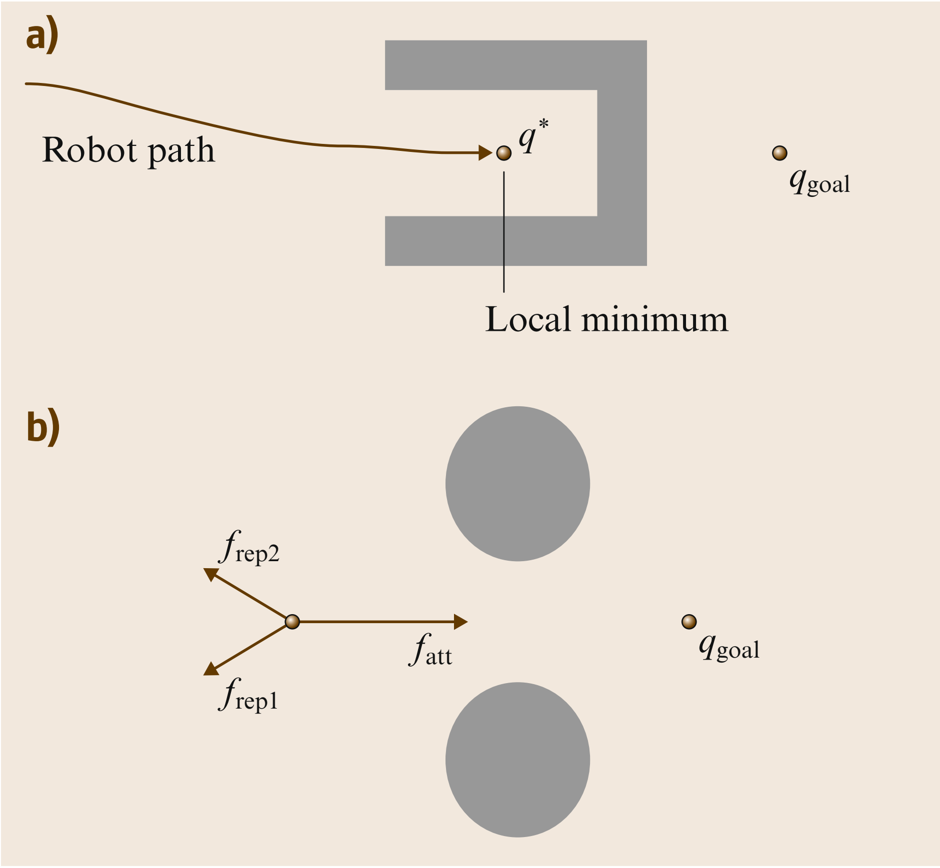
sum

[Kavraki, LaValle 2016]



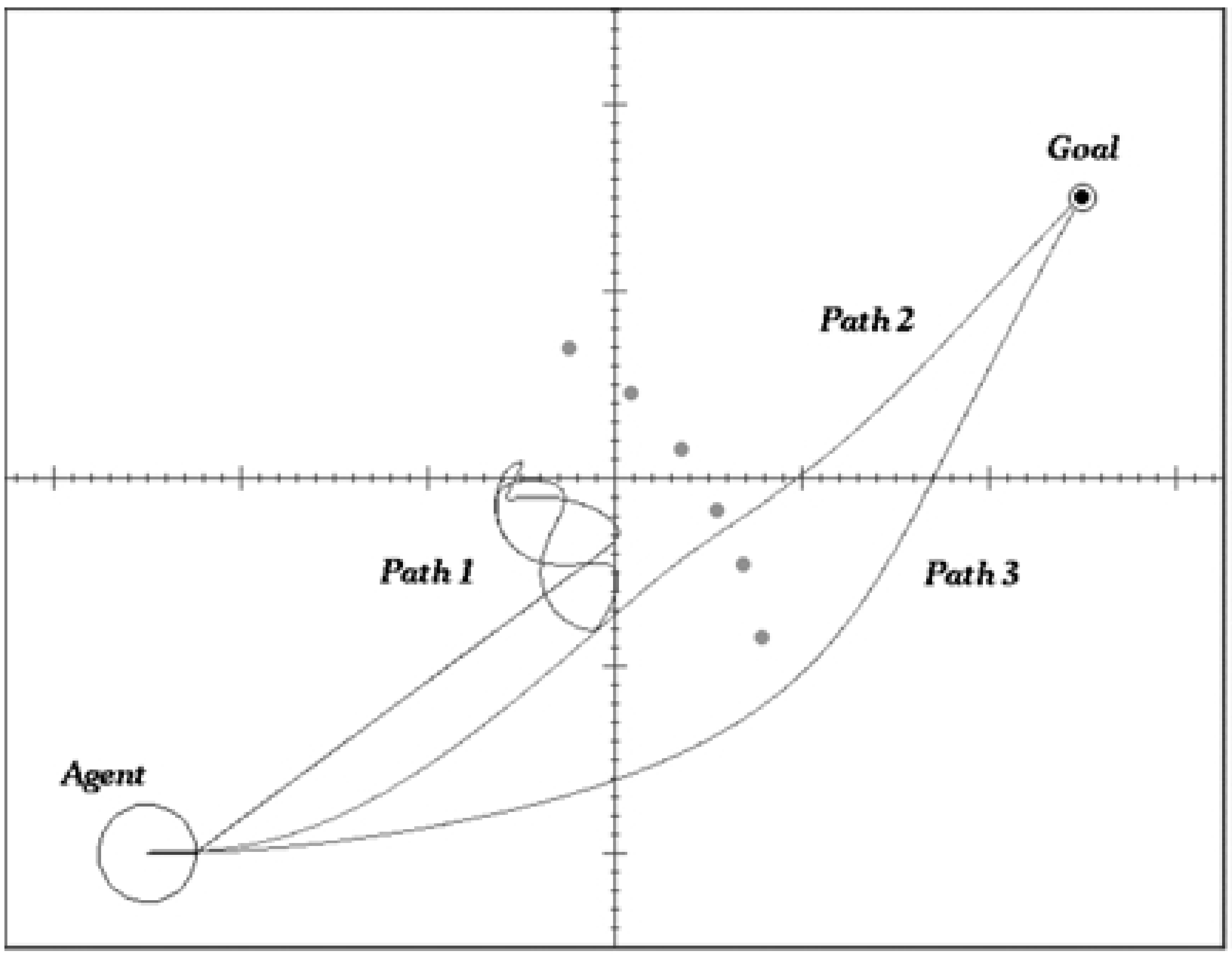
Potential field approach

- heuristic approach: no guarantee
- problem of local minima



[Kavraki, LaValle 2016]

spurious attractors in potential field approach vs. constraint violation in attractor dynamics approach



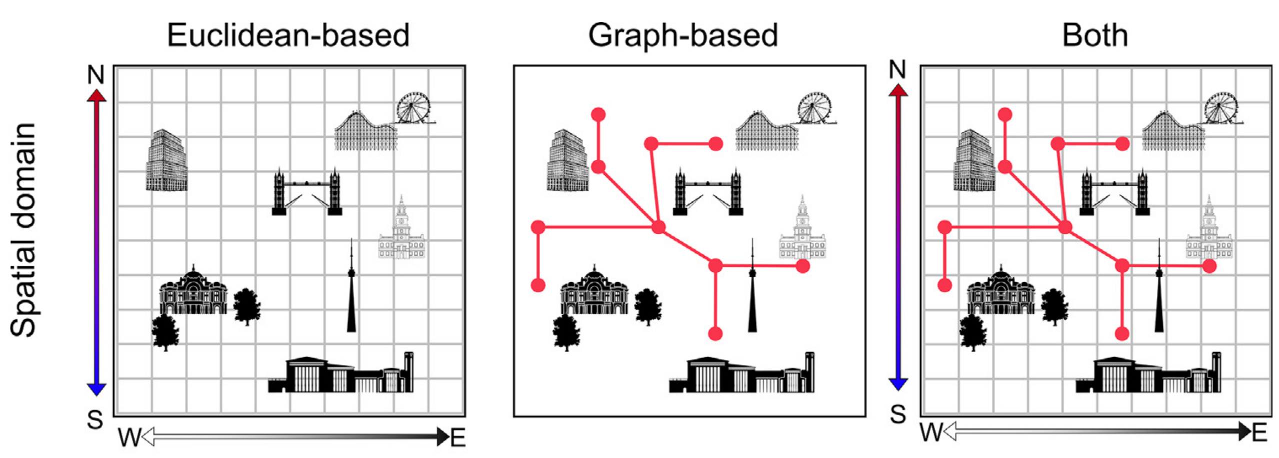
[Fajen et al. 2003]

potential field vs attractor dynamics

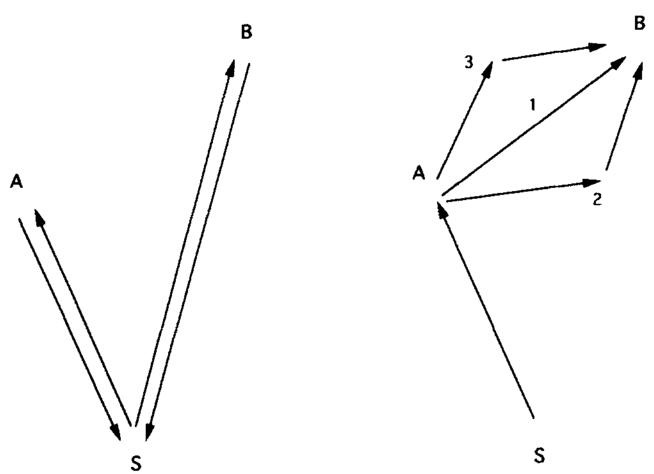
- potential field:
  - variables ~ position-like
  - attractor = target position
  - repelled from position of obstacle
  - motion plan = transient from any initial condition to target
- attractor dynamics:
  - variables~velocity-like
  - attractor = direction to target
  - repulsion from direction to obstacle
  - motion plan = sitting in (tracking) attractor that shifts as vehicle moves

Maps

- a 1:1 mapping from the world to a map that preserves angles/distances
- to use a map, need to know where you are on the map: ego-position estimation
- humans and animals use maps



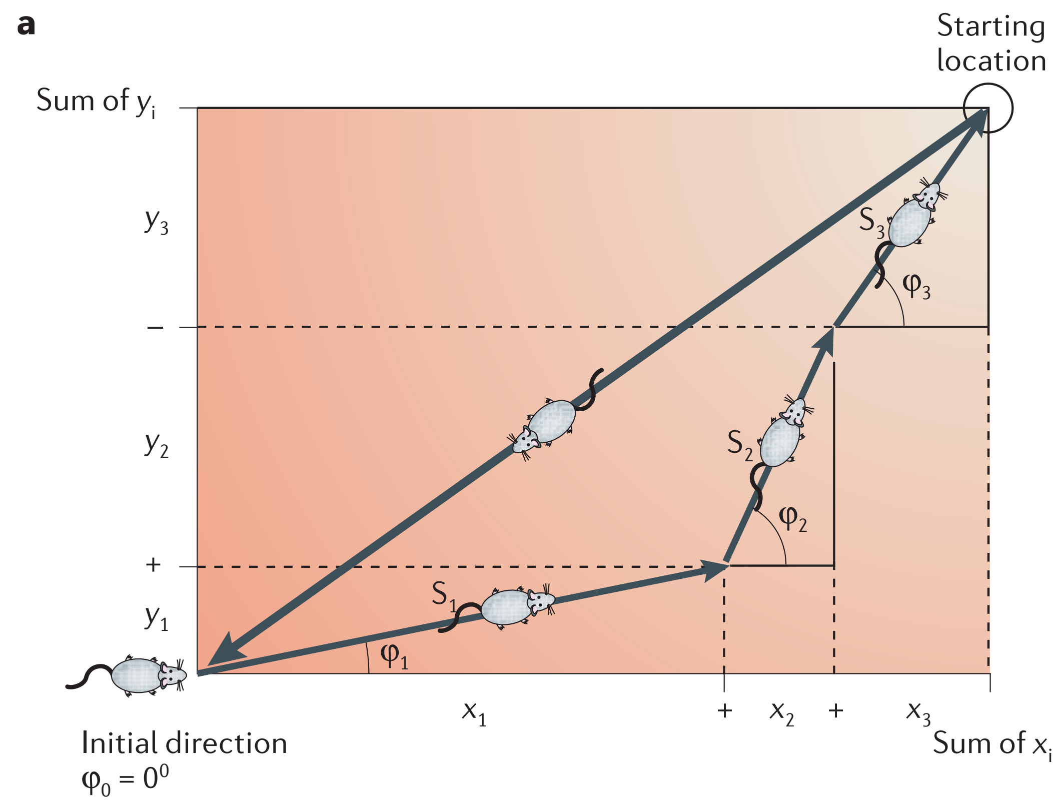
[Peer et al, 2020]



[Poucet, 1993]

Dead-reckoning/path integration

- simplest form of ego-position stimation
  - given current velocity=heading direction * speed
  - integrate (sum) to estimate change of position

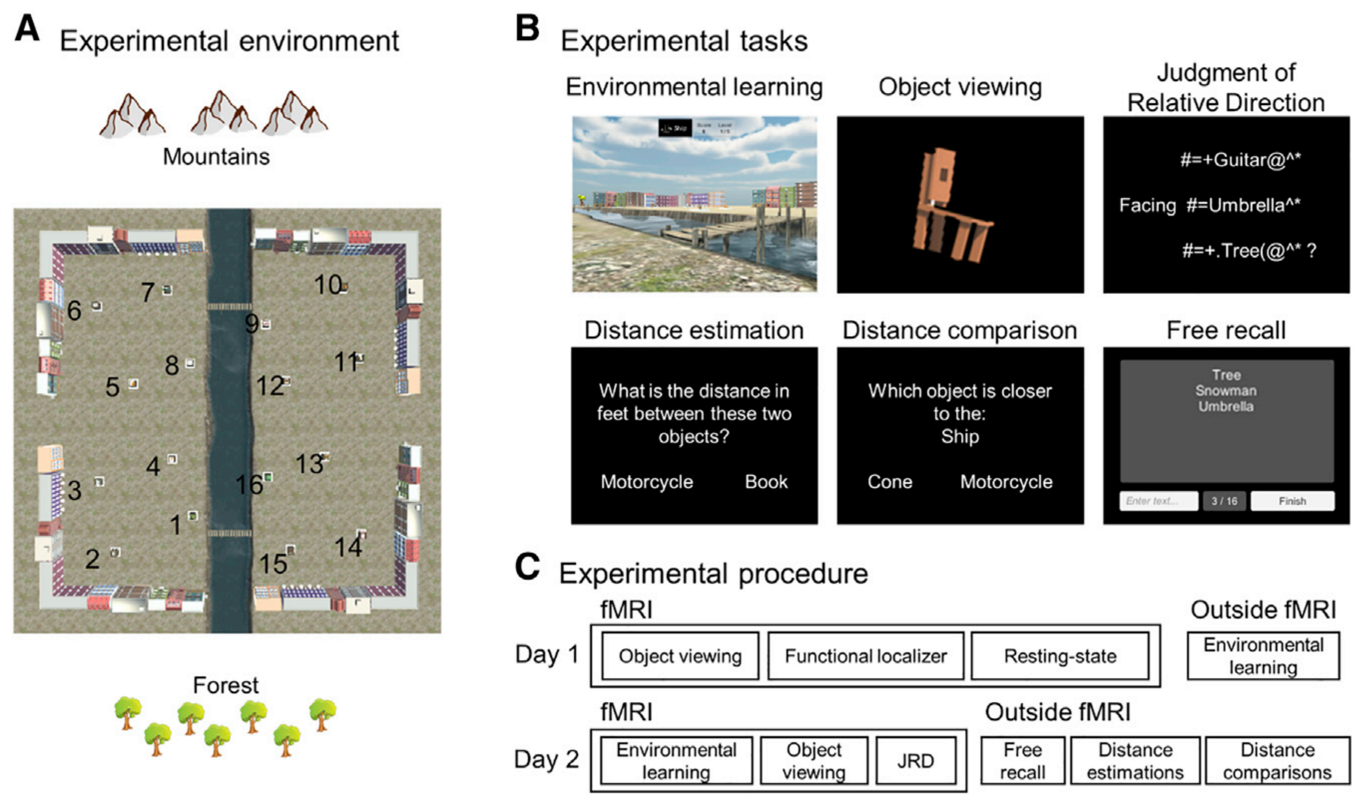


- but: errors accumulate … need to reset occasionally
- humans and animals do dead-reckoning

[McNaughton et al., *Nature reviews neuroscience* 2006]

Resetting ego-position by landmark recognition

- landmarks may be views, not necessarily objects…
- associated position with landmark => reset
- humans and animals use views



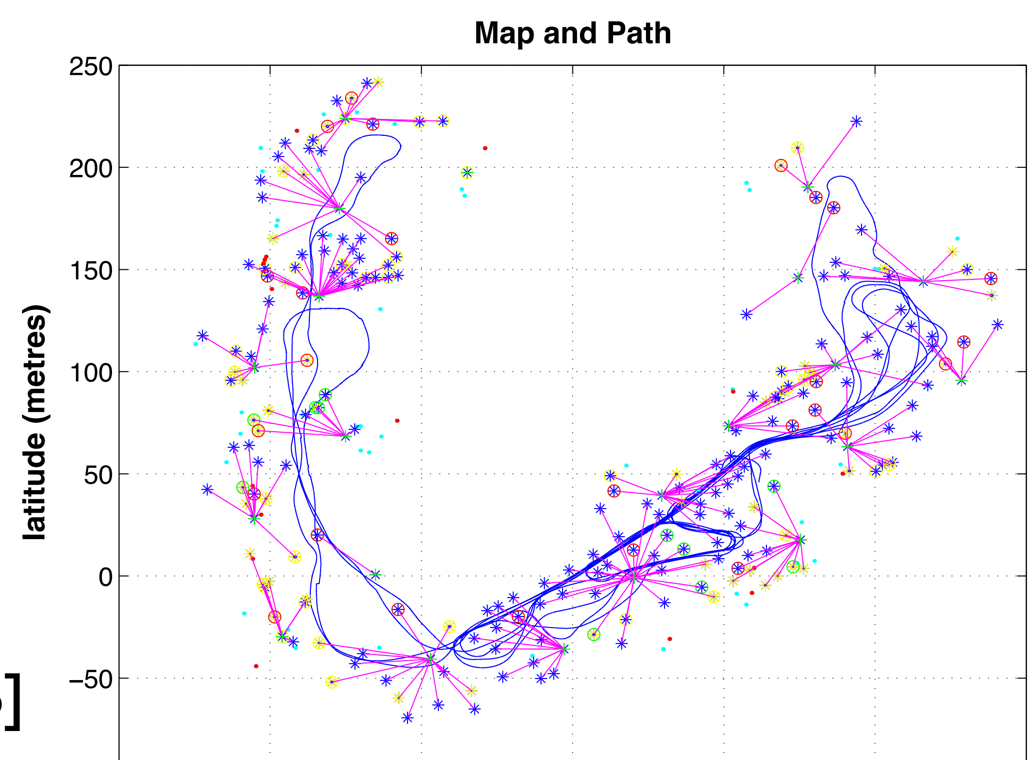
[Peer, Epstein, 2021]

SLAM

- Simultaneous Localization and Mapping

- optimize path integration
- associate path integration position estimate to landmark information

- loop closure when same landmark is approached

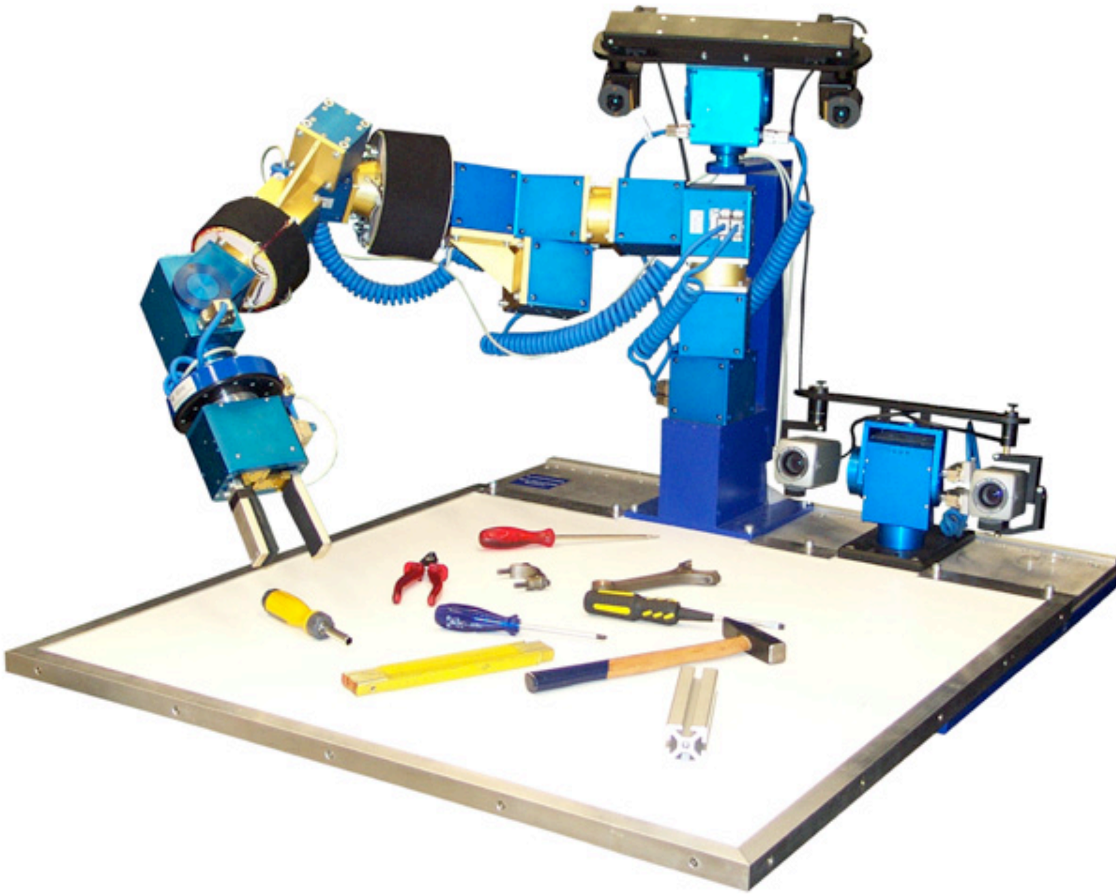


[Durrant-Whyte, Baily, 2006]

Spaces for robotic motion planning

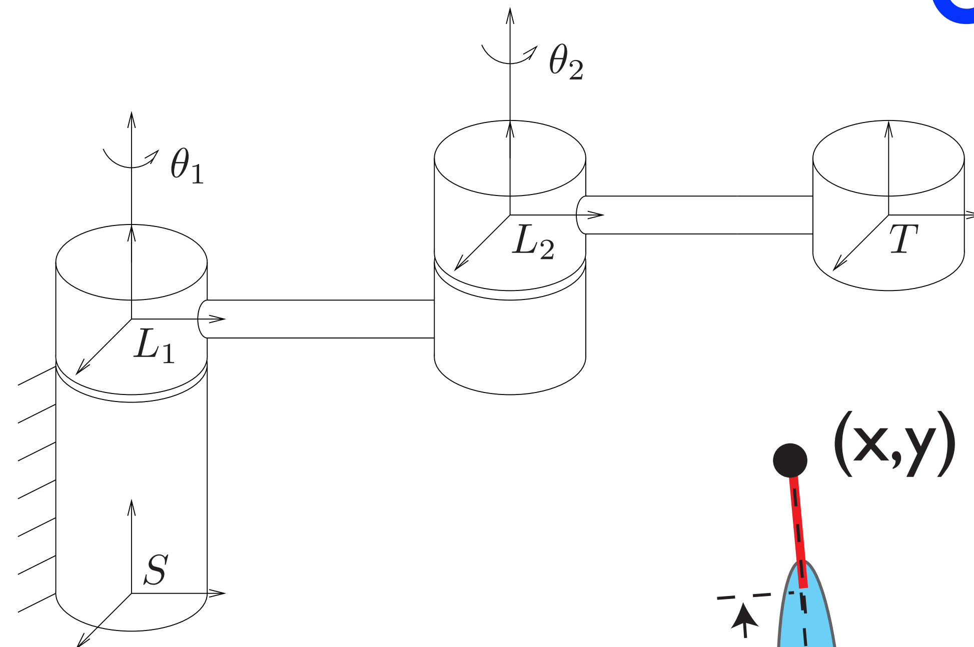
		differential form
forward kinematics	$\mathbf{x} = \mathbf{f}(\theta)$	$\dot{\mathbf{x}} = \mathbf{J}(\theta)\dot{\theta}$
inverse kinematics	$\theta = \mathbf{f}^{-1}(\mathbf{x})$	$\dot{\theta} = \mathbf{J}^{-1}(\theta)\dot{\mathbf{x}}$

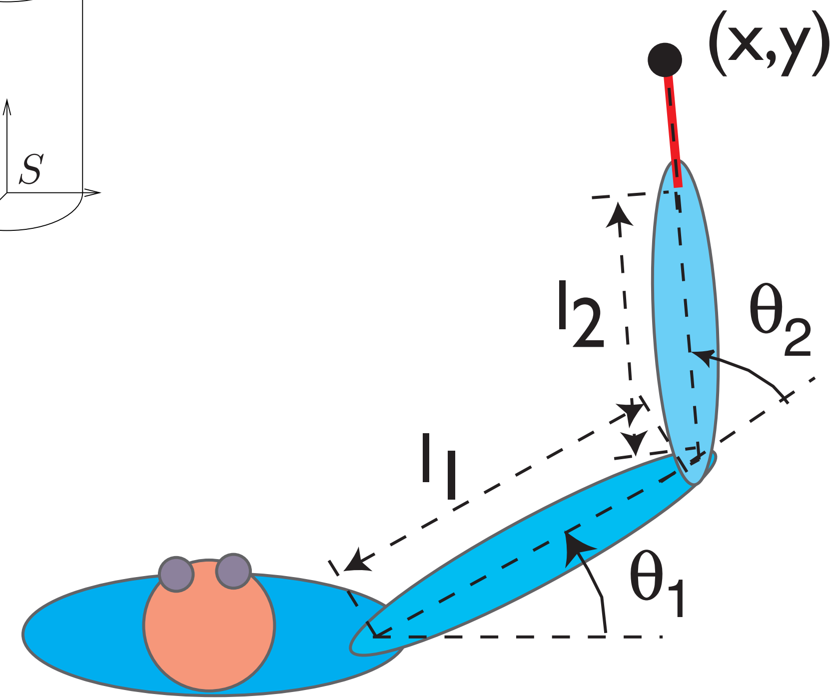
- need inverse kinematics to translate task demands into joint configurations that can be controlled



Forward kinematics

[Murray, Li, Sastry1994]





- where is the hand, given the joint angles..

$$\mathbf{x} = \mathbf{f}(\theta)$$

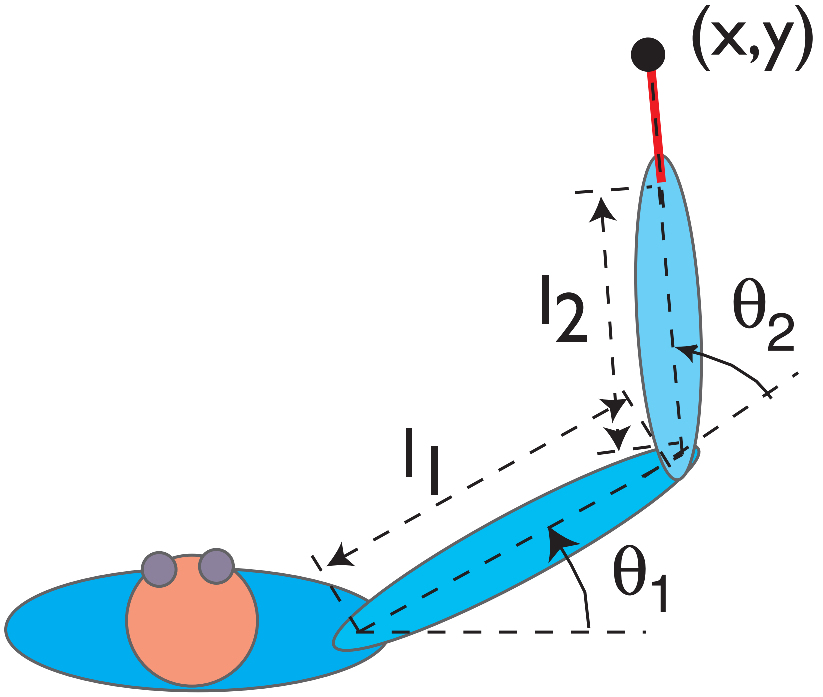
$$x = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)$$

$$y = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2)$$

Inverse kinematics

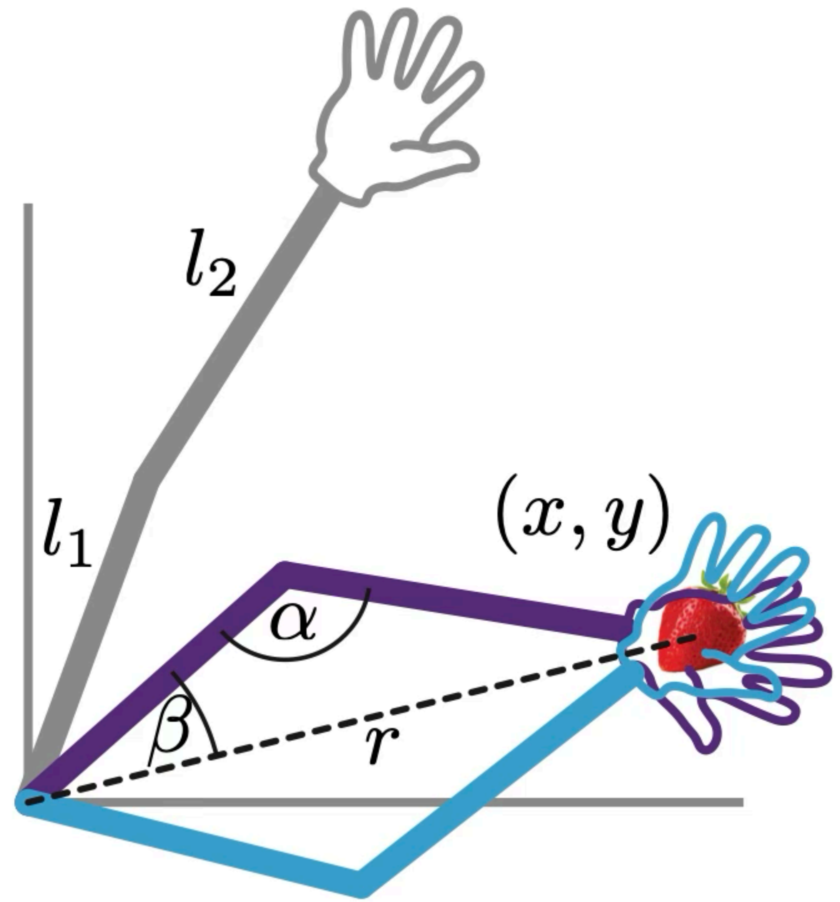
- what joint angles are needed to put the hand at a given location
- exact solution:

$$\theta = \mathbf{f}^{-1}(\mathbf{x})$$



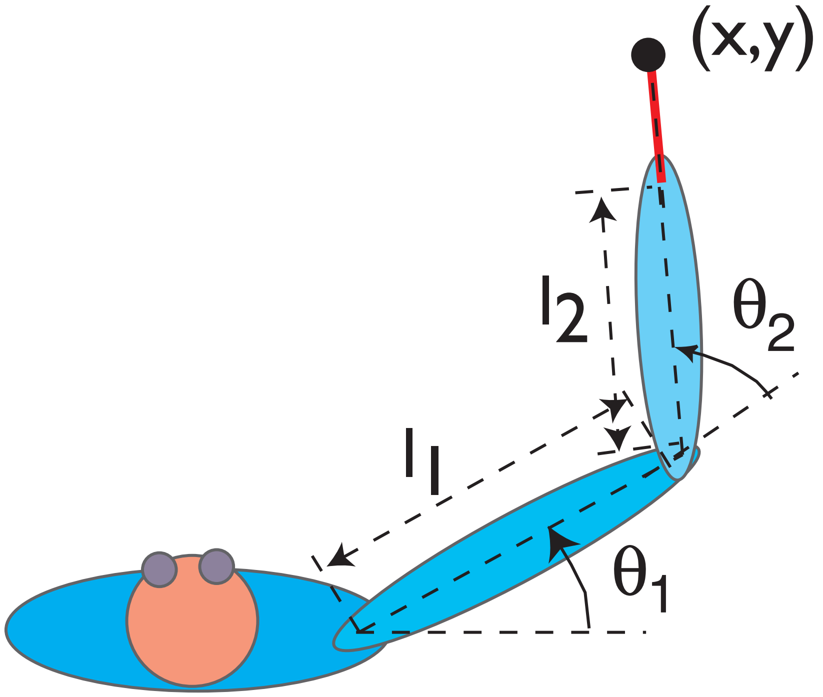
Inverse kinematics

- problem of multiple “leaves” of the invers kinematics



Differential inverse kinematics

- which joint velocities to move the hand in a particular way

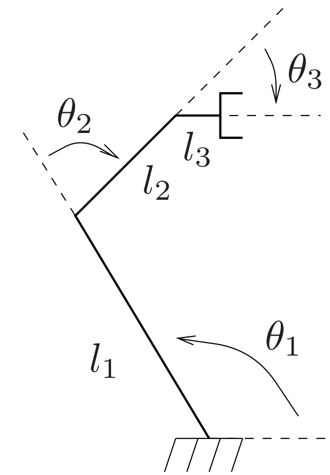


$$\dot{\theta} = \mathbf{J}^{-1}(\theta)\dot{\mathbf{x}}$$

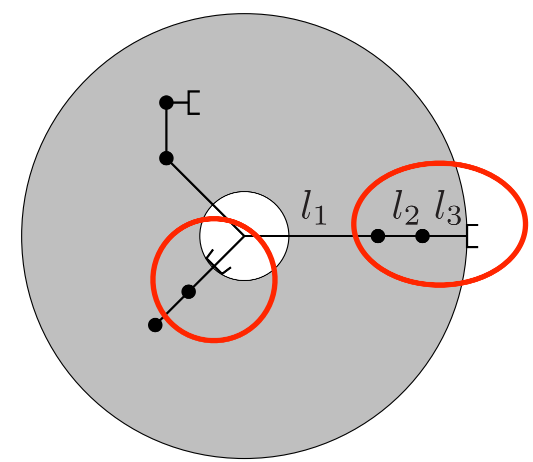
with the inverse, $\mathbf{J}^{-1}$, of $\mathbf{J}$, if it exists

Workspace / Singularities

- where the (real part of the)Eigenvalue of the Jacobian becomes zero
- => movement in a particular direction is not possible…
- typically at extended postures or inverted postures at limits of workspace



(a)



(c)

Redundant kinematics

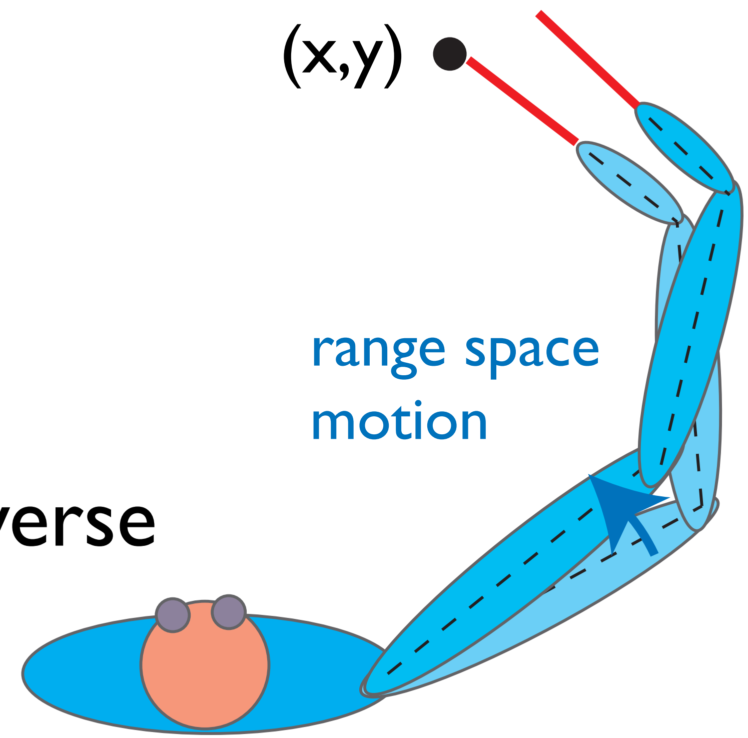
- use pseudo-inverses that minimize a functional (e.g., total joint velocity or total momentum)

$$\dot{\mathbf{x}} = \mathbf{J}(\theta)\dot{\theta}$$

$$\dot{\theta} = \mathbf{J}^{+}(\theta)\dot{\mathbf{x}}$$

$$\mathbf{J}^{+}(\theta) = \mathbf{J}^{T}(\mathbf{J}\mathbf{J}^{T})^{-1}$$ pseudo-inverse

minimizes $\dot{\theta}^2$



Timing

- generate movements that are "timed", that is,
  - they arrive "on time"
  - the are coordinated across different effectors
  - the are coordinated with moving objects (e.g., catching)
- timing implies some form of anticipation…

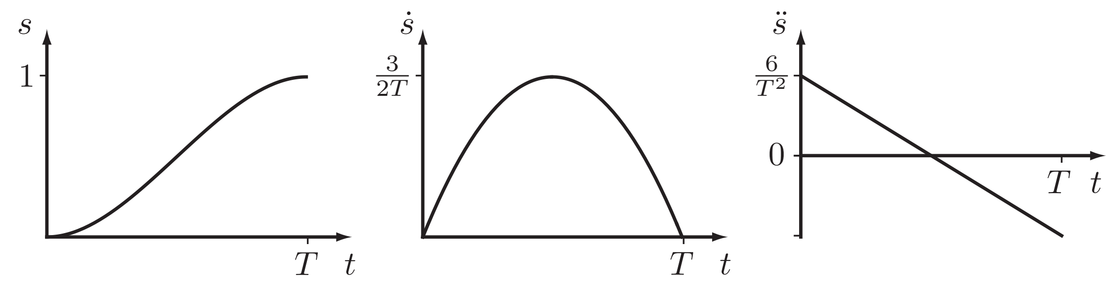
Conventional robotic timing

- time scaling

$$s(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3.$$

$$X(s) = X_{\text{start}} + s(X_{\text{end}} - X_{\text{start}}), s \in [0, 1].$$

$$\theta(s) = \theta_{\text{start}} + s(\theta_{\text{end}} - \theta_{\text{start}}),$$



- compute parameters to achieve a particular movement time T, with zero velocity at target

[Lynch, Park, 2017 (Chapter 9)]

Human movement is timed

- rhythmic:
  - locomotion, interlimb and intralimb
  - speaking
  - mastication
  - music production

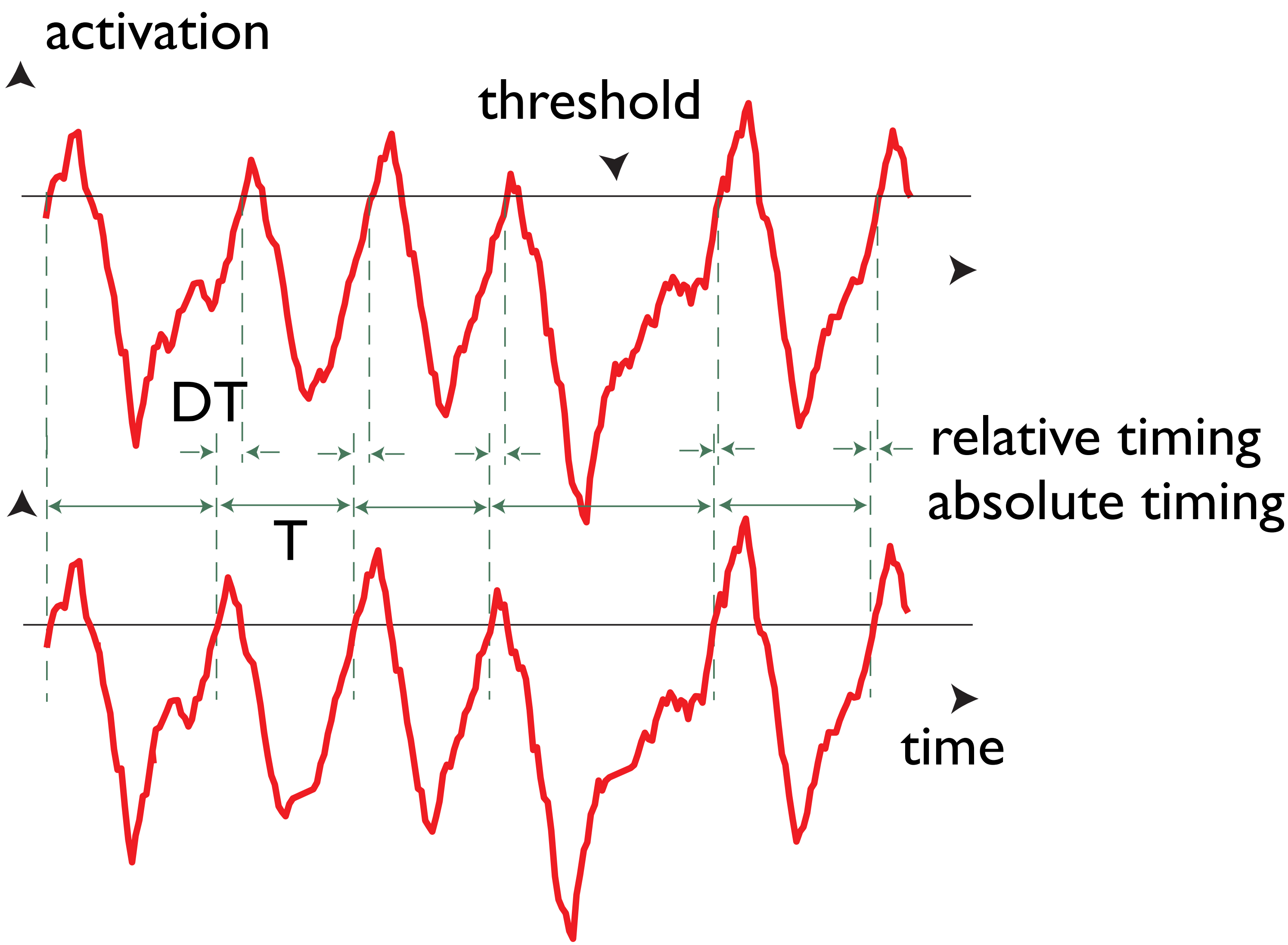
Human movement is timed

- temporally discrete:
  - reaching and grasping
  - coordination among fingers during grasp
  - catching, intercepting
  - bimanual manipulation

Coordination

- the maintenance of stable timing relationships between components of voluntary movement.
- => resists change/perturbation

Relative vs. absolute timing



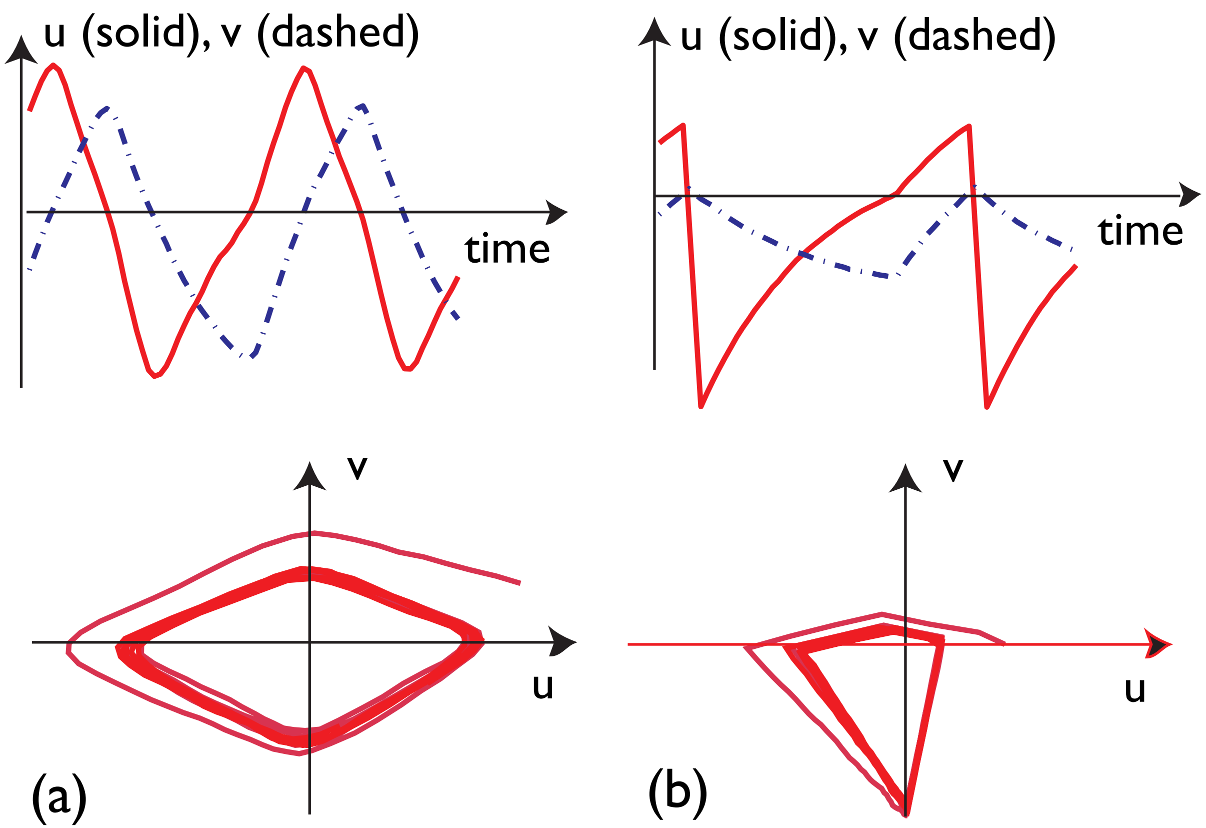
relative phase=DT/T

Account for timing in human movment

- (neural) oscillator autonomously generates timing signal, from which timing events emerge
- => limit cycle oscillators = clocks

Neural oscillator

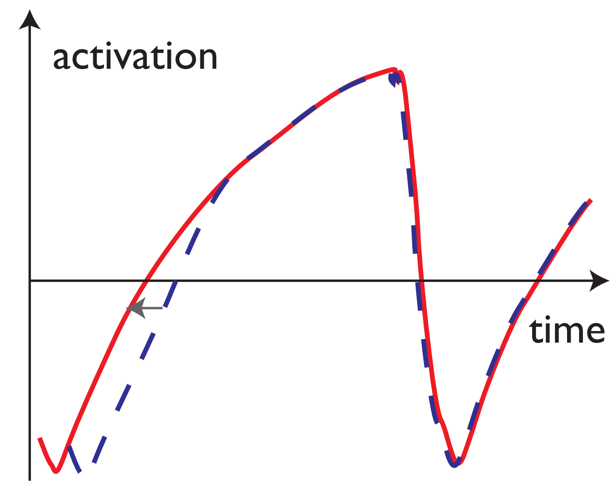
$$\tau \dot{u} = -u + h_u + w_{uu} f(u) - w_{uv} f(v)$$
$$\tau \dot{v} = -v + h_v + w_{vu} f(u),$$

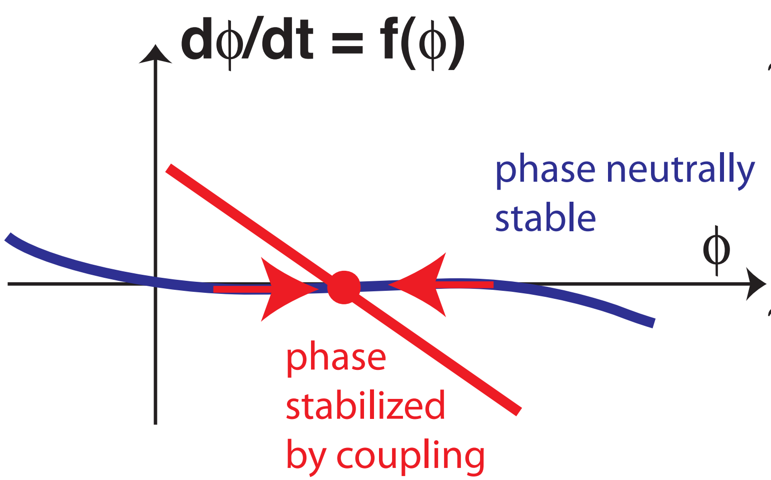


[Amari 77]

Coordination from coupling

- coordination=stable relative timing emerges from coupling of neural oscillators





$$\tau \dot{u}_1 = -u_1 + h_u + w_{uu} f(u_1) - w_{uv} f(v_1)$$

$$\tau \dot{v}_1 = -v_1 + h_v + w_{vu} f(u_1) \boxed{+ c f(u_2)}$$

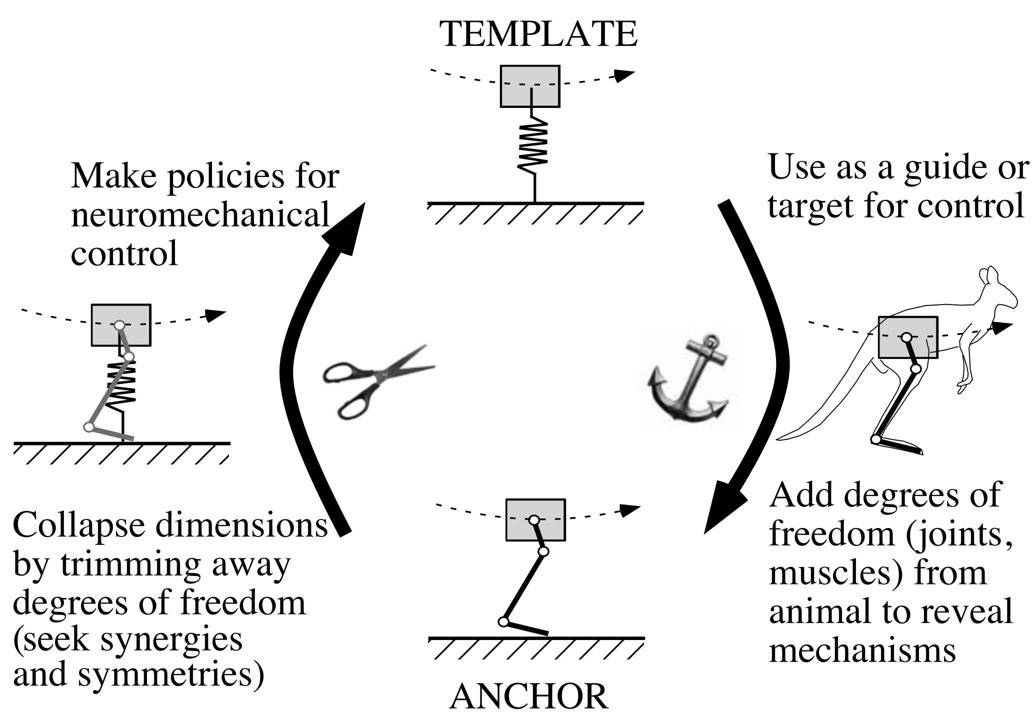
$$\tau \dot{u}_2 = -u_2 + h_u + w_{uu} f(u_2) - w_{uv} f(v_2)$$

$$\tau \dot{v}_2 = -v_2 + h_v + w_{vu} f(u_2) \boxed{+ c f(u_1)}$$

[Schöner: Timing, Clocks, and Dynamical Systems. *Brain and Cognition* 48:31-51 (2002)]

Timing in autonomous robotics

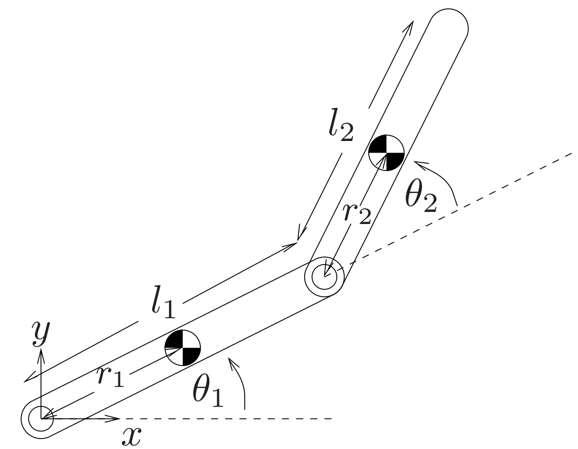
- borrow basic idea from human (animal) movement:
- coupled oscillators that are turned on/off



[Full Koditschek 99]

Kinetics and control

- to determine the equations of motion of a kinematic chain, consider the constraints that reduce the effective numbers of degrees of freedom. . .
- => generalized coordinates
- => Lagrange formulation of dynamics



n Newtonian coordinates r_i

m generalized coordinates q_i

n-k=m

k constraints

$$r_i = f_i(q_1, \ldots, q_m) \quad \Longleftrightarrow \quad g_j(r_1, \ldots, r_n) = 0$$

$$i = 1, \ldots, n \qquad\qquad j = 1, \ldots, k.$$

Lagrangian dynamics of an open-chain manipulator

$$\boxed{M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + N(\theta, \dot{\theta}) = \tau}$$

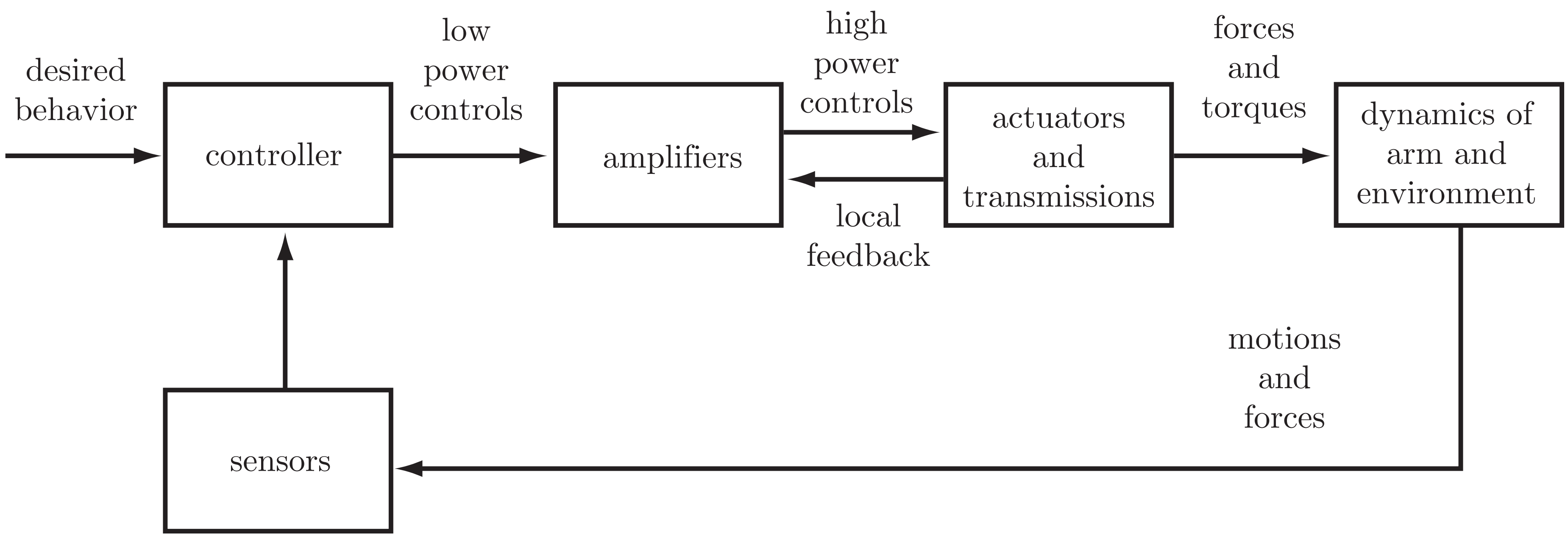
inertial

centrifugal/ coriolis

gravitational

active torques

Robotic control



(a)

[Lunch, Park, 2017]

Motion control single joint

- $\tau = M\ddot{\theta} + mgr\cos(\theta) + b\dot{\theta}$
- feedback PID controller
- $\tau = K_p\theta_e + K_d\dot{\theta}_e + K_i \int \theta(t')dt'$

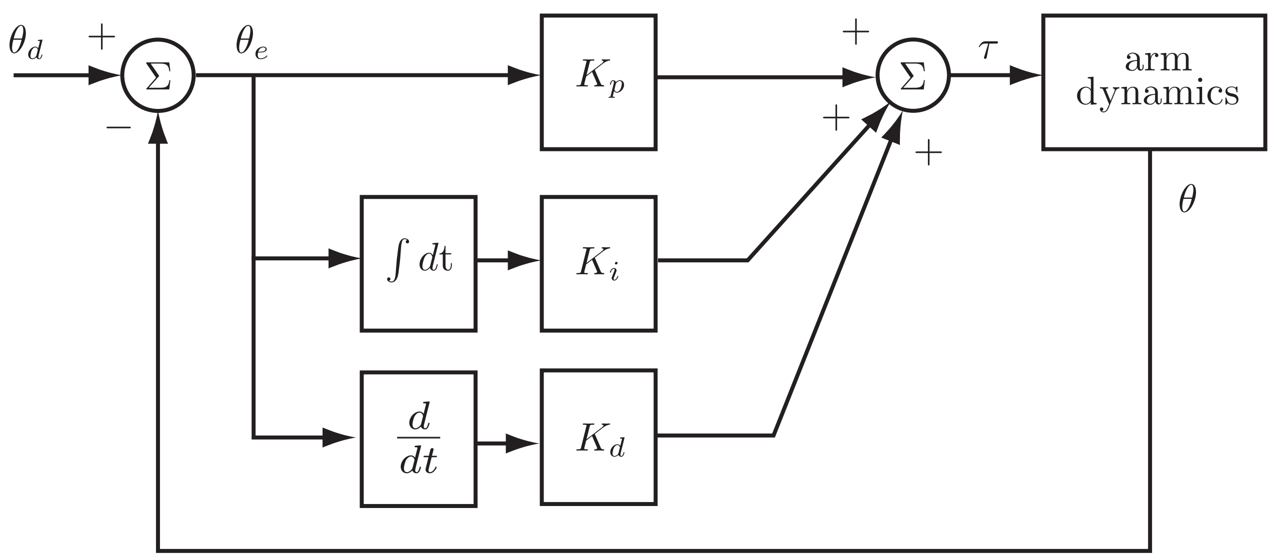


Figure 11.12: Block diagram of a PID controller.

[Lunch, Park, 2017]

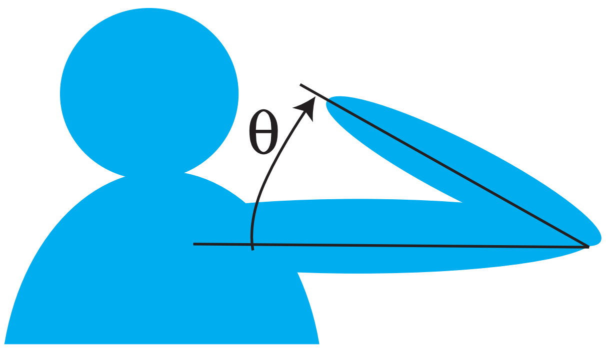
Control of multi-joint arm

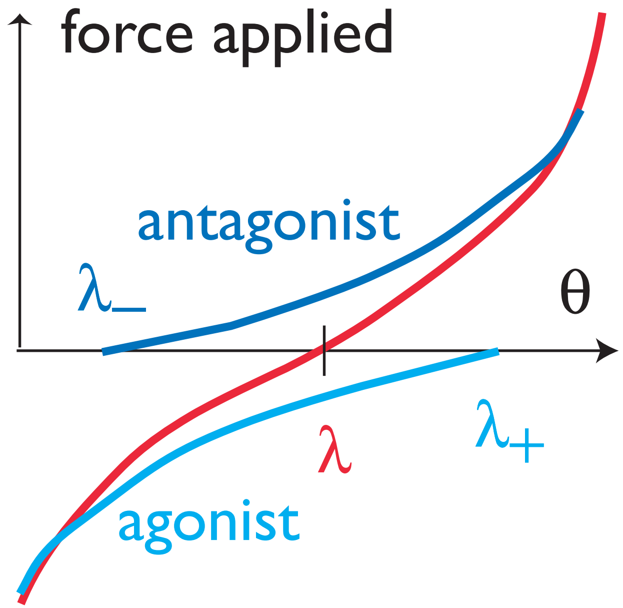
- generate joint torques that produce a desired motion… θ_d
- error $\theta_e = \theta - \theta_d$
- PID control $\tau = K_p\theta_e + K_e\dot{\theta}_d + K_i\int\theta_e(t')dt'$
- => controlling joints independently

$$\boxed{M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + N(\theta,\dot{\theta}) = \tau}$$

Human motor control

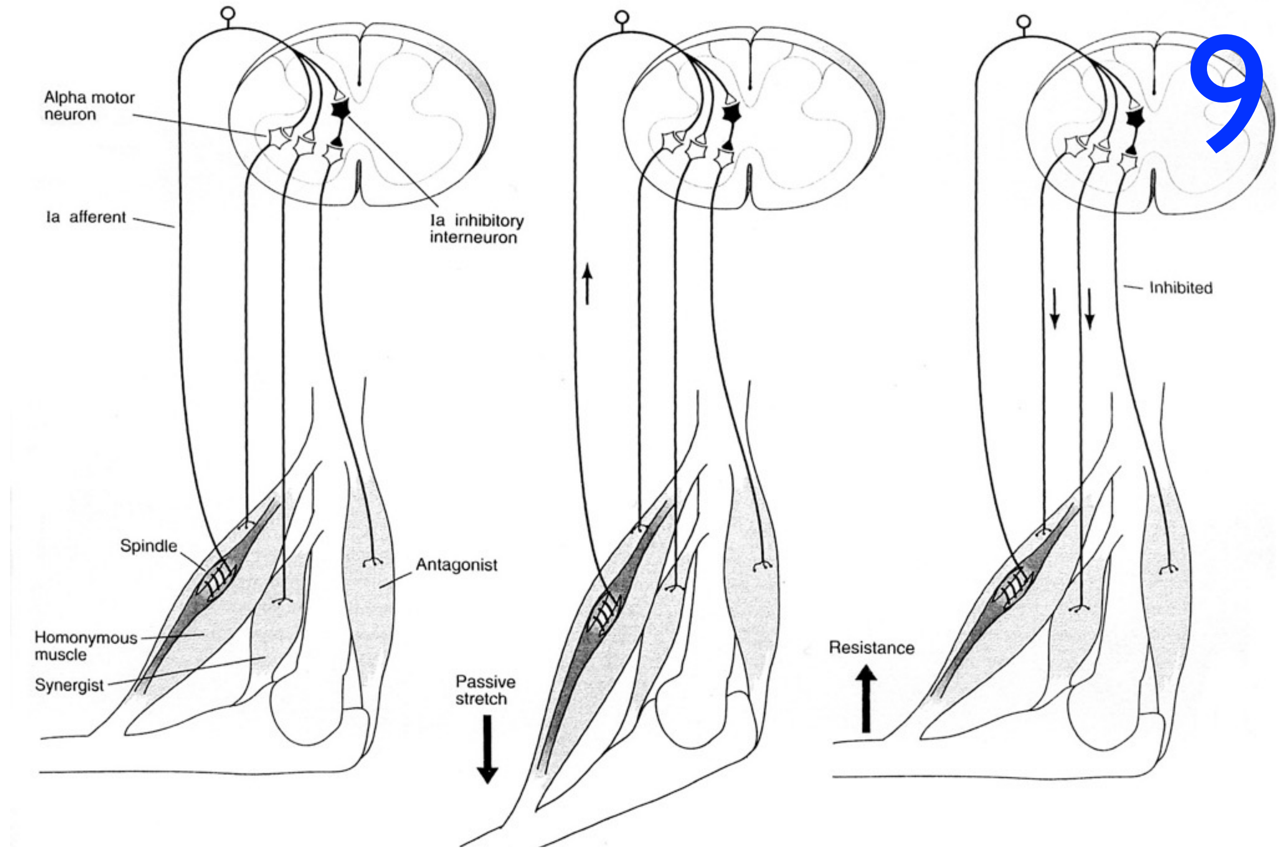
- posture resists when pushed => is actively controlled = stabilized by feedback
- invariant characteristic
  - one lambda per muscle
  - co-contraction controls stiffness

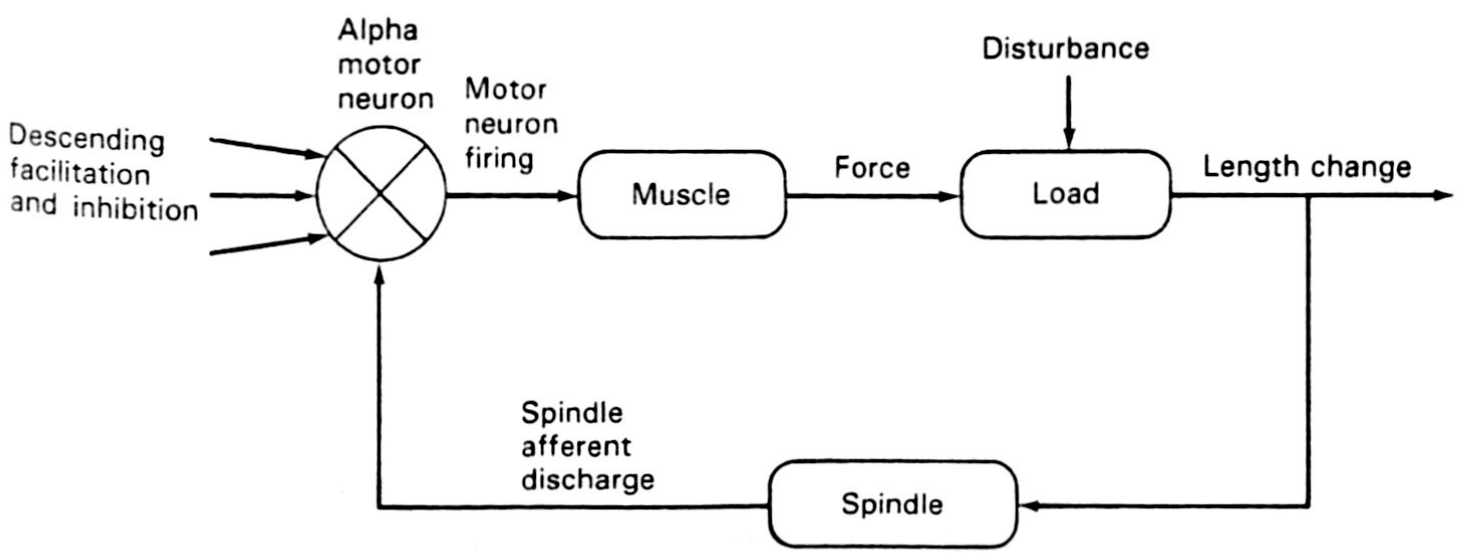




based on spinal reflexes

- stretch reflex





[Kandel, Schartz, Jessell, Fig. 37-11]

Learning experience

- interdisciplinary experience: using analogies with nervous systems to design/think about autonomous robots
- learn concepts from dynamical systems theory
- experience the reading and writing of mathematical/technical material