Stochastic Processes, Kalman Filtering and Stochastic Control

Dejan Milutinović, UC Santa Cruz

Hellman Fellow

http://people.ucsc.edu/~dmilutin/

dejan@soe.ucsc.edu

2013 ASME DSCC, October 21, Stanford

Stochastic Processes

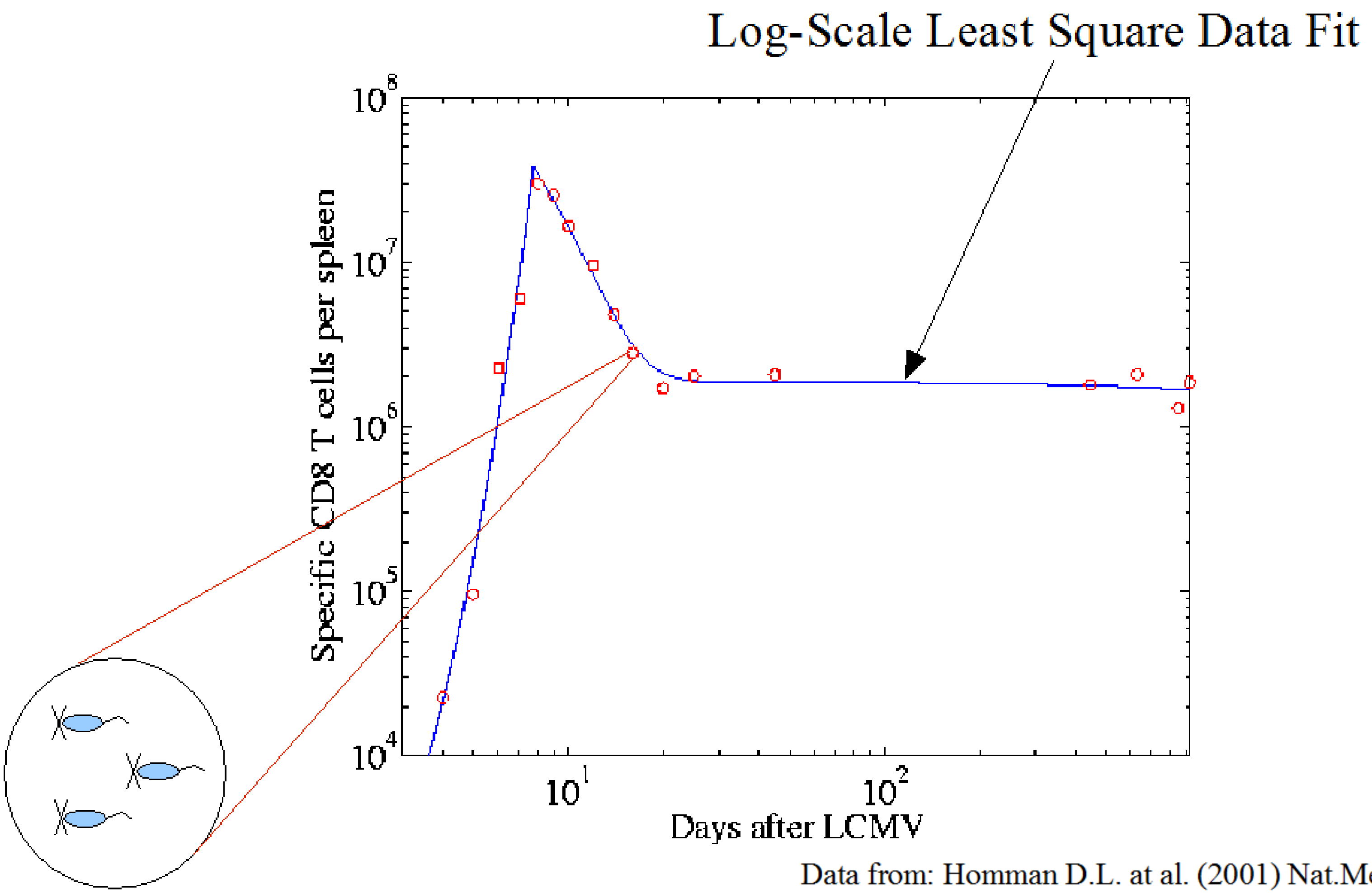
- Development of stochastic process theory is from the very beginning in connection with biology (e.g. Brownian motion).
- In early days, it was assumed that a randomly moving micro-particle suspended in water moved because it was alive.
- Contradiction was reached when it was observed that some of certainly "dead" particles were moving in the same way.
- For many years, random motion was ignored. One reason was that it was considered unimportant. The other reason was that they needed new tools.

Good reading: "Uncertainty: Einstein, Heisenberg, Bohr, and the Struggle for the Soul of Science" by David Lindley

Outline

- Physical basis
- Stochastic differential equations
- Kalman filter projects
- Feedback stochastic optimal control in robotics
- Open-loop stochastic optimal control in robotics
- Recent results

Experiment and Data Fit



3 x 17 = 51 mice

Data from: Homman D.L. at al. (2001) Nat.Med.7:913

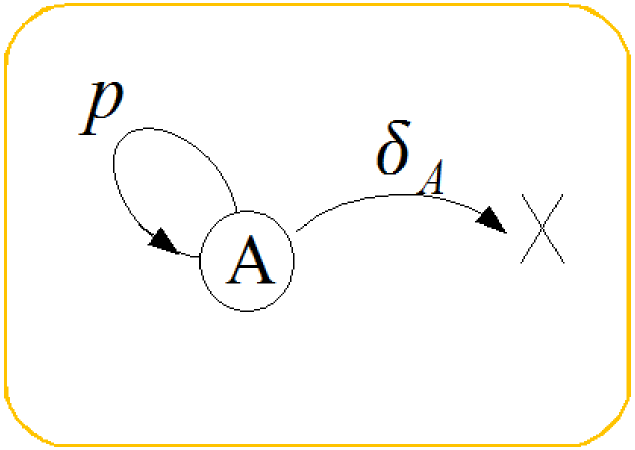
Published : De Boer R.J at al. (2003) J. Immunol., 171

Dynamical Model

- Expansion phase $t < T$

$$\dot{A}(t) = (p - \delta_A)A(t) = \rho A(t)$$

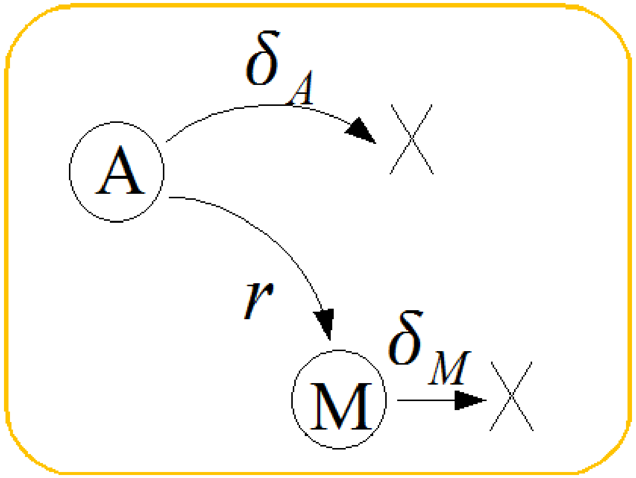
$$M(t) = 0$$



- Contraction phase $t > T$

$$\dot{A}(t) = -(r + \delta_A)A(t)$$

$$\dot{M}(t) = rA(t) - \delta_M M(t)$$

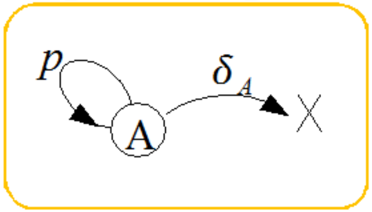


$$y(t) = A(t) + M(t)$$

Stochastic Differential Equation Model

(The chemical Langevin equation)

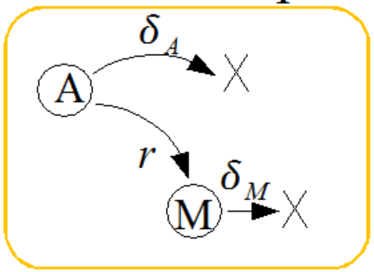
Expansion phase



$$dA = \rho A(t)\,dt + \sqrt{\rho A(t)}\,d\omega$$

$$\frac{dA(t)}{dt} = \rho A(t) + \sqrt{\rho A(t)}\,d\xi$$

Contraction phase



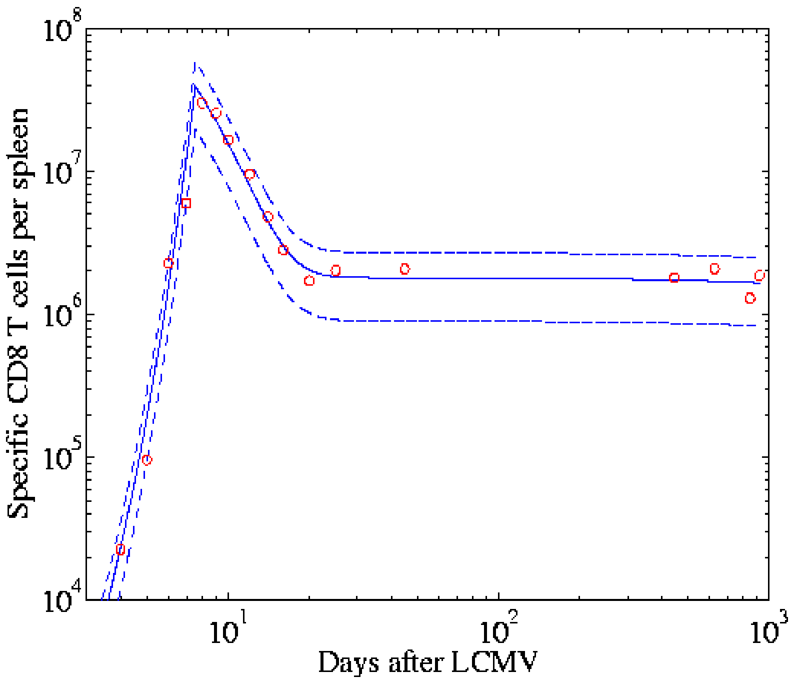
$$dA = -(r+\delta_A)A(t)\,dt - \sqrt{(r+\delta_A)A(t)}\,d\omega_1$$

$$dM = rA(t)\,dt - \delta_M M(t)\,dt + \sqrt{rA(t)}\,d\omega_1 - \sqrt{\delta_M M(t)}\,d\omega_2$$

$$\frac{dA(t)}{dt} = -(r+\delta_A)A(t) - \sqrt{(r+\delta_A)A(t)}\,\xi_1$$

$$\frac{dM(t)}{dt} = rA(t) - \delta_M M(t) + \sqrt{rA(t)}\,\xi_1 - \sqrt{\delta_M M(t)}\,\xi_2$$

Itô calculus can be used to predict the variance



Process noise + Multiplicative error

$$y(t_k) = y_p(t_k) + y_p(t_k)\theta_k$$

Process noise is dominant

$$\Theta = 0$$

Van Kampen, N. G., “Stochastic Processes in Physics and Chemistry”, Elsevier

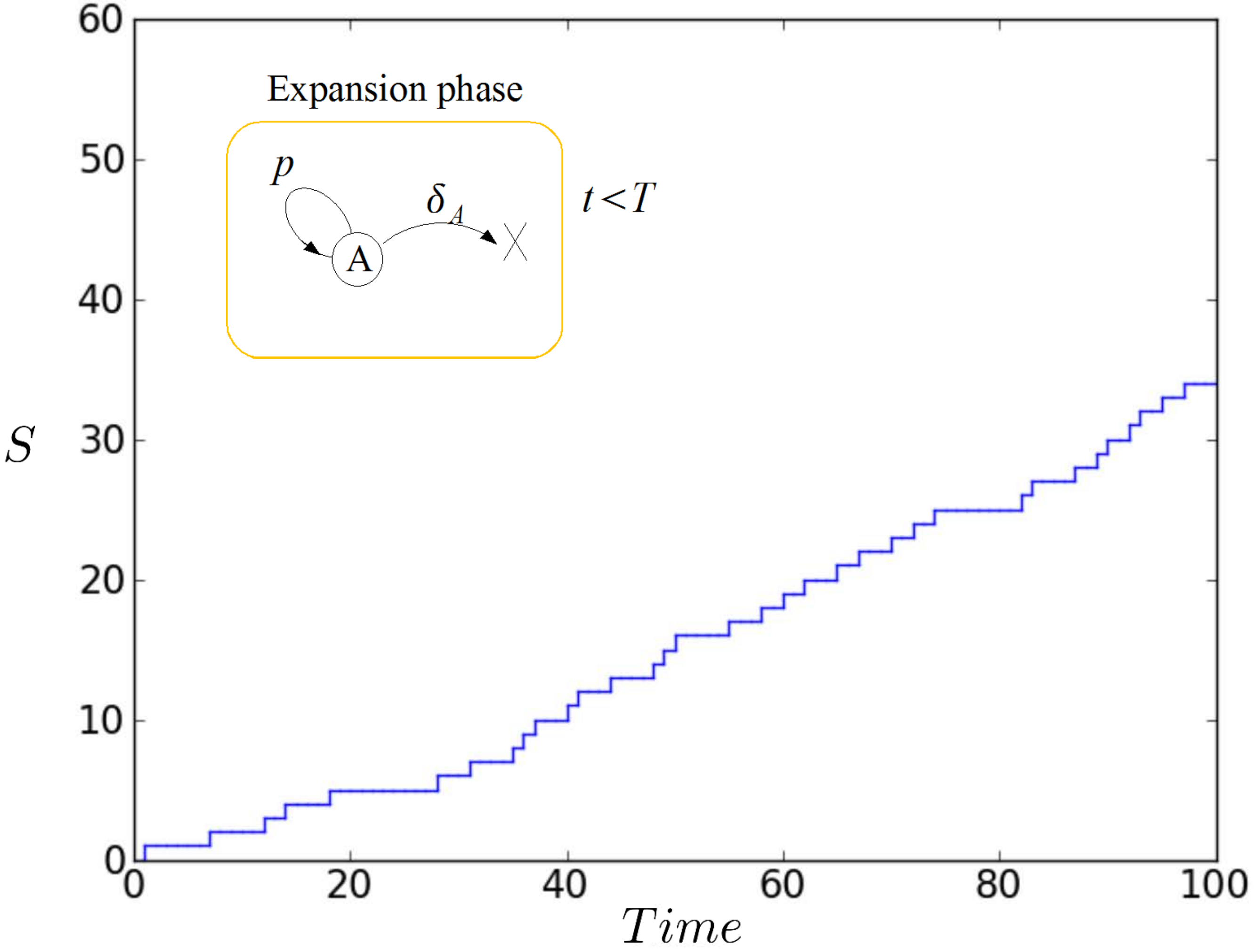
Gardiner, C., “Stochastic Methods: A Handbook for the Natural and Social Sciences”, Springer

Gillespie, D.T., “The Chemical Langevin Equation”, Journal of Chemical Physics, Vol. 113, pp.297-306, 2000

Milutinović, D., De Boer, R. J., Process Noise: An Explanation for the Fluctuations in the Immune Response During Viral Infection, Biophysical Journal, Vol. 92, pp. 3358-67, 2007

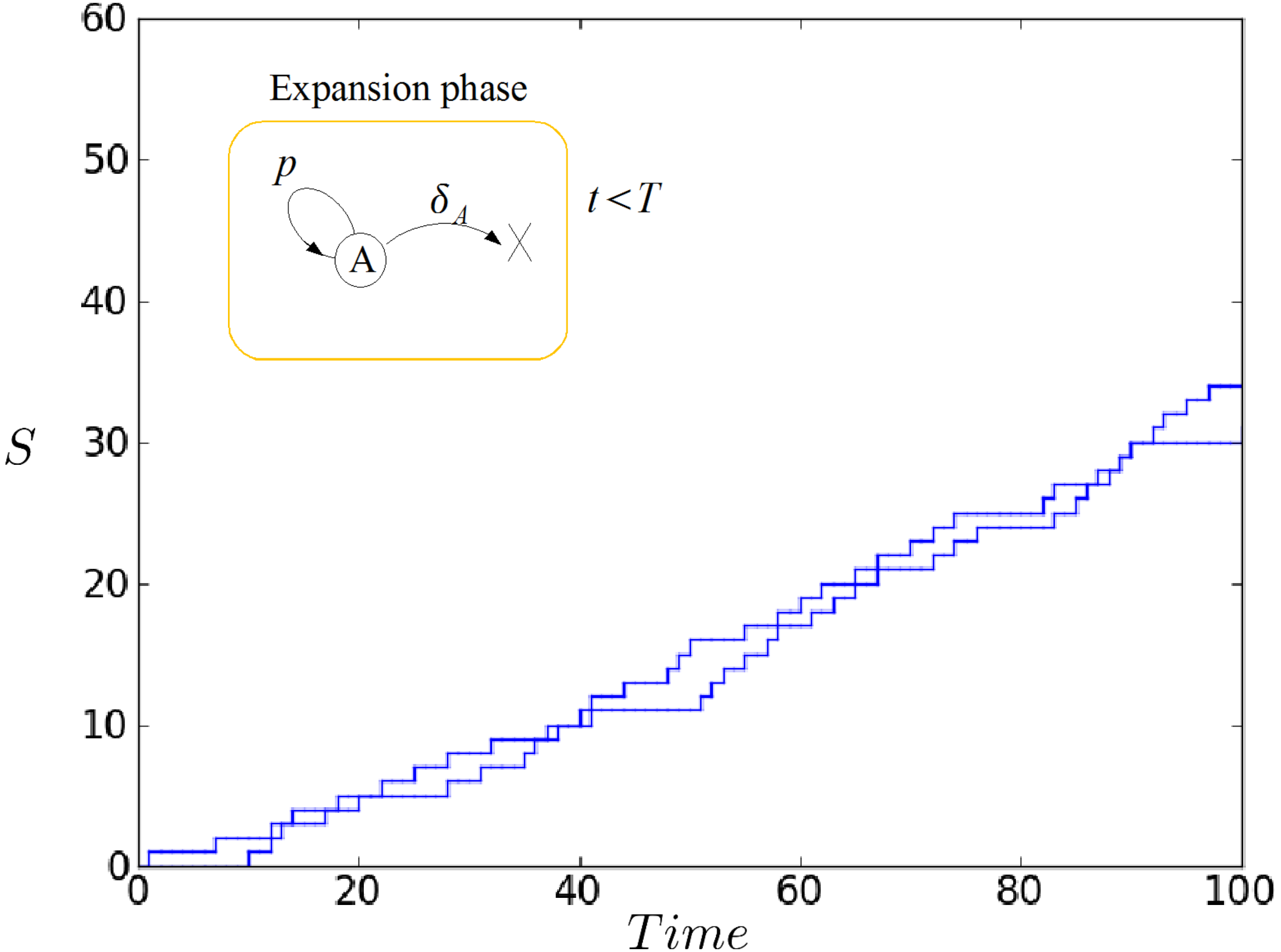
Binomial Distribution and Continuous Time Processes

$$S = \sum_{i=1}^{N} x_i \qquad x_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p, \quad \delta_A \approx 0 \end{cases}$$



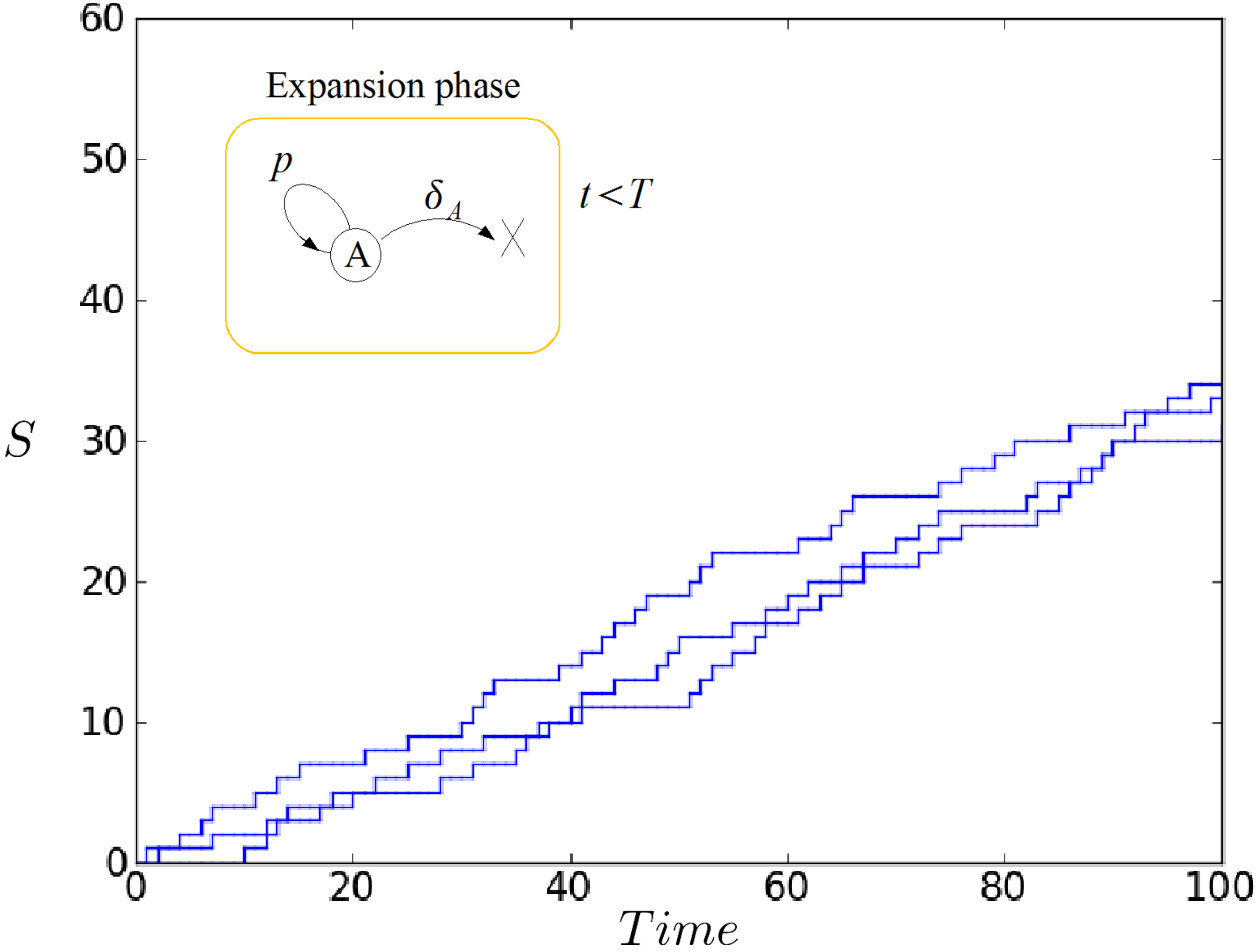
Binomial Distribution and Continuous Time Processes

$$S = \sum_{i=1}^{N} x_i \qquad x_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p, \quad \delta_A \approx 0 \end{cases}$$



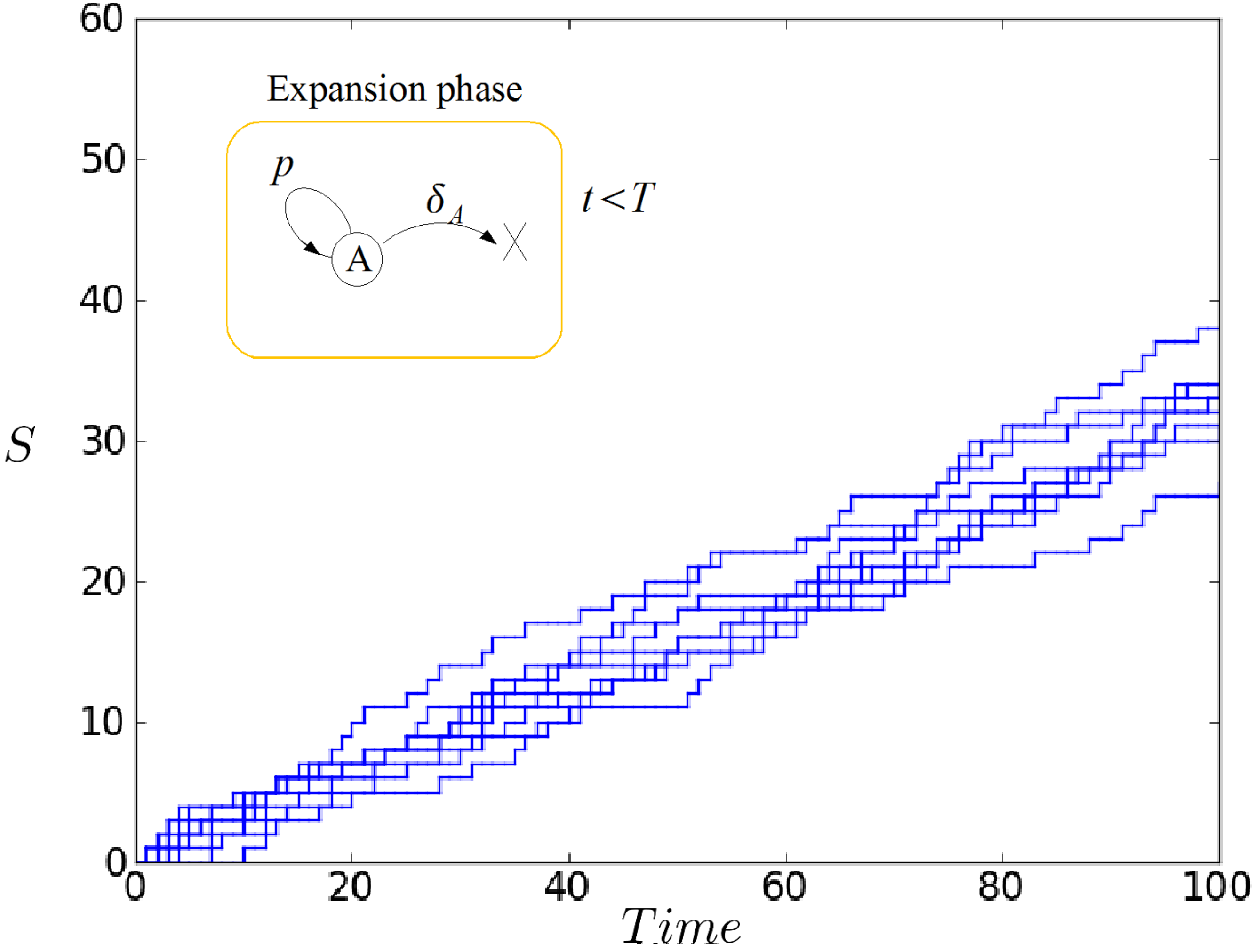
Binomial Distribution and Continuous Time Processes

$$S = \sum_{i=1}^{N} x_i \qquad x_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p, \quad \delta_A \approx 0 \end{cases}$$



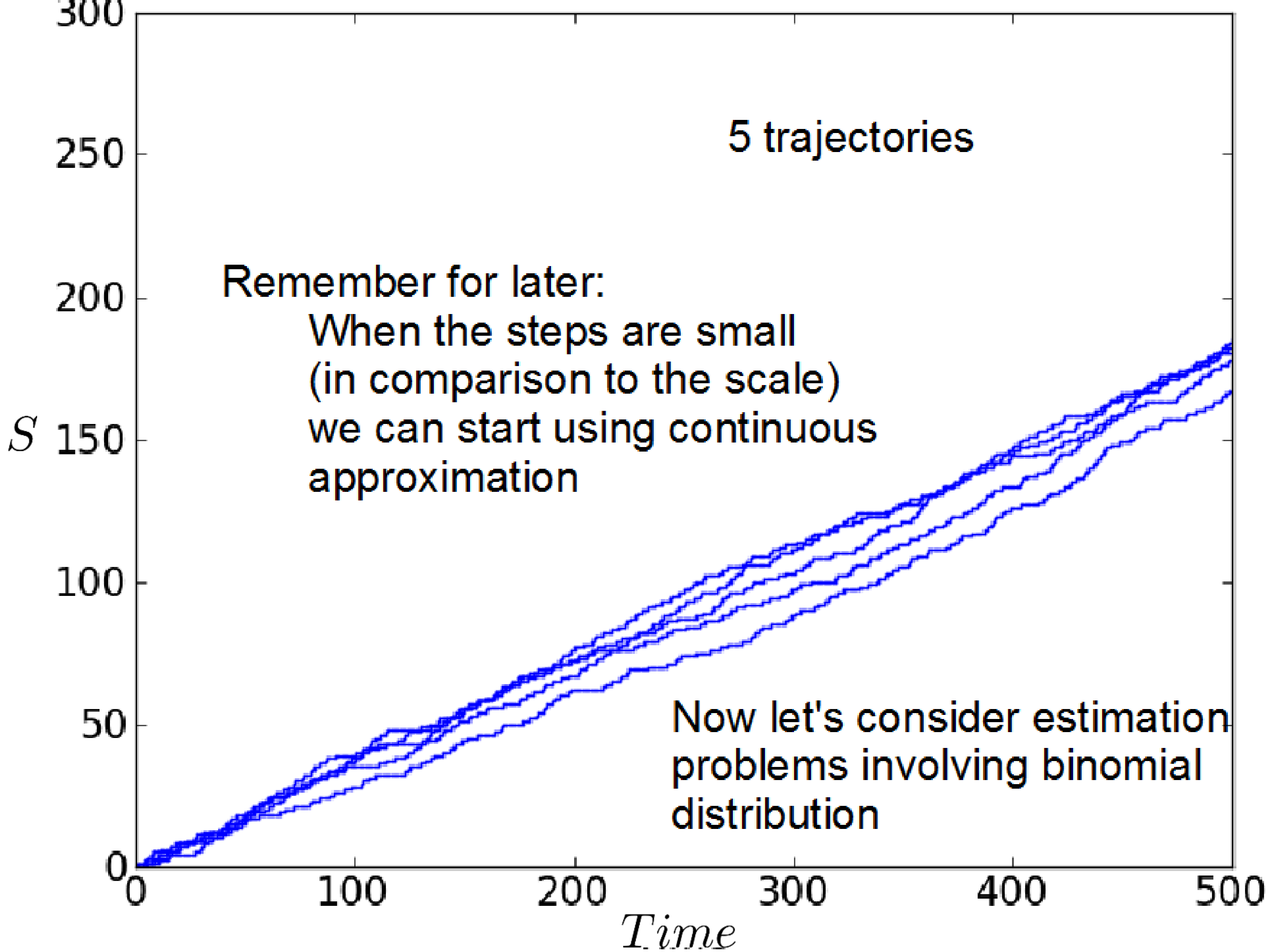
Binomial Distribution and Continuous Time Processes

$$S = \sum_{i=1}^{N} x_i \qquad x_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p, \quad \delta_A \approx 0 \end{cases}$$



Binomial Distribution and Continuous Time Processes

$$S = \sum_{i=1}^{N} x_i \qquad x_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p, \quad \delta_A \approx 0 \end{cases}$$



Stochastic Differential Equations

$$\frac{dx(t)}{dt} = f(x(t), \xi(t), t)$$ x is a state and, for $x(t_0) = x_0$, the solution is $x(t)$

$\xi(t)$ is random forcing

For suitable restrictions on f and $\xi(t)$, we can find the solution as

$$x(t) = x_0 + \int_{t_0}^{t} f(x, \xi, \tau) d\tau$$

Important special case is the Langevin equation:

$$\frac{dx(t)}{dt} = a(x(t), t) + b(x(t), t)\xi(t)$$

$$dx(t) = a(x(t), t)dt + b(x(t), t)\underbrace{\xi(t)dt}_{dw}$$

$$x(t) = x_0 + \int_{t_0}^{t} a(x(\tau), \tau)d\tau + \int_{t_0}^{t} b(x(\tau), \tau)dw$$

- $\xi(t)$ is a white noise

  $E\{\xi(t)\xi(t')\} = \delta(t - t')$

  (sometimes 'Gaussian')

- $dw = \xi(t)dt$

  dw is increment of the Wiener process

Langevin Equation

$$dx(t) = a(x(t),t)dt + b(x(t),t)dw$$

Solution: $$x(t) = x_0 + \int_{t_0}^{t} a(x(\tau),\tau)d\tau + \int_{t_0}^{t} b(x(\tau),\tau)\underbrace{dw}_{\xi(t)dt}$$

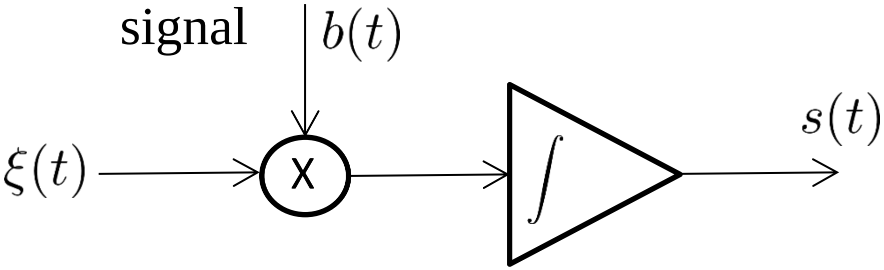
$\int_{t_0}^{t} a(x(\tau),\tau)d\tau$ is the standard Riemann integral

$\int_{t_0}^{t} b(x(\tau),\tau)\underbrace{\xi(t)dt}_{dw}$ is a stochastic integral

- $\xi(t)$ is a white noise $E\{\xi(t)\xi(t')\} = \delta(t-t')$ (sometimes 'Gaussian')
- $dw = \xi(t)dt$
  dw is increment of the Wiener process

Example: Solution of $s(t) = \int_{0}^{t} b(\tau)\xi(\tau)d\tau$

Deterministic signal



Electronic circuit that solves this integral

Astrom, K., "Introduction to Stochastic Control Theory"

Wiener Process

$$dw = \xi(t)dt \Rightarrow w(t) = \underbrace{\int_0^t \xi(\tau)d\tau}_{\text{elementary stochastic integral}}$$

we can also write it as

$$dw = w(t+dt) - w(t) = \xi dt$$

$$dw_k = w(t_k + dt) - w(t_k)$$

$$dt = const$$

$$dx = dw \Rightarrow \begin{aligned} x_1 &= x_0 + dw_1 \\ x_2 &= x_1 + dw_2 = x_0 + dw_1 + dw_2 \\ \ldots &= \ldots \\ x_k &= x_{k-1} + dw_k = x_0 + dw_1 + dw_2 + \ldots + dw_k \end{aligned}$$

Let us assume that on a scale of *dt*, the random increments have the variance σ_{dt}^2 and the mean value 0

$$x_k = x_0 + \sum_{i=1}^{k} dw_i \Rightarrow E\{x_k\} = x_0$$

$$x_k = x_0 + \sum_{i=1}^{k} dw_i \Rightarrow E\{x_k\} = x_0$$

$$E\{(x_k - x_0)^2\} = E\{(\sum_{i=1}^{k} dw_i^2)\} = k\sigma_{dt}^2$$

- $\xi(t)$ is a white noise
  $E\{\xi(t)\xi(t')\} = \delta(t - t')$
  (sometimes ‘Gaussian’)
- $dw = \xi(t)dt$
  dw is increment of the Wiener process

Jazwinski, A.H.,“Stochastic Processes and Filtering Theory”

Wiener Process

$$x_k = x_0 + \sum_{i=1}^{k} dw_i \Rightarrow E\{x_k\} = x_0$$

$$E\{(x_k - x_0)^2\} = E\{(\sum_{i=1}^{k} dw_i)^2\} = k\sigma_{dt}^2 = \frac{t - t_0}{dt}\sigma_{dt}^2$$

The case when $dt = \sigma_{dt}^2$ is called the unit intensity Wiener process.

$$E\{(x_k - x_0)^2\} = t - t_0$$

Finally, note that when $dt \to 0$, then the sum is infinite and due to the central limit theorem, the distribution of x_k is Gaussian.

Summary: $dx = dw, x_0 = w(0) \Rightarrow x(t) = w(t)$

$$p(x_k) = p(x(t)) \Rightarrow \boxed{p(w(t)) = \frac{1}{\sqrt{2\pi(t - t_0)}} e^{-\frac{1}{2}\frac{(w(t) - w(t_0))^2}{(t - t_0)}}} = N(w(t_0), t - t_0)$$

$$p(w(t)|w(t_0)) = N(w(t_0), t - t_0)$$

Wiener Process

$$p(w(t)) = \frac{1}{\sqrt{2\pi(t-t_0)}} e^{-\frac{1}{2}\frac{(w(t)-w(t_0))^2}{(t-t_0)}}$$

This distribution depends on $w(t_0)$, therefore, we can consider it as a conditional probability density function (it is common to assume $w(t_0) = 0$).

$$p(w(t)|w(t_0)) = N(w(t_0), t - t_0)$$

Wiener process sampling: for the initial value $w(t_0)$ and time points $t_0, t_1, t_2, \ldots$

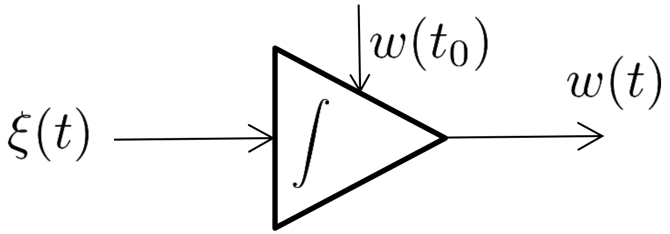
$$w(t_1) = w(t_0) + \Delta w_1, \Delta w_1 \sim N(0, t_1 - t_0)$$

$$w(t_2) = w(t_1) + \Delta w_2, \Delta w_2 \sim N(0, t_2 - t_1)$$

$$w(t_3) = w(t_2) + \Delta w_3, \Delta w_3 \sim N(0, t_3 - t_2)$$

etc….

This is a discrete time realization of the following analog circuit

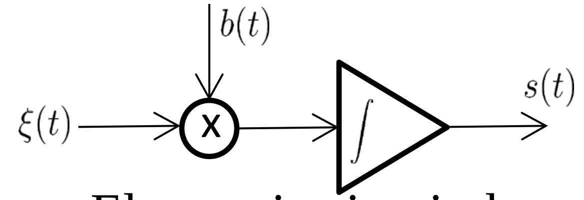


Stochastic Integrals

$$dx(t) = a(x(t),t)dt + b(x(t),t)dw$$

Solution: $$x(t) = x_0 + \int_{t_0}^{t} a(x(\tau),\tau)d\tau + \int_{t_0}^{t} b(x(\tau),\tau)dw$$

$$s(t) = \int_{t_0}^{t} b(x(\tau),\tau)dw$$ is a stochastic integral

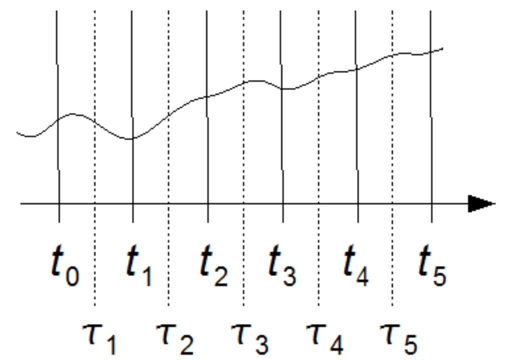


Electronic circuit that solves this integral

$$s(t) \approx s_N(t) = \sum_{i=1}^{N} b(\tau_i)(w(t_i) - w(t_{i-1}))$$

If $\tau_i = t_{i-1}$, then we have the $\hat{I}to$ integral

$$s(t) \approx s_N(t) = \sum_{i=1}^{N} b(t_{i-1})(w(t_i) - w(t_{i-1}))$$



The result is a random value (process).

If we accept to deal with this type of integrals, then there is the associated so-called $\hat{I}to$ differentiation rule.

Øksendal, B., “Stochastic Differential Equations: An Introduction with Applications”

Îto Differentiation Rule

$$dx(t) = a(x(t),t)dt + b(x(t),t)dw$$

$f(x(t))$ is a scalar function. What is df?

Standard calculus: $df = \frac{\partial f}{\partial x}dx = \frac{\partial f}{\partial x}a(x(t),t)dt + \frac{\partial f}{\partial x}b(x(t),t)dw$

$$df = f(x(t+dt)) - f(x) = f(x) + \frac{\partial f}{\partial x}dx + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(dx)^2 + \ldots - f(x(t))$$

$$= \frac{\partial f}{\partial x}(adt + bdw) + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(adt + bdw)^2 + \ldots$$

$$= \frac{\partial f}{\partial x}(adt + bdw) + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}(a^2(dt)^2 + b^2(dw)^2 + 2ab(dt)(dw)) + \ldots$$

Substitute $(dw)^2 = dt$ and ignore all terms that are of order greater than dt

$$df = \left(\frac{\partial f}{\partial x}a(x(t),t) + \frac{1}{2}\frac{\partial^2 f}{\partial x^2}b(x(t),t)^2\right)dt + \frac{\partial f}{\partial x}b(x(t),t)dw$$

Îto Differentiation Rule

Multivariate version

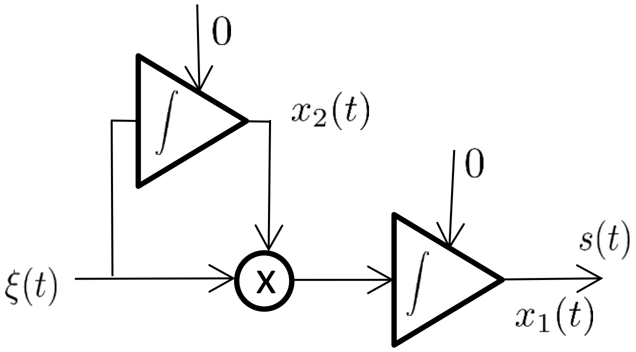
$$dx(t) = a(x(t), t)dt + b(x(t), t)dw \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{bmatrix}$$

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_N \end{bmatrix} \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_N \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & b_{12} & \dots \\ b_{21} & b_{22} & \dots & \\ \dots & & & \\ b_N & \dots & & \end{bmatrix} \qquad dw = \begin{bmatrix} dw_1 \\ dw_2 \\ \dots \\ dw_N \end{bmatrix}$$

$$df(x) = \left\{ \sum_{i=1}^{N} a_i(x,t) \frac{\partial f(x)}{\partial x_i} + \frac{1}{2} \sum_{i,j,k=1}^{N} \frac{\partial^2 f(x)}{\partial x_i \partial x_j} \sigma_{ik} \sigma_{jk} \right\} dt + \sum_{i=1}^{N} \frac{\partial f}{\partial x_i} (bdw)_i$$

Astrom, K., “Introduction to Stochastic Control Theory”

Îto Differentiation Rule Applications



- Find $s(t) = \int_0^t w dw$

In standard calculus, we will have $d(w^2) = 2w dw$

In Îto calculus, we have

$$d(w^2) = 2w dw + \frac{1}{2} 2 (dw)^2 = 2w dw + dt$$

$$d(\frac{1}{2}w^2) = w dw + \frac{1}{2}dt \Rightarrow (\frac{1}{2}w^2) = \int_0^t w dw + \frac{1}{2}t$$

$$s(t) = \int_0^t w dw = \frac{1}{2}w^2 - \frac{1}{2}t$$

Øksendal, B., "Stochastic Differential Equations: An Introduction with Applications"

Îto Differentiation Rule Applications

$$dx = -kxdt + bdw$$

- Find $E\{x\}$ and $E\{(x - E\{x\})^2\} = var\{x\} = \sigma_x^2$ for

$$dE\{x\} = -kE\{x\}dt + E\{bdw\}$$
$$dE\{x\} = -kE\{x\}dt$$

$$d(x^2) = 2xdx + (dx)^2$$
$$d(x^2) = 2x(-kxdt + bdw) + (-kxdt + bdw)^2$$

$$d(x^2) = -2kx^2dt + bdt + bdw$$
$$d(E\{x^2\}) = (-2kE\{x^2\} + b)dt$$

$$d\sigma_x^2 = dE\{x^2\} - d(E\{x\})^2$$

$$d(E\{x\})^2 = 2E\{x\}dE\{x\} = -2k(E\{x\})^2dt$$

$$d\sigma_x^2 = dE\{x^2\} - d(E\{x\})^2 = -2k(E\{x^2\} - E\{x\}^2)dt + bdt$$

$$d\sigma_x^2 = -2k\sigma_x^2dt + bdt \Rightarrow \sigma_x^2(\infty) = \frac{b}{2k}$$

Îto Differentiation Rule Applications

Kalman Filter: Continuous Time Dynamics, Discrete Observation

When applied to linear systems

$$dX = AXdt + Budt + \Gamma dW$$

$$Y = CX + \theta \qquad \theta \sim N(0, \Theta)$$

- Initial guess $\hat{X}_0, \Sigma_{x0}^+$

Prediction

$$dX^- = AX^- dt + Budt, \quad X^+(t_k) \Rightarrow X^-(t_{k+1})$$

$$d\Sigma_x^- = (A\Sigma_x^- + \Sigma_x^- A^T + \Gamma\Gamma^T)dt, \quad \Sigma_x^-(t_k) \Rightarrow \Sigma_x^-(t_{k+1})$$

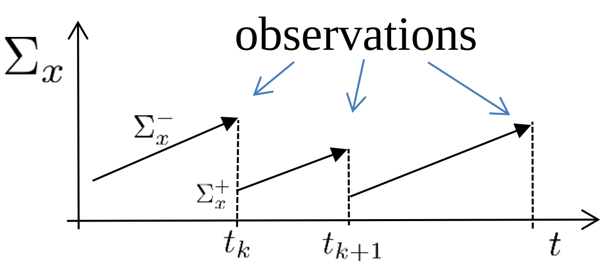
Kalman gain fuses these predictions with the observations

Update

$$X^+(t_{k+1}) = X^-(t_{k+1}) + K_{k+1}(Y(t_{k+1}) - CX^-(t_{k+1}))$$

$$K_{k+1} = \Sigma_x^-(t_{k+1})C^T(C\Sigma_x^- C^T + \Theta)$$

$$\Sigma_x^+(t_{k+1}) = (I - KC)\Sigma_x^-(t_{k+1})$$



Other topics:
- Discrete Dynamic Discrete Observation
- Continuous Dynamics Continuous Observations
- Kalman Smoother
- Nonlinear: Extended and 2nd order Kalman Filter

Gelb, A., "Applied Optimal Estimation"

Kalman Filter Project I

Input Data:

- Digital camera movie of a robot
- Resolution
- Approximate robot dimensions

Pre-processing:

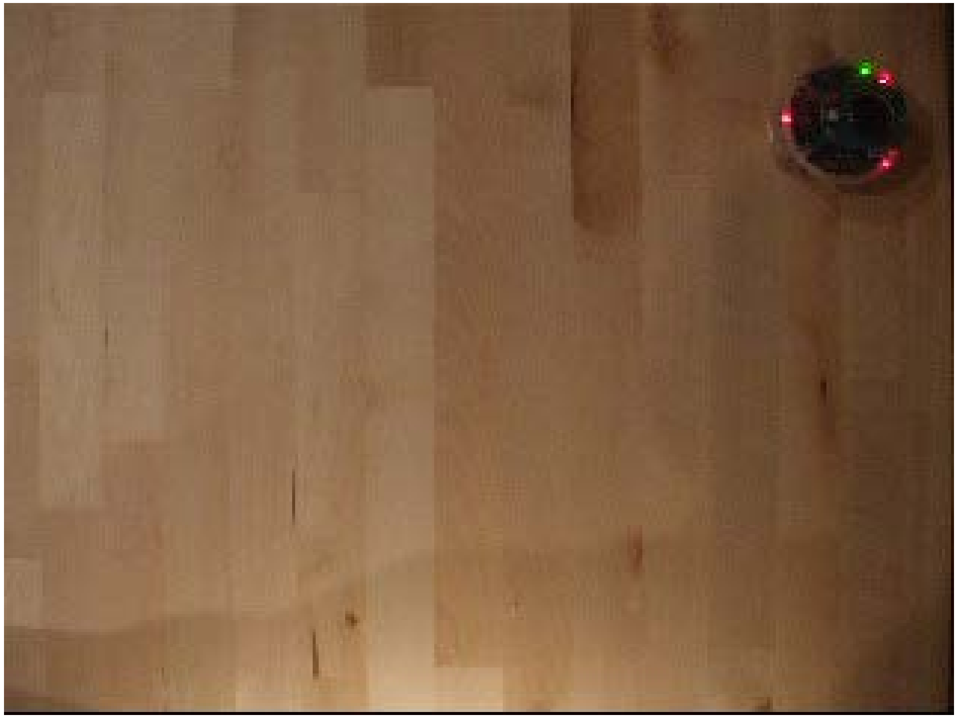
- Find the heading angle of the robot based on three red lights
- Find the center of the robot

Problem:

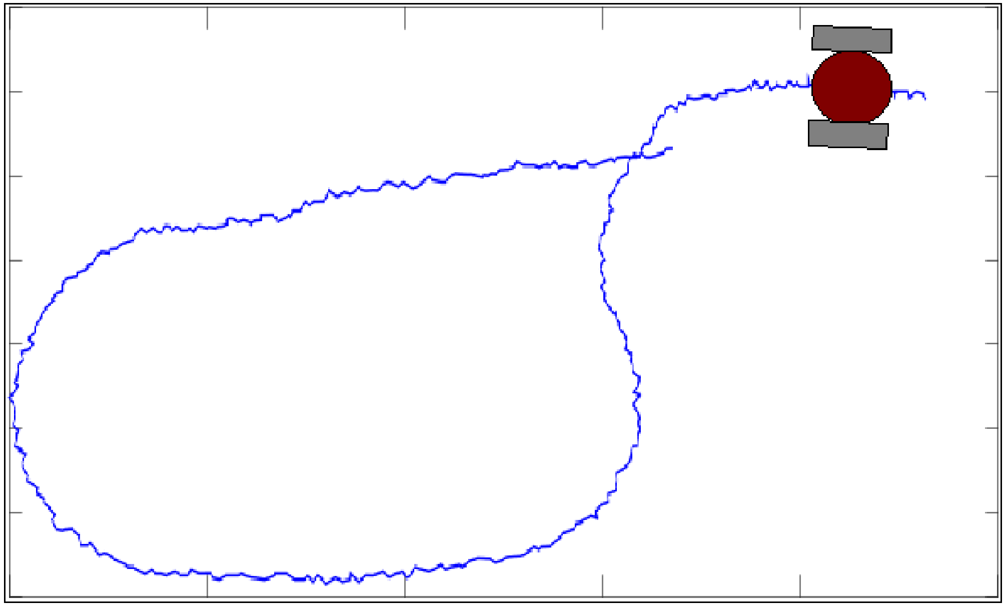
- Use the robot center measurements to find velocity and robot heading angle

Verification:

- Compare the KF estimated robot heading angle with the one based on three red lights (image based)



Kalman Filter Project I



$$dx(t) = v\cos(\theta)dt$$

$$dy(t) = v\sin(\theta)dt$$

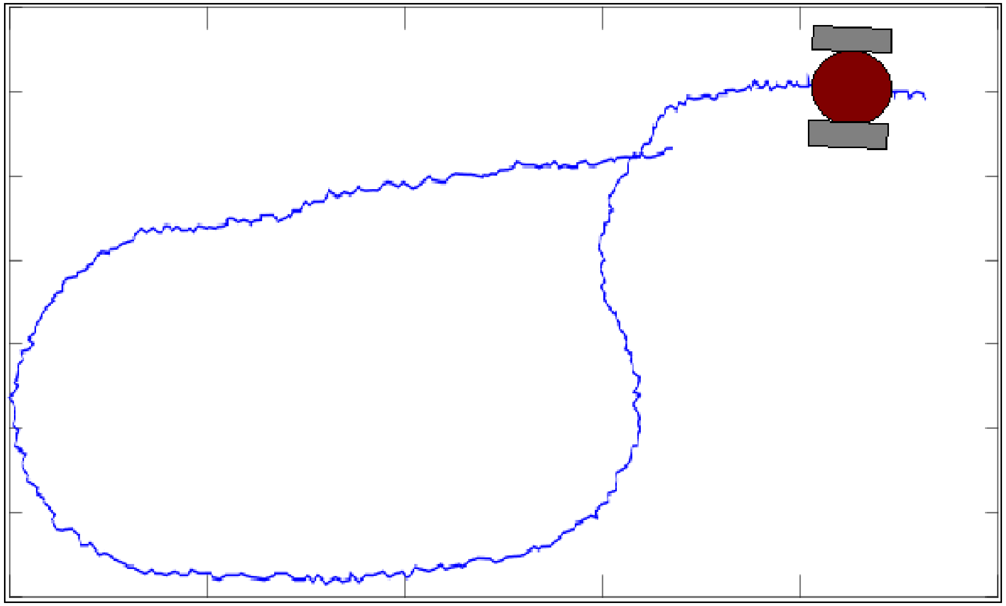
$$dv(t) = udt$$

$$d\theta(t) = \omega dt$$

Given control, the trajectory is defined.

If the trajectory is known, what are $v(t)$ and $\theta(t)$?

Kalman Filter Project I



$$dx(t) = v\cos(\theta)dt$$

$$dy(t) = v\sin(\theta)dt$$

$$dv(t) = dw_v$$

$$d\theta(t) = dw_\theta$$

Robot center observation model

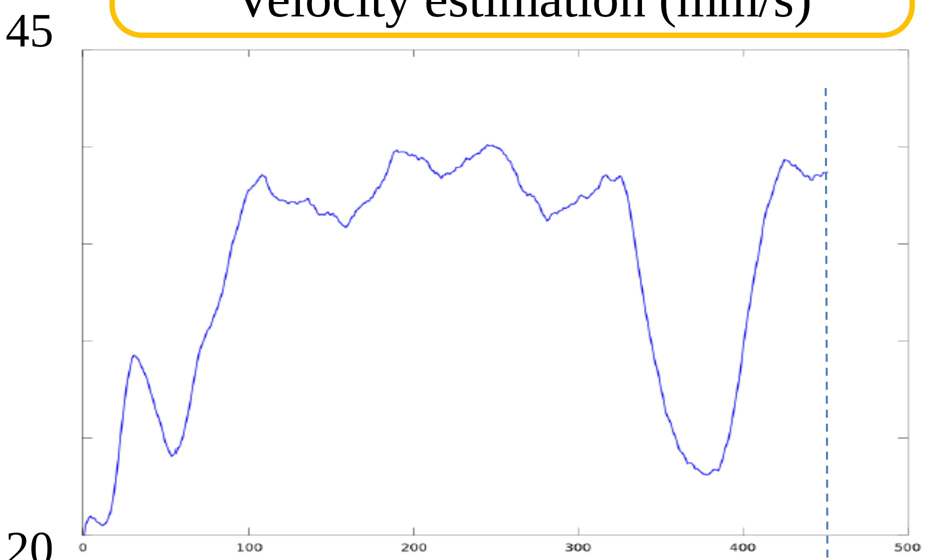
$$x_m(t) = x(t) + n_x(t)$$

$$y_m(t) = y(t) + n_y(t)$$

Unknown control variables are modeled by stochastic processes.

Kalman Filter Project I

Velocity estimation (mm/s)



$$dx(t) = v\cos(\theta)dt$$

$$dy(t) = v\sin(\theta)dt$$

$$dv(t) = dw_v$$

$$d\theta(t) = dw_\theta$$

Robot center observation model

$$x_m(t) = x(t) + n_x(t)$$

$$y_m(t) = y(t) + n_y(t)$$

Heading angle estimation (rad)

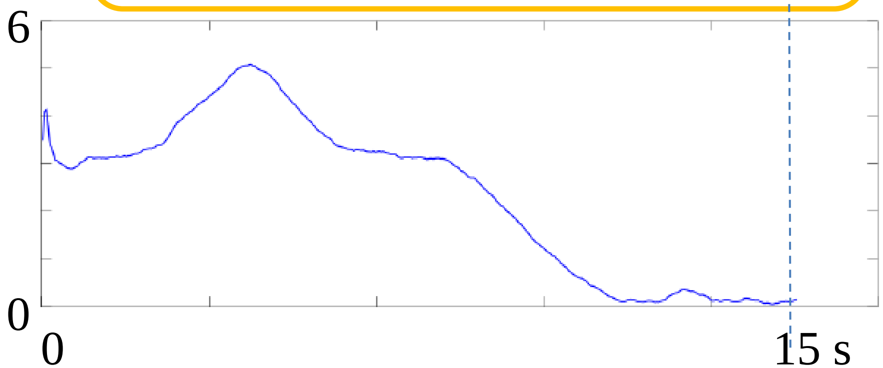
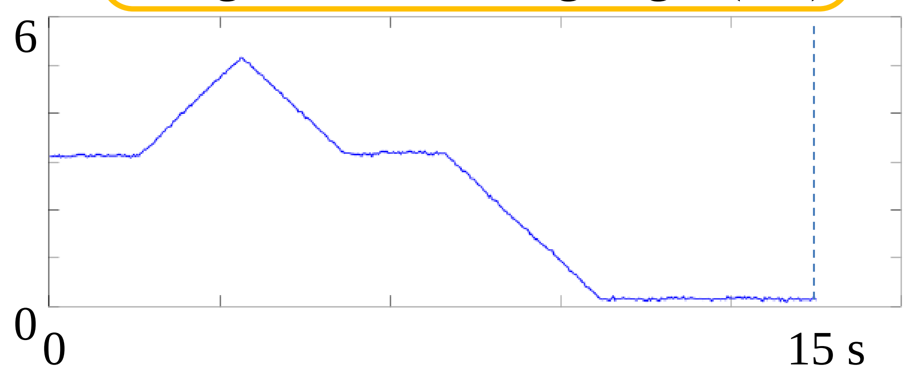
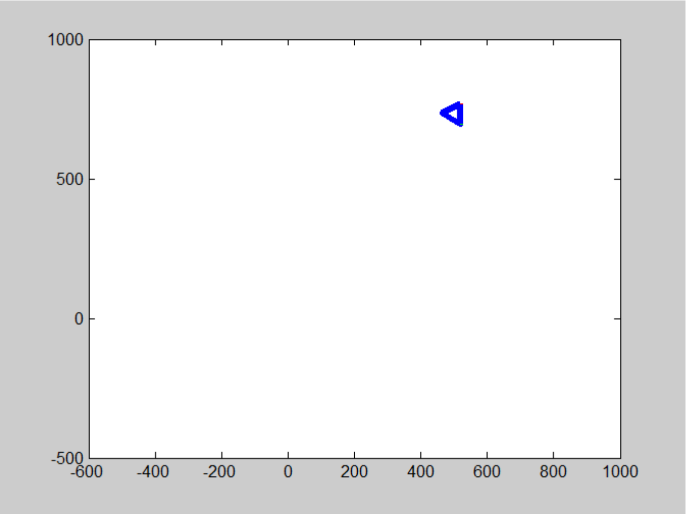


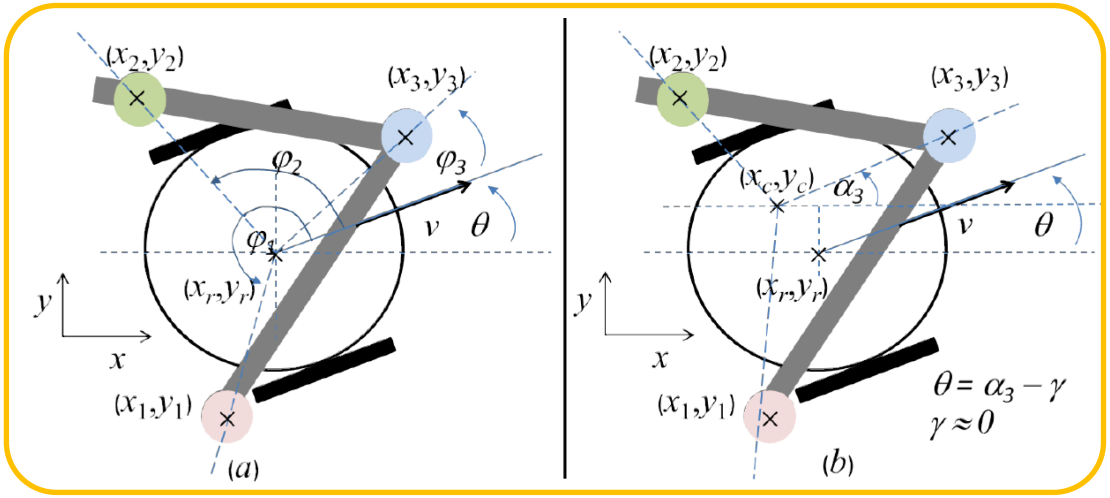
Image based heading angle (rad)



Kalman Filter Project II

Estimation of the relative position of the triangular configuration of markers with respect to the robot center and its heading angle





Robot motion model

$$dx(t) = v\cos(\theta)dt$$

$$dy(t) = v\sin(\theta)dt$$

$$dv(t) = dw_v$$

$$d\theta(t) = dw_\theta$$

Observation model for markers $n_i \sim N(0, 25)$

$$x_1(k) = x_r(k) + r_1^c\cos(\theta(k)) - r_1^s\sin(\theta(k)) + n_1(k)$$

$$y_1(k) = y_r(k) + r_1^c\sin(\theta(k)) + r_1^s\cos(\theta(k)) + n_2(k)$$

$$x_2(k) = x_r(k) + r_2^c\cos(\theta(k)) - r_2^s\sin(\theta(k)) + n_3(k)$$

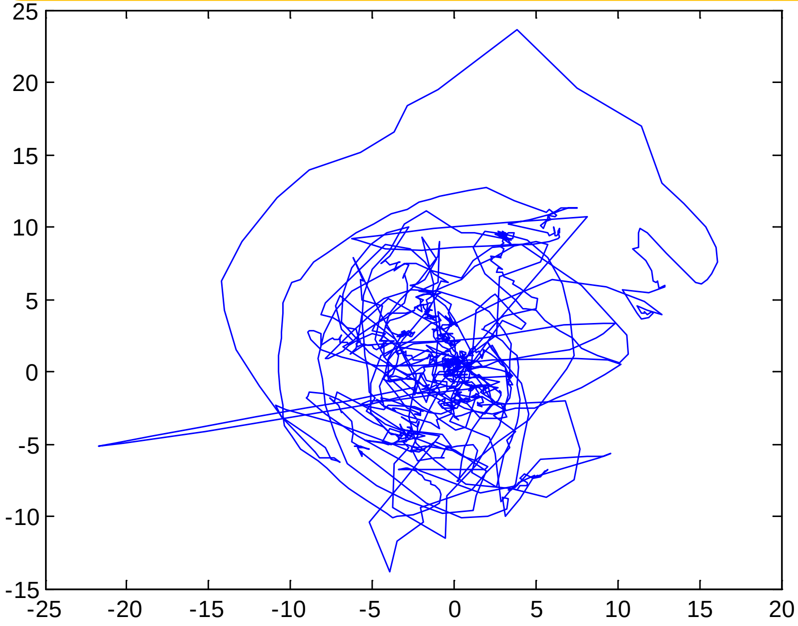
$$y_2(k) = y_r(k) + r_2^c\sin(\theta(k)) + r_2^s\cos(\theta(k)) + n_4(k)$$

$$x_3(k) = x_r(k) + r_3^c\cos(\theta(k)) - r_3^s\sin(\theta(k)) + n_5(k)$$

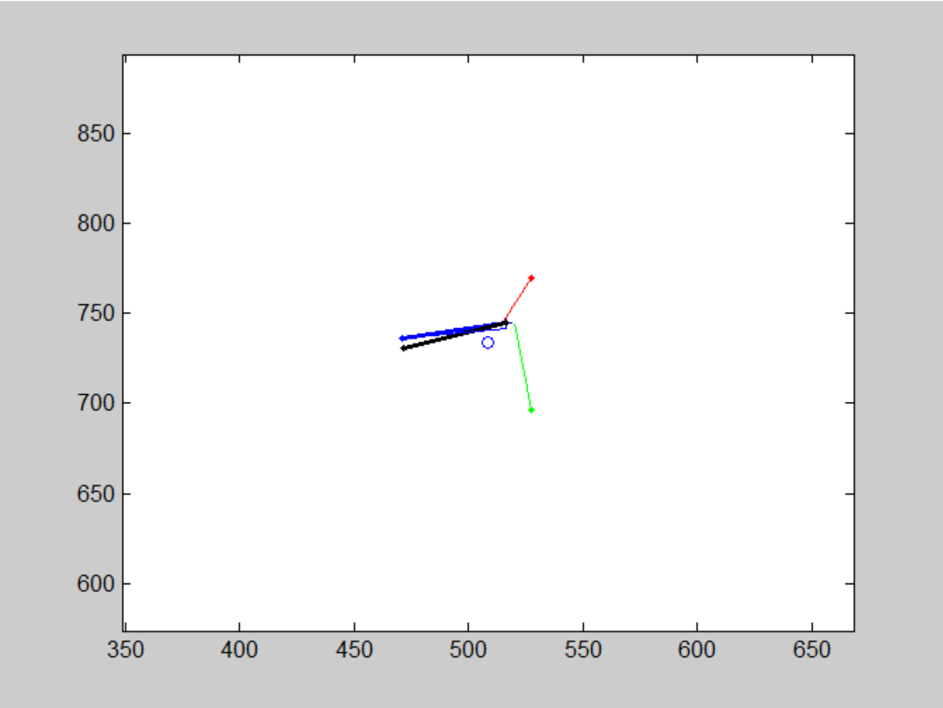
$$y_3(k) = y_r(k) + r_3^c\sin(\theta(k)) + r_3^s\cos(\theta(k)) + n_6(k)$$

Kalman Filter Project II

x-y displ. of center estimations (mm)



Samples



Euler-Murayama Method

Simple algorithm that generates a sample of SDE:

$$dx(t) = a(x(t), t)dt + b(x(t), t)dw$$

The sample points are equidistant in time (Δt)

$$x(t_{k+1}) = x(t_k) + a(x(t_k), t_k)\Delta t + b(x(t_k), t_k)\Delta W \qquad \Delta W \sim N(0, \Delta t)$$

A critical component of the method is the random generator

For more sophisticated methods, see:

Kloeden, P.E., Platen, E., Numerical Solution of Stochastic Differential Equations, Springer 1992.

Fokker-Planck Equation

Describes the evolution of the state probability density function

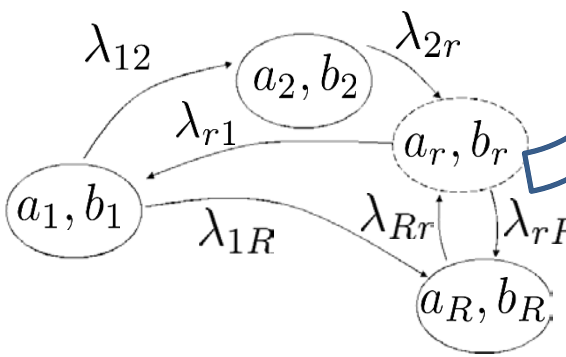
SDE: $dx(t) = a(x(t),t)dt + b(x(t),t)dw$

$$\frac{\partial \rho}{\partial t} = \sum_{i=1}^{N} \frac{\partial(-a_i \rho)}{\partial x_i} + \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{1}{2} \frac{\partial^2([bb^T]_{ij}\rho)}{\partial x_i \partial x_j} = F\rho$$

Switching diffusions : $dx(t) = a_{r(t)}(t)(x(t),t)dt + b_{r(t)}(t)(x(t),t)dw$

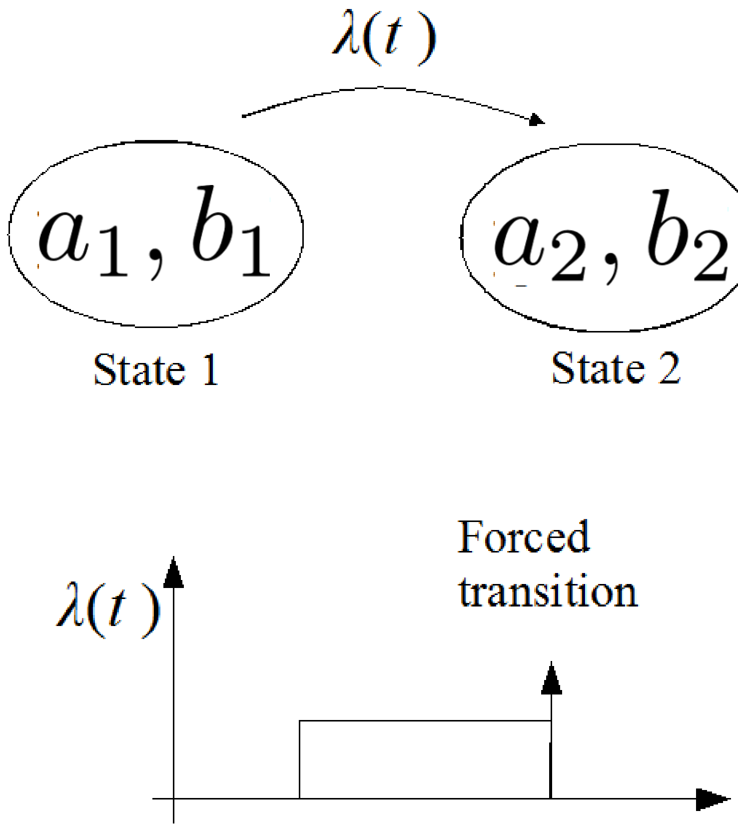
Probability density function is: $\rho = [\rho_1 \ \rho_2 \ \rho_3 \ \dots \ \rho_R]^T$
a vector of functions

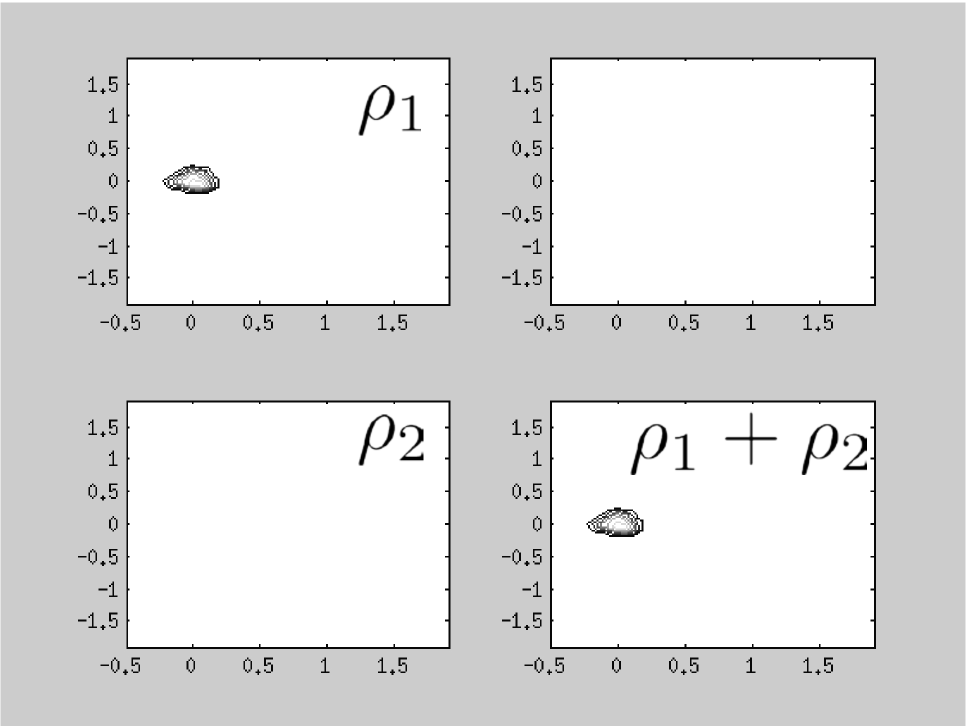
$$\left.\begin{aligned}
\frac{\partial \rho_1}{\partial t} &= \sum_{j=1}^{R} \lambda_{j1}\rho_j - \frac{\partial}{\partial x}(a_r \rho_1) + \frac{1}{2}\frac{\partial^2}{\partial x^2}(b_1 \rho_1) \\
\frac{\partial \rho_r}{\partial t} &= \sum_{j=1}^{R} \lambda_{jr}\rho_j - \frac{\partial}{\partial x}(a_r \rho_r) + \frac{1}{2}\frac{\partial^2}{\partial x^2}(b_r \rho_r) \\
\frac{\partial \rho_R}{\partial t} &= \dots
\end{aligned}\right\} = F\rho$$



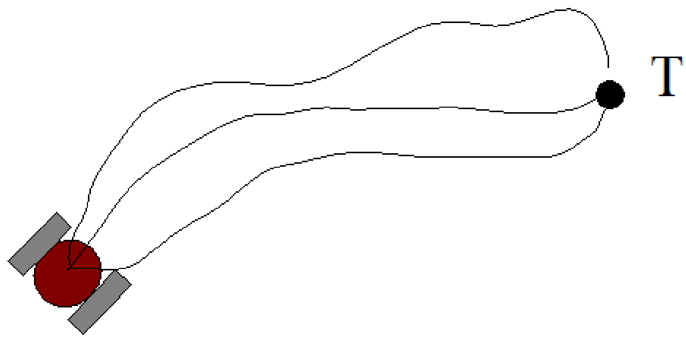
Yin, G.G., Zhu, C.: “Hybrid Switching Diffusions”, Springer, 2010

Fokker-Planck Equation





What is Stochastic in Robotics ?

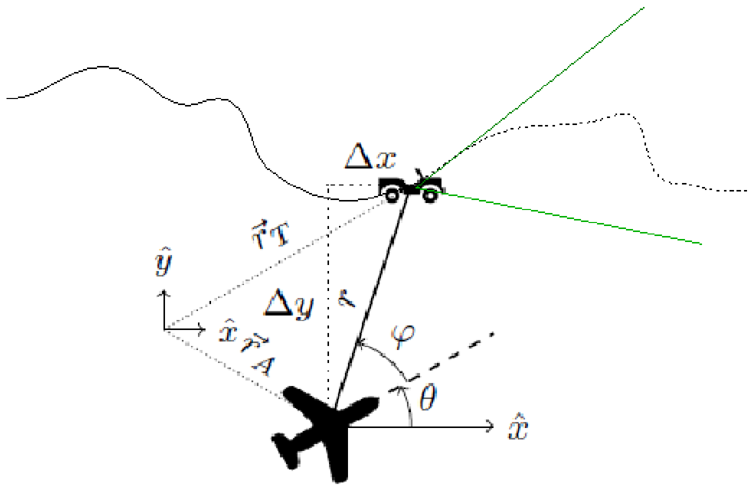


- Reaching the target is possible in many ways
- Single model for the family of all possible paths is a stochastic process
- Any particular trajectory can be considered as a realization of a stochastic process

Stochasticity models available options

Options are either a part of decision making, or chosen by nature (disturbances)

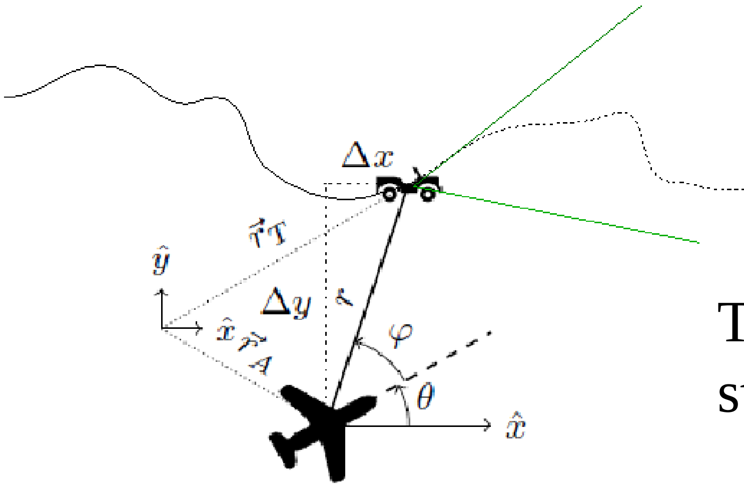
Dubins Vehicle Following a Target



- Following the target at a fixed distance
- The future of the target trajectory is unknown (uncertain)
- We model it as a stochastic process
- This process serves as a prior for the target trajectory future

Anderson R. and Milutinović D., “ Dubins Vehicle Tracking of a Target With Unpredictable Trajectory”, Proceedings of the 2011 ASME Dynamic Systems and Control Conference (DSCC), Arlington, VA

Dubins Vehicle Following a Target



Vehicle model (VM):

$$dx(t) = v\cos(\theta)dt$$

$$dy(t) = v\sin(\theta)dt$$

$$d\theta(t) = -u dt$$

Target kinematics is unknown (therefore stochastic prior) (TM):

$$dx_T(t) = \sigma dw_x$$

$$dy_T(t) = \sigma dw_y$$

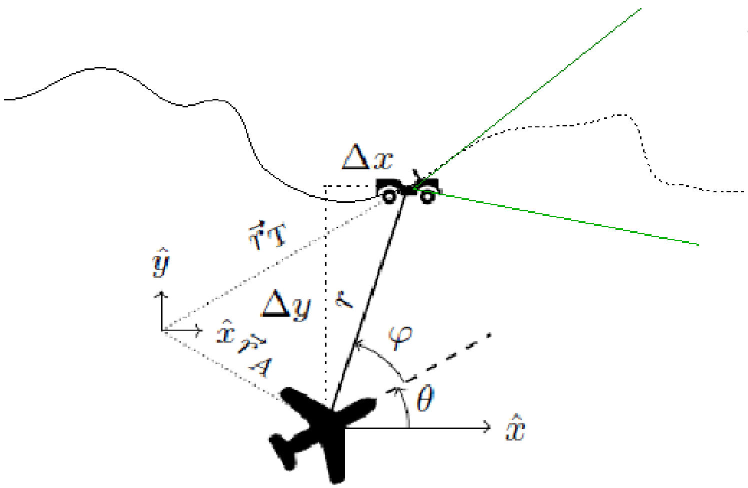
To follow the target at a constant distance (*d*), we formulate the optimal control problem of minimizing the cost function

$$W(u) = E\int_0^\infty e^{-\beta t}(r-d)^2 dt, \quad r = \sqrt{(\Delta x)^2 + (\Delta y)^2}$$

under constraints of (VM) and (TM)

Note: We use the type of cost function for which a feedback solution exists.

Dubins Vehicle Following a Target



Îto calculus:

$$r = \sqrt{(\Delta x)^2 + (\Delta y)^2}, \quad \varphi = \arctan \frac{\Delta y}{\Delta x}$$

Relative kinematics:

$$dr = -(v\cos(\varphi) + \frac{\sigma^2}{2r})dt + \sigma_{w_0} dw_0$$

$$d\varphi = (\frac{v}{r}\sin(\varphi) - u)dt + \frac{\sigma}{r} dw_n$$

Cost function:

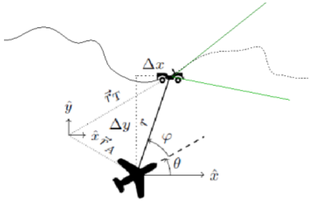
$$W(u) = E\int_0^\infty e^{-\beta t}(r-d)^2 dt$$

- The cost function allows for the feedback solution
- There is no prediction, or any sort of estimation
- The control anticipates the uncertainty of target motion

Now it is all about computing the solution.

Anderson R. and Milutinović D., “ Dubins Vehicle Tracking of a Target With Unpredictable Trajectory”, Proceedings of the 2011 ASME Dynamic Systems and Control Conference (DSCC), Arlington, VA

Dubins Vehicle Following a Target



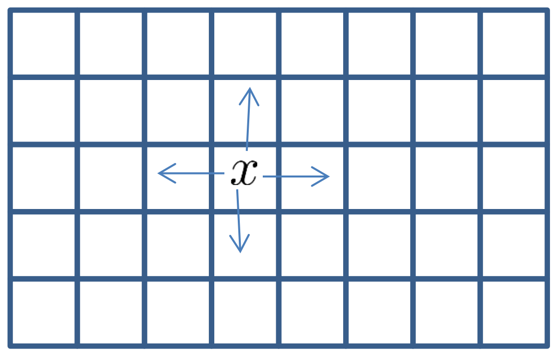
$$dr = -(v\cos(\varphi) + \frac{\sigma^2}{2r})dt + \sigma_{w_0} dw_0$$

$$d\varphi = (\frac{v}{r}\sin(\varphi) - u)dt + \frac{\sigma}{r} dw_n$$

$$W(u) = E\int_0^\infty e^{-\beta t}(r-d)^2 dt$$

Dynamic programming – Value iterations

$$W(x) = E_x \sum_{n=0}^{\infty} e^{-\beta n} c(\xi_n)$$



Update can be done in any order

$$V(x) = min_u \left\{ c(x,u) + e^{-\beta} \sum_y p(y|x,u) W(y) \right\}$$

$$\pi(x) = argmin_u \left\{ c(x,u) + e^{-\beta} \sum_y p(y|x,u) W(y) \right\}$$

For transition rates, we use a locally consistent Markov Chain approximation.

Dubins Vehicle Following a Target

$$W((r,\varphi);u^*) = E\int_0^\infty e^{-\beta t}(r-d)^2 dt$$

$$0 = inf_u\left\{L^u V - \beta(x)V(x) + k(x(t))\right\}$$

$$L^u = \sum_{i=1}^{2} a_i \frac{\partial}{\partial x} + \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2} b_{ij}\frac{\partial^2}{\partial x_i \partial x_j}$$

$$x = [x_1 \; x_2]^T, \; x_1 = r, \; x_2 = \varphi$$

$$d\varphi = (\frac{v}{r}\sin(\varphi) - u)dt + \frac{\sigma}{r}dw_n$$

$$dr = -(v\cos(\varphi) + \frac{\sigma^2}{2r})dt + \sigma_{w_0}dw_0$$

$$a_1 = -(v\cos(\varphi) + \frac{\sigma^2}{2r}) \qquad b_{11} = \sigma_{w_0}^2$$

$$a_2 = (\frac{v}{r}\sin(\varphi) - u) \qquad b_{22} = \sigma^2/r^2$$

$$k = (r-d)^2 \qquad b_{12} = b_{21} = 0$$

- Locally consistent Markov chain approximation

$$0 = L^u W(x) - \beta(x)W(x) + k(x(t))$$

$$W(x) = \Delta t^h k(x,u) + \sum_y p(y|x,u)W(y)$$

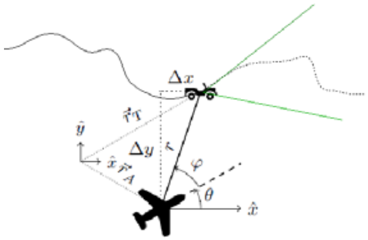
Locally consistent approximation provides the relation between the discretization steps in the state space $\Delta r, \Delta\varphi$ and the time step Δt^h

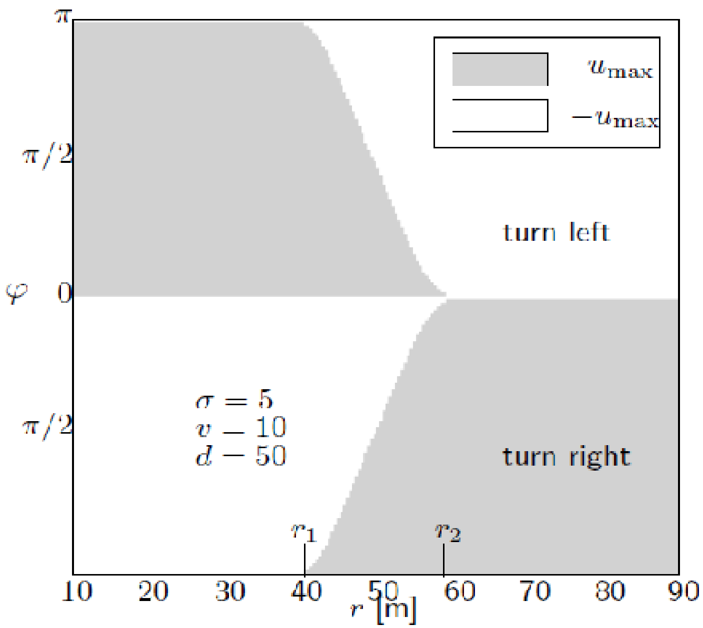
- Value iterations

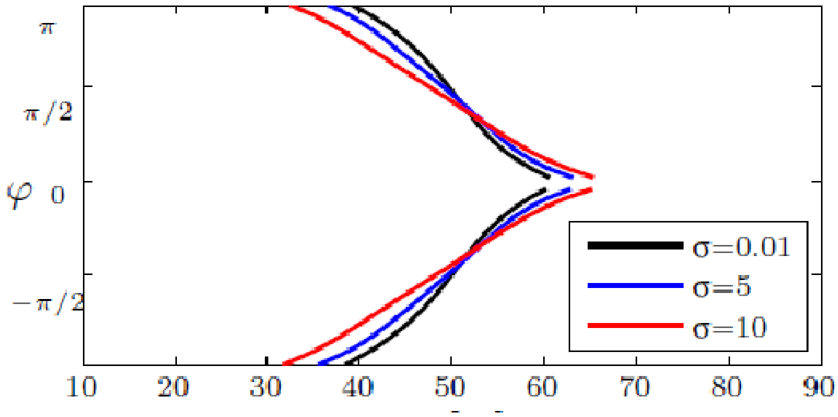
$$V(x) = min_u\left\{\Delta t^h k(x,u) + \sum_y p(y|x,u)W(y)\right\}$$

Kushner, H.J., Dupuis, P.: “Numerical Methods for Stochastic Control Problems in Continuous Time”, 2001

Dubins Vehicle Following a Target

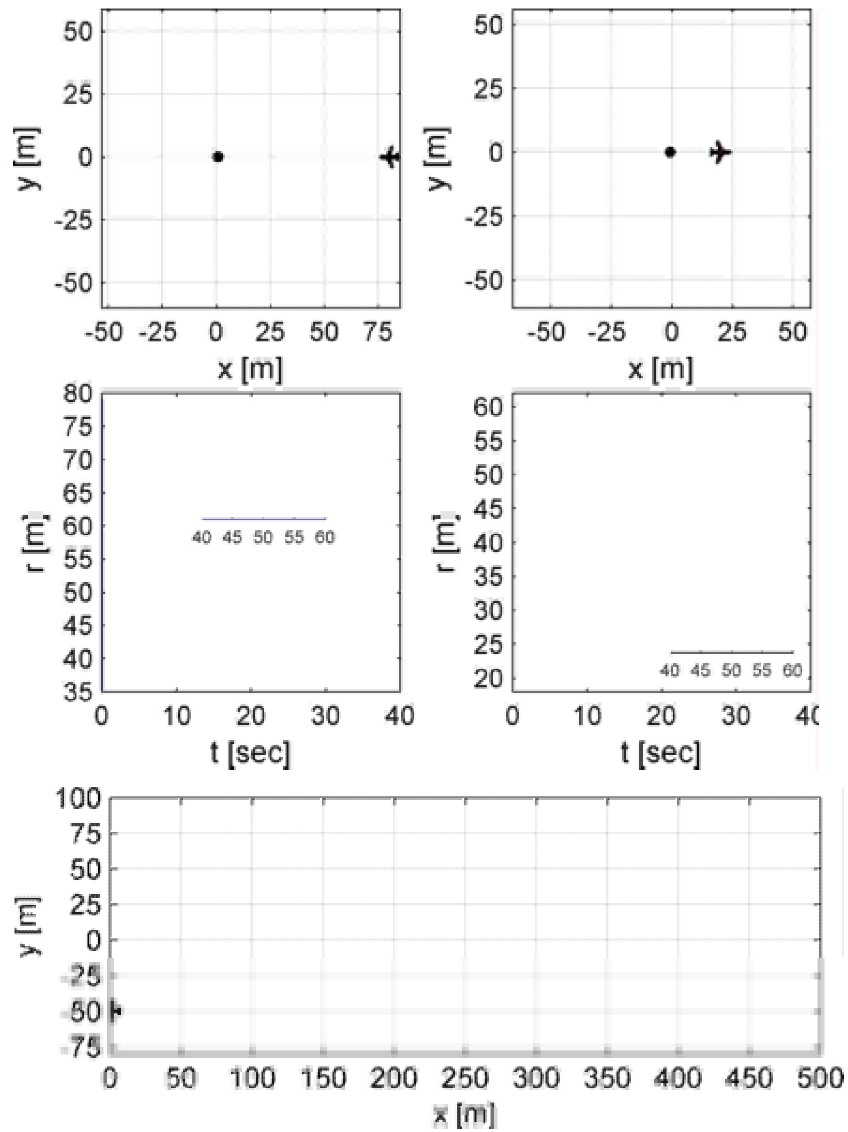


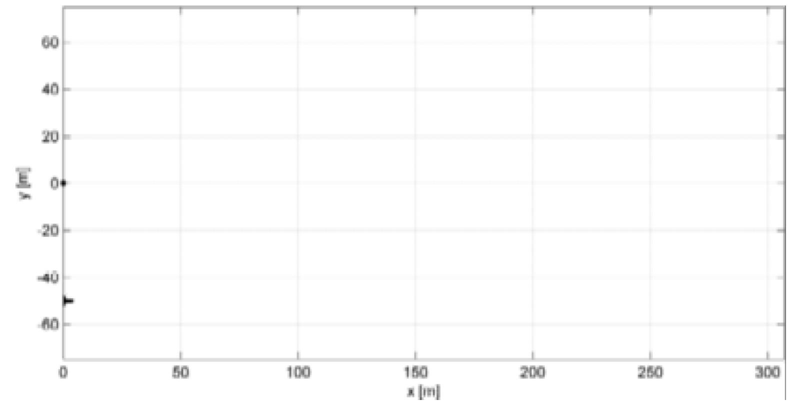


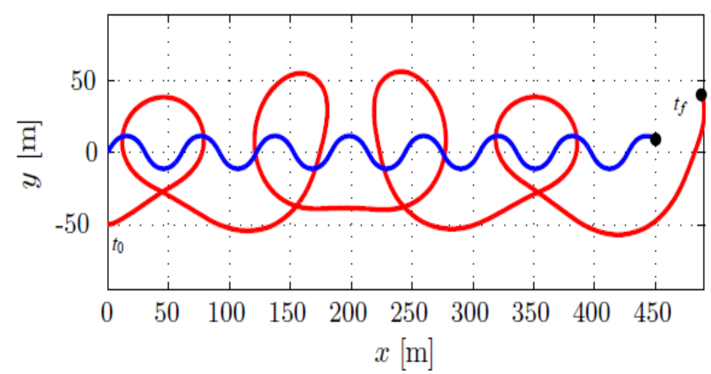


With higher noise intensities, the UAV begins entry into circular pattern earlier

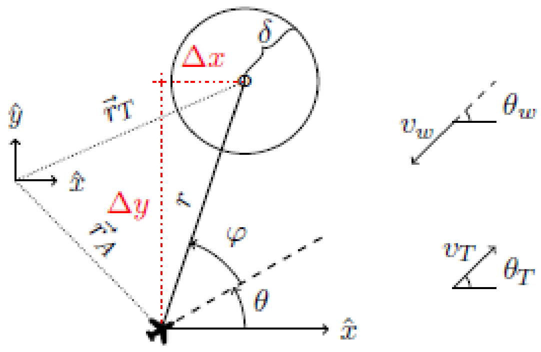
Dubins Vehicle Following a Target







Dubins Vehicle and Stochastic Wind



$$d\Delta x(t) = v\cos(\theta)dt + v_w\cos(\theta_w)dt$$

$$d\Delta y(t) = v\sin(\theta)dt + v_w\sin(\theta_w)dt$$

$$d\theta(t) = u(t)dt$$

$$d\theta_w(t) = \sigma_\theta(t)dw_\theta$$

Minimize: $J(u) = E\left\{\int_0^\tau dt\right\}$

τ is the time until the target is reached

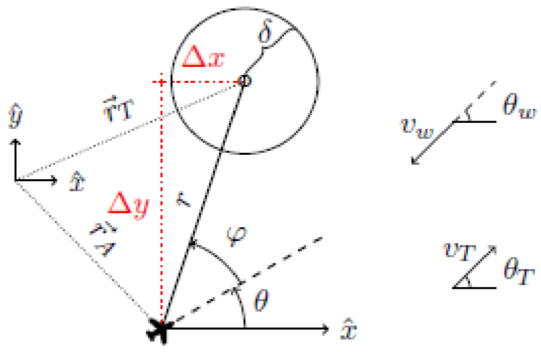
$$dr(t) = -(v\cos(\varphi+\gamma) + v_w\cos(\varphi))dt$$

$$d\varphi(t) = \left(\frac{v}{r}\sin(\varphi+\gamma) + \frac{v_w}{r}\sin(\varphi) - u\right)dt$$

$$d\gamma = udt - \sigma_\theta dw_\theta$$

Anderson, R., Efstathios, B., Milutinović D., Panagiotis, T., Optimal Feedback Guidance of a Small Aerial Vehicle in the Presence of Stochastic Wind, AIAA Journal of Guidance, Control and Dynamics, Vol. 36, No. 4, pp. 975-985, 2012

Dubins Vehicle and Stochastic Wind



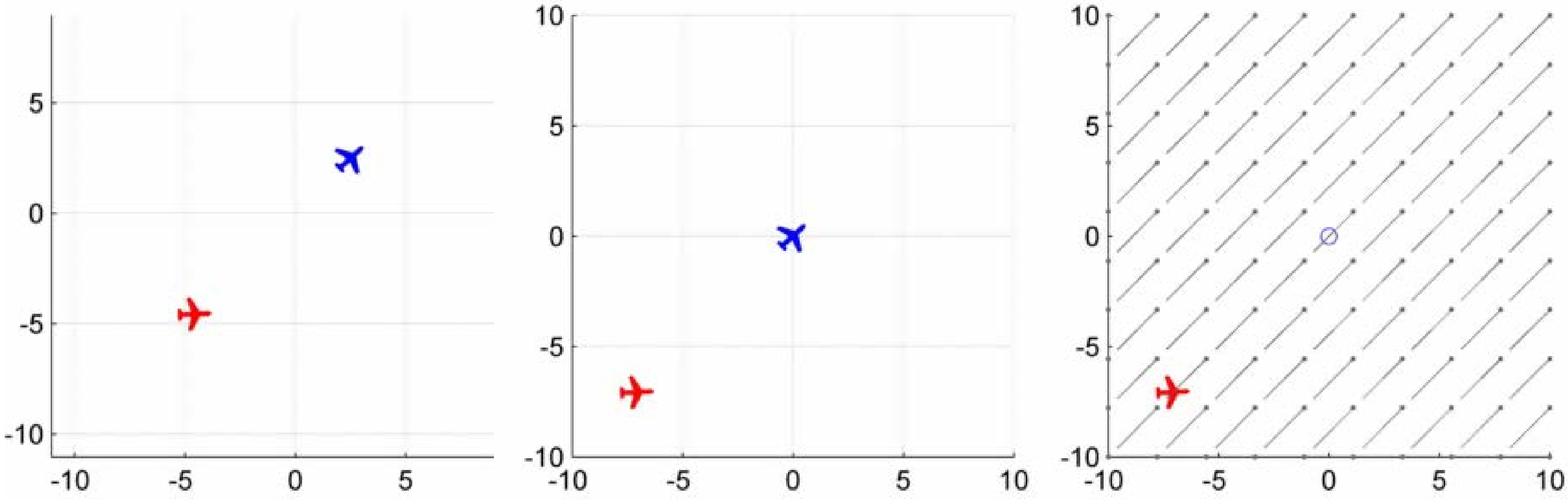
$$dr(t) = -(v\cos(\varphi+\gamma) + v_w\cos(\varphi))dt$$

$$d\varphi(t) = \left(\frac{v}{r}\sin(\varphi+\gamma) + \frac{v_w}{r}\sin(\varphi) - u\right)dt$$

$$d\gamma = udt - \sigma_\theta dw_\theta$$

Minimize: $J(u) = E\left\{\int_0^\tau dt\right\}$

τ is the time until the target is reached

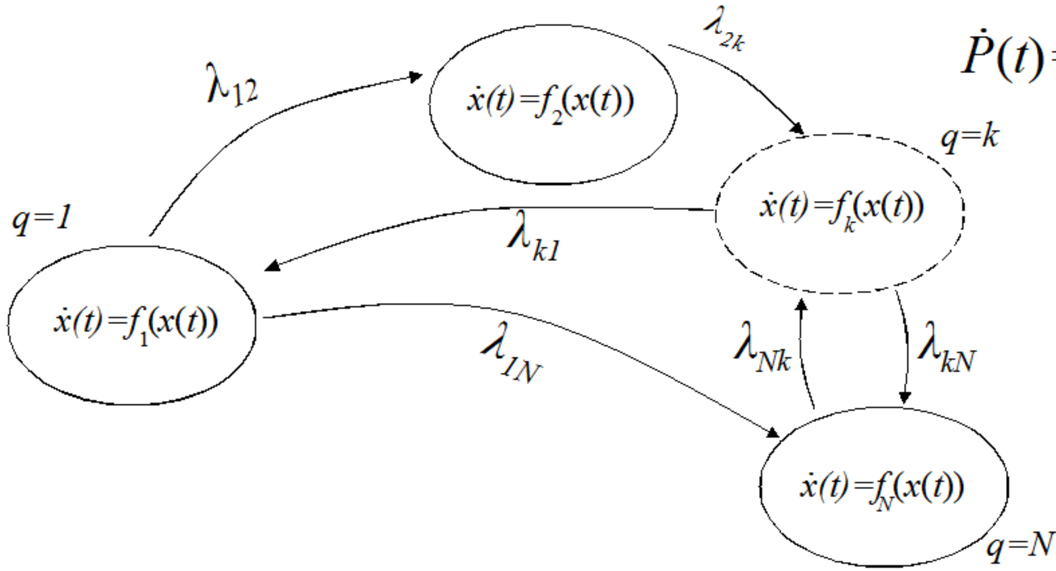


Open-loop Stochastic Optimal Control Problems

- Solutions of continuous optimal control problems (deterministic/stochastic) are in close relation with partial differential equations (PDEs)
    - Hamilton-Jacoby-Belman (HJB) PDE
    - Stochastic Hamilton-Jacoby-Belman PDE
- Hamiltonian formulation for stochastic control problems is necessary to solve open-loop stochastic control problems (minimum principle).
  There are several attempts for stochastic differential equations. See:
  Stochastic Controls: Hamiltonian Systems and HJB Equations by J. Yong and X. Y. Zhou,
- Our approach is to control state probability density function evolutions that are defined based on PDEs, or PDE systems.
    - Pontryagin-like minimum principle (PMP) for infinite dimensional systems

It can be applied to Stochastic Differential Equations and Stochastic Hybrid Systems

Hybrid State Probability Dynamics



$$\dot{P}(t)=L^T P(t) \text{ "+" } \dot{x}(t)=f(x(t))$$

Hybrid State: (x,q)

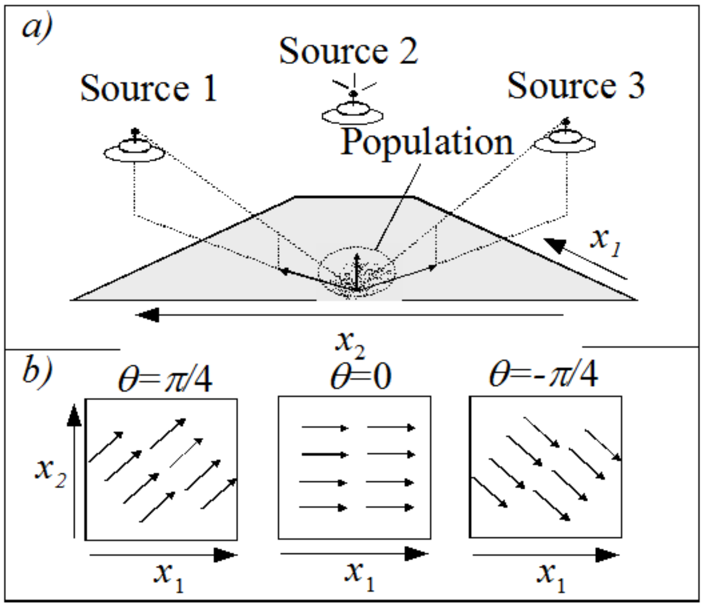
$$\rho(x,t)=\begin{bmatrix}\rho_1(x,t)\\ \rho_2(x,t)\\ \vdots\\ \rho_N(x,t)\end{bmatrix}$$

$$\frac{\partial\rho(x,t)}{\partial t}=L^T\rho(x,t)-\begin{bmatrix}\nabla\cdot(f_1(x,t)\rho_1(x,t))\\ \nabla\cdot(f_2(x,t)\rho_2(x,t))\\ \vdots\\ \nabla\cdot(f_N(x,t)\rho_N(x,t))\end{bmatrix}$$

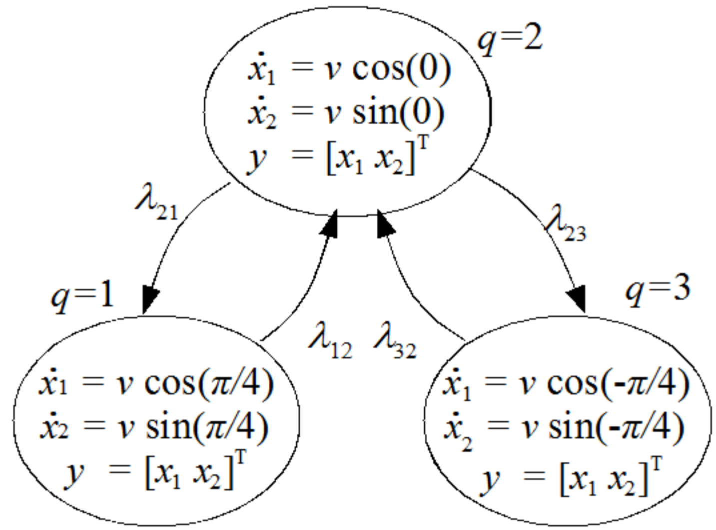
$$\nabla\cdot(f_k\,\rho_k)=\begin{bmatrix}\frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} & \cdots & \frac{\partial}{\partial x_N}\end{bmatrix}\begin{bmatrix}f_k^1\rho_k\\ f_k^2\rho_k\\ \vdots\\ f_k^N\rho_k\end{bmatrix}$$

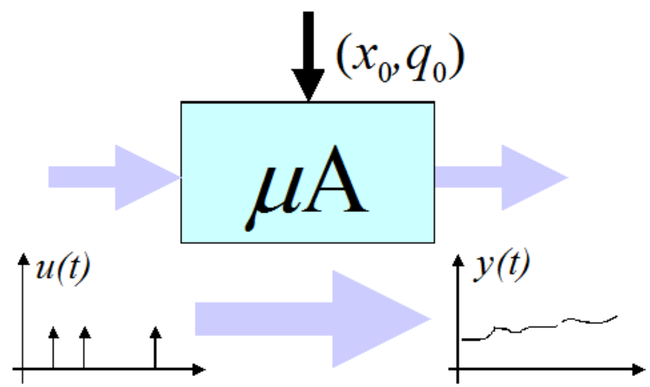
D. Milutinovic, P. Lima, “Cells and Robots”, Springer, 2007

Robotic Population Mission Scenario



Copyright IEEE, 2003

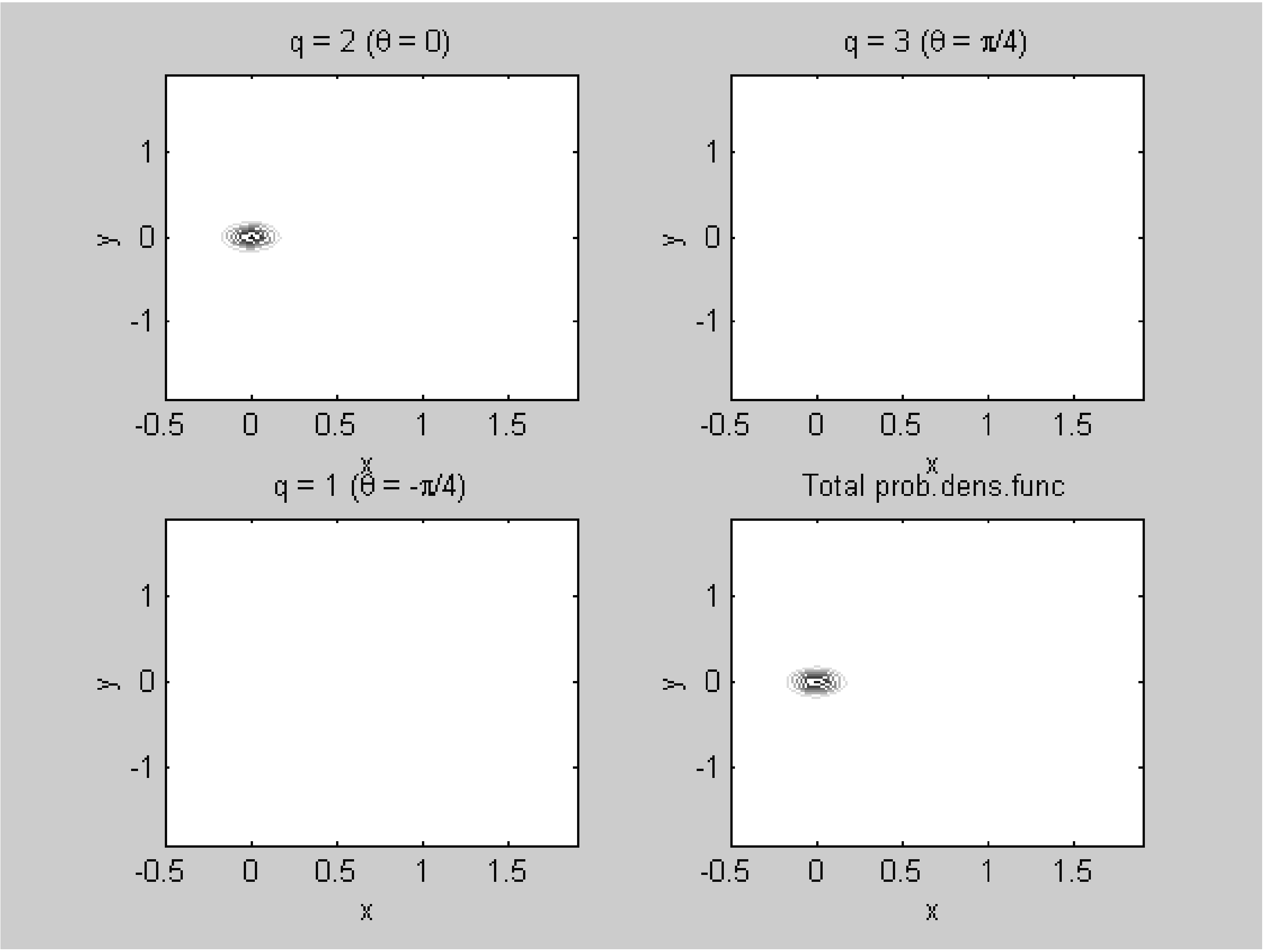




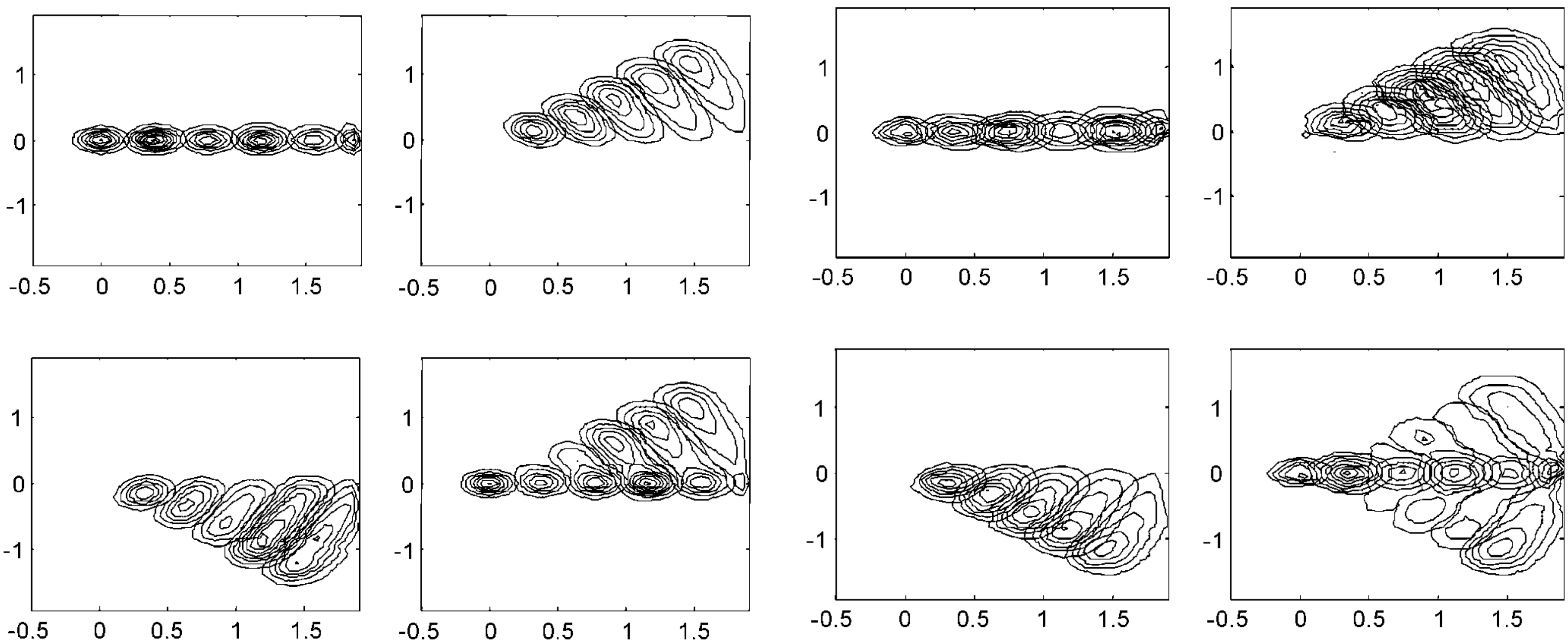
D, Milutinovic et. al (2003), ICRA

D. Milutinovic, P. Lima (2007), “Cells and Robots”

Robot Distribution



Optimal Control Problem

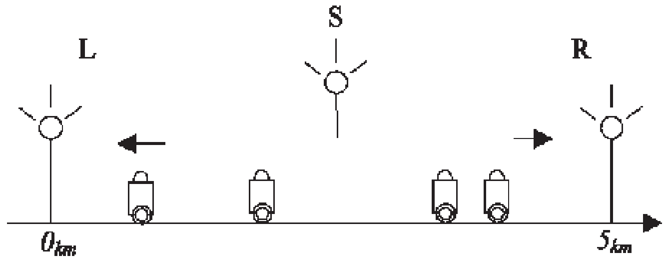


Case I: λ_{12}= 0.5, λ_{21}= 0.1, λ_{23}= 0.9, λ_{32}= 0.1 at time instants t = 0, 0.39, 0.79,1.18,1.57, 1.96.

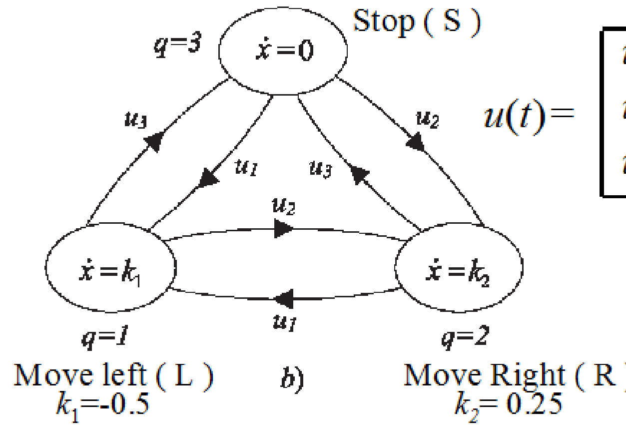
Case II: λ_{12}=0.1, λ_{21}=0.5, λ_{23}=0.5, λ_{32}=0.4 at time instants t= 0, 0.39, 0.79,1.18,1.57, 1.96.

Optimal Control 1D Example

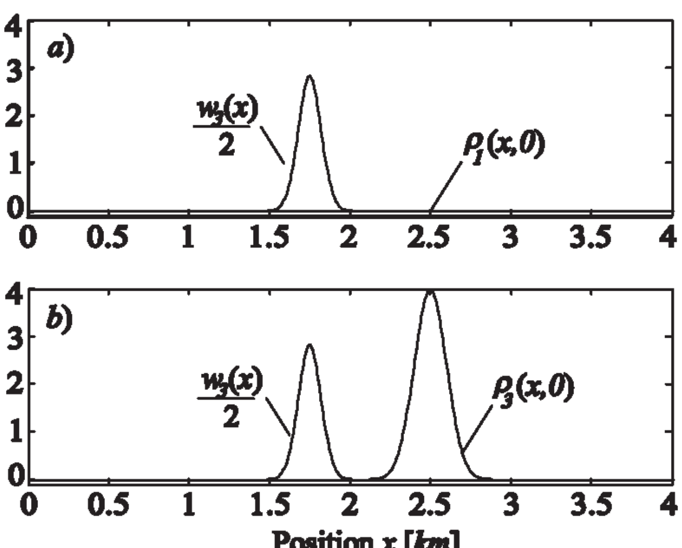
$$u^*(t) = \max_{u \in U_{ad}} \int_X w(x)^T \rho(x,T)\,dx \qquad T=3h$$



a)



$$u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix}$$



$$\rho(x,t) = \begin{bmatrix} \rho_1(x,t) \\ \rho_2(x,t) \\ \rho_3(x,t) \end{bmatrix} \quad \rho(x,0) = \begin{bmatrix} \rho_1(x,0) \\ \rho_2(x,0) \\ \rho_3(x,0) \end{bmatrix} \quad w(x) = \begin{bmatrix} 0 \\ 0 \\ w_3(x) \end{bmatrix}$$

D, Milutinovic, P. Lima (2006), IEEE Trans.Robotics

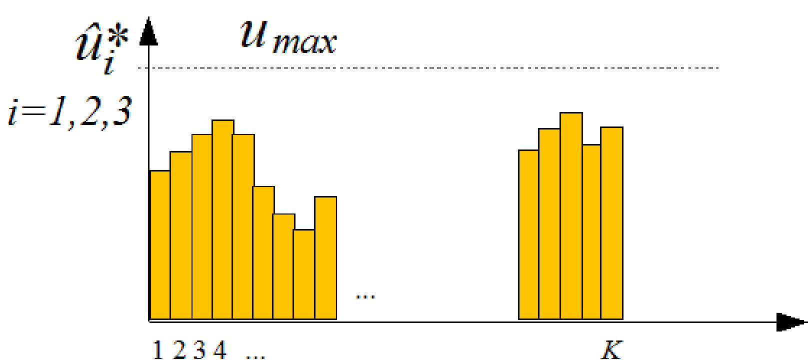
D. Milutinovic, P. Lima (2007), “Cells and Robots”

Optimal Control

$$u^* = \max_{u\in U_{ad}} \int_X w(x)^T \rho(x,T)dx \quad \Leftrightarrow \quad u^* = \min_{u\in U_{ad}} J(u) = \min_{u\in U_{ad}} \int_X -w(x)^T \rho(x,T)dx$$

$$\frac{\partial \rho(x,t)}{\partial t} = L^T(u)\rho(x,t) - \begin{bmatrix} \nabla\cdot(f_1(x,t)\rho_1(x,t)) \\ \nabla\cdot(f_2(x,t)\rho_2(x,t)) \\ \vdots \\ \nabla\cdot(f_N(x,t)\rho_N(x,t)) \end{bmatrix} \quad \Leftrightarrow \quad \frac{\partial \rho(x,t)}{\partial t} = F(u)\rho(x,t)$$

Large-scale optimization problem



- $K=100$
  $dim(\hat{u}^*) = 3K=300$
- Gradient estimation involves computation of the PDE system
- Computationally complex

Minimum Principle for PDE

$$u* = \max_{u \in U_{ad}} \int_X w(x)^T \rho(x,T)dx \Leftrightarrow u* = \min_{u \in U_{ad}} J(u) = \min_{u \in U_{ad}} \int_X -w(x)^T \rho(x,T)dx$$

$$\frac{\partial \rho(x,t)}{\partial t} = L^T(u)\rho(x,t) - \begin{bmatrix} \nabla \cdot (f_1(x,t)\rho_1(x,t)) \\ \nabla \cdot (f_2(x,t)\rho_2(x,t)) \\ \vdots \\ \nabla \cdot (f_N(x,t)\rho_N(x,t)) \end{bmatrix} \Leftrightarrow \frac{\partial \rho(x,t)}{\partial t} = F(u)\rho(x,t)$$

$$u^*(t) = \min_{u \in U_{ad}} H(u) = \min_{u \in U_{ad}} \int_X \pi(x,t)^T F(u)\rho^*(x,t)dx$$

$$\frac{\partial \pi(x,t)}{\partial t} = -F(u^*)^T \pi(x,t) , \quad \pi(x,T) = -w(x)$$

H. O. Fattorini, Infinite Dimensional Optimization and Control Theory

Minimum Principle for PDE

$$u^*(t) = \min_{u \in U_{ad}} H(u) = \min_{u \in U_{ad}} \int_X \pi(x,t)^T L^T(u)\rho^*(x,t)dx$$

$$H(u) = u_1(t)I_1(t)+u_2(t)I_2(t)+u_3(t)I_3(t)$$

$$I_1(t) = \int_X (\pi_1 - \pi_2)\rho_2^* + (\pi_1 - \pi_3)\rho_3^*\, dx$$

$$I_2(t) = \int_X (\pi_2 - \pi_1)\rho_1^* + (\pi_2 - \pi_3)\rho_3^*\, dx$$

$$I_3(t) = \int_X (\pi_3 - \pi_1)\rho_1^* + (\pi_2 - \pi_2)\rho_2^*\, dx$$

$$I_i(t) > 0 \quad \Rightarrow \quad u_i^*(t) = 0$$

$$I_i(t) < 0 \quad \Rightarrow \quad u_i^*(t) = u_{max}$$

$$I_i(t) = 0 \quad \Rightarrow \quad u_i^*(t) = ?$$

$$u(t) = \begin{bmatrix} u_1^*(t) \\ u_2^*(t) \\ u_3^*(t) \end{bmatrix} \quad i=1,2,3$$

Singular Control Problem

Numerical Optimal Control

$$u* = \min_{u \in U_{ad}} J^{\varepsilon}(u) = \min_{u \in U_{ad}} \int_X - w(x)^T \rho(x,T)dx + \varepsilon \int_0^T u_1^2(t) + u_2^2(t) + u_3^2(t)dt$$

$$H^{\varepsilon}(u) = H(u) + \varepsilon(u_1^2(t) + u_2^2(t) + u_3^2(t))$$

$\varepsilon \approx 0$

$T = 3h$

$$u_i^*(t) = -\frac{I_i(t)}{2\varepsilon}, \; u_i^*(t) \in U_{ad}, \; u(t) = \begin{bmatrix} u_1^*(t) \\ u_2^*(t) \\ u_3^*(t) \end{bmatrix}$$

$$10^{-d} > \varepsilon \int_0^T 3u_{max}^2 dt = 3Tu_{max}^2 \Rightarrow \varepsilon < \frac{10^{-d}}{3Tu_{max}^2}$$

Numerical Algorithm (minimizes at each point k)

- Discrete approximation of $u(t) \approx u(k\Delta) = \hat{u}(k)$
- Forward solution for state $\rho(t)$, given $\hat{u}(k)$
- Backward solution for co-state $\pi(t)$, given $\hat{u}(k)$
- Update $\hat{u}(k)$ at each point k towards minimum of $H(k)$

Numerical Optimal Control

$\hat{u}^*$ optimal sequence is a stationary point of iterations

$$\hat{u}^{j+1} = \hat{u}^{j} + \alpha^j d^j$$

satisfying $\quad \hat{J}^{\varepsilon}(\hat{u}^{j} + \alpha^j d^j) < \hat{J}(\hat{u}^j)$

where $\alpha^j \in R$ and search vector $d^j \in R^{dim(\hat{u}^j)}$, $dim(\hat{u}^j) = 3K$

For example $d^j = -\nabla_{\hat{u}} \hat{J}^{\varepsilon}$ $\qquad \nabla_{\hat{u}(k)} \hat{J}^{\varepsilon} = \nabla_{\hat{u}(k)} \hat{H}^{\varepsilon}$

or Nonlinear Conjugate Gradient Method

Search for α^j is computationally expensive, includes solving PDE

$$J^{\varepsilon}(u) = \int_X -w(x)^T \rho(x,T)dx + \varepsilon \int_0^T u_1^2(t) + u_2^2(t) + u_3^2(t)dt$$

we can use $\qquad [\nabla_{\hat{u}} \hat{J}^{\varepsilon}(\hat{u}^j + \alpha^j d^j)]^T d^j = 0$

that has a closed form solution for $\alpha^j = -\dfrac{\Sigma_k \hat{u}^j(k)^T d^j(k)}{\Sigma_k d^j(k)^T d^j(k)} - \dfrac{\Sigma_k \Sigma_i I_i^j(k) d_i^j(k)}{2\varepsilon \Sigma_k d^j(k)^T d^j(k)}$

Nonlinear Conjugate Gradient Method

- Discrete approximation of $u(t) \approx u(k\Delta) = \hat{u}(k)$
- Forward solution for state $\rho(t)$, given $\hat{u}(k)$
- Backward solution for co-state $\pi(t)$, given $\hat{u}(k)$
- Compute : Hamiltonian $H^{\varepsilon}(k)$

  gradient $g^{j} = -\nabla \hat{J}^{\varepsilon}$

  search direction $d^{j} = g^{j} + \beta^{j} d^{j-1}$

  scalar value α^{j}

  control update $\tilde{u}^{j+1} = \hat{u}^{j} + \alpha^{j} d^{j}$

  $\hat{u}_i^{j+1}(k) = 0, \ \tilde{u}_i^{j}(k) < 0$

  $\hat{u}_i^{j+1}(k) = u_{max}, \ \tilde{u}_i^{j}(k) > u_{max}$

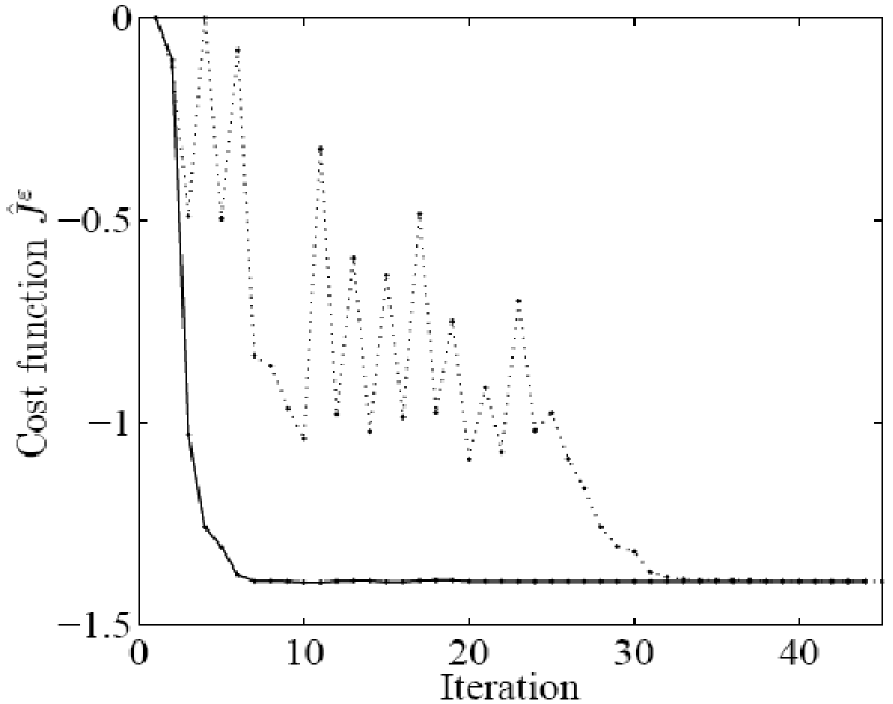
Polak-Ribeire direction

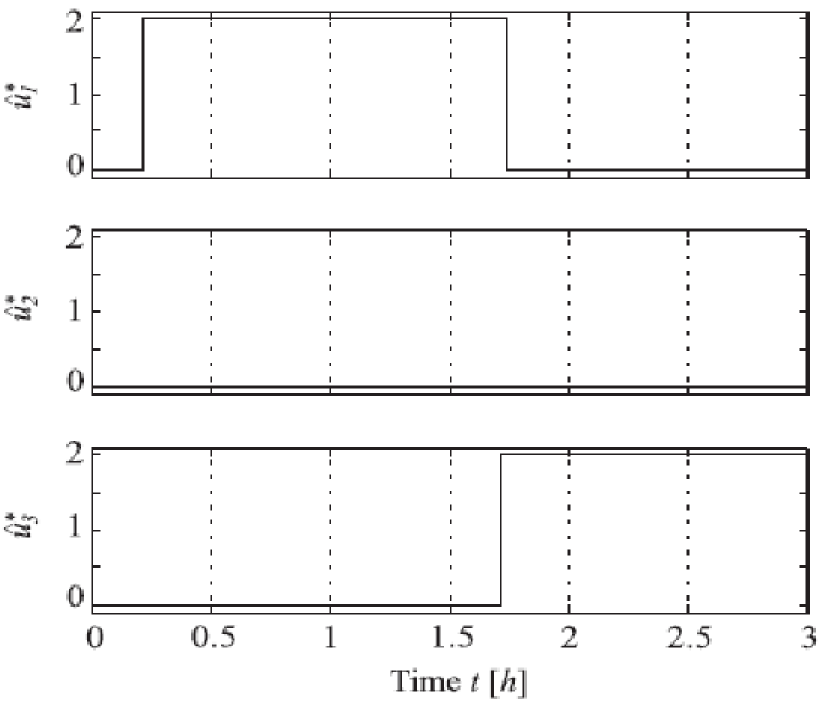
$$\beta^{j} = max\left(\frac{(g^{j})^{T}(g^{j} - g^{j-1})}{(g^{j-1})^{T} g^{j-1}}, 0 \right)$$

$$\beta^{j} = 0, \quad (g^{j-1})^{T} g^{j-1} = 0$$

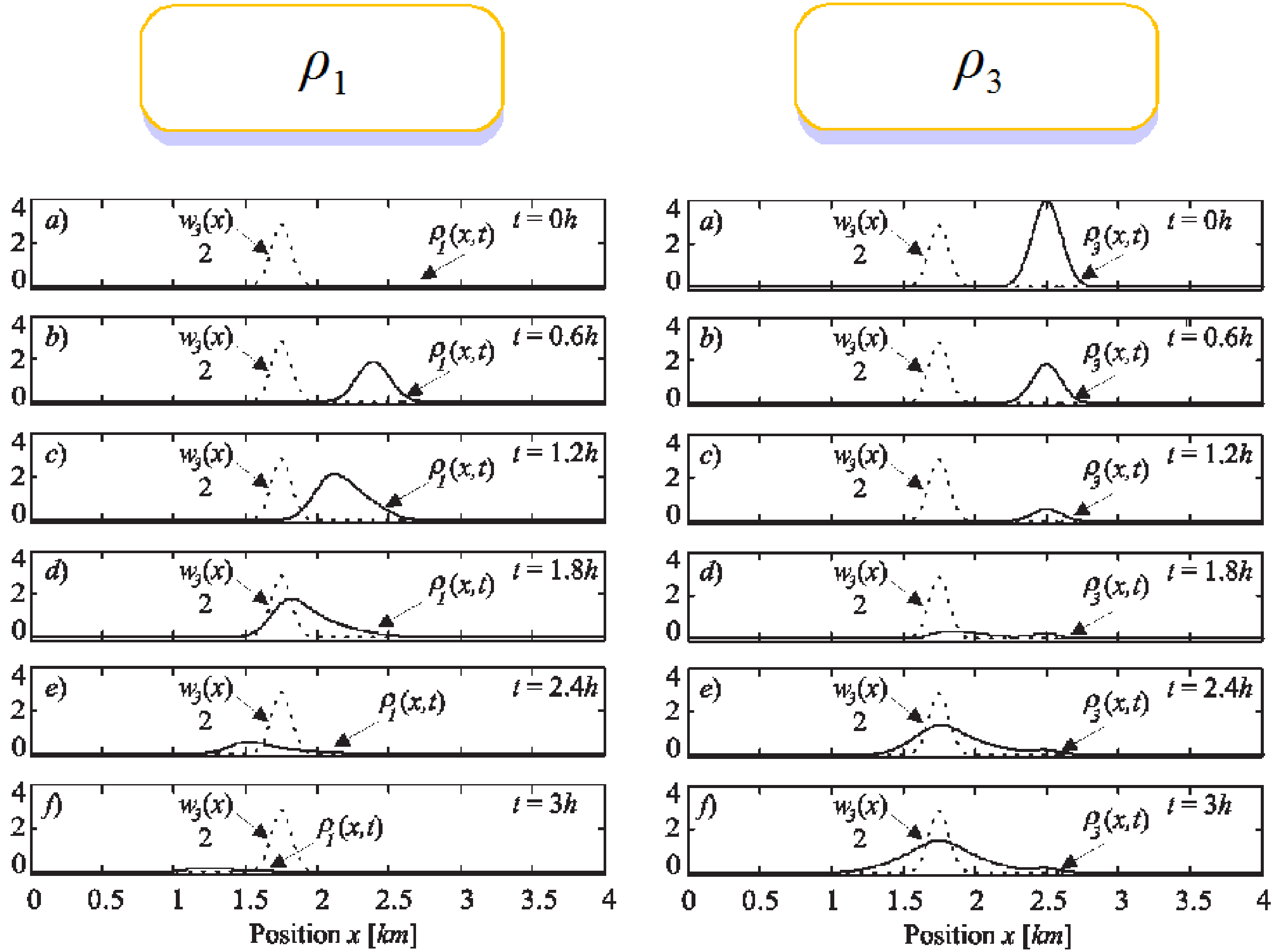
Numerical Optimal Control

Initial guess: u^0=[0.5 0.5 0.5]



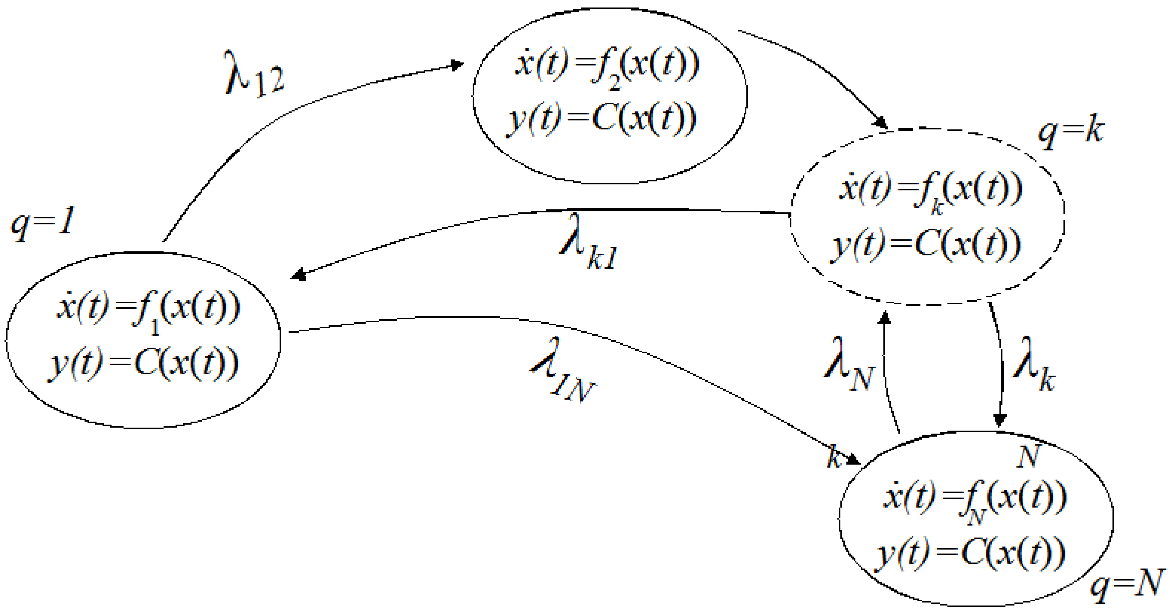


Optimal Control – State Evolution



Why Is the Control of Hybrid System Probability Density Function Important ?

- Practical problems in robotics, manufacturing, traffic management can be described by Hybrid Systems



- Control of transitions is performance based and depends on continuous dynamics
- Control takes into account that some of transitions are controllable, while others are not

Optimal Control Problem Formulation

Stochastic differential equation $dX = b(X, t, u(t))dt + L(X, t, u(t))dw$

$X(t)$ n dimensional stochastic process

dw derivative of n dimensional Wiener process

$u(t)$ control

Probability density function evolution of X *(Fokker-Planck Eq.)*

$$\frac{\partial \rho}{\partial t} = \sum_{i,j=1}^{n} \frac{\partial(-b_i(u)\rho)}{\partial x_i} + \frac{1}{2}\frac{\partial^2([LL^T]_{ij}(u)\rho)}{\partial x_i \partial x_j} = F(u)\rho$$

Scalar product $\langle f, g\rangle = \int_D f(X)g(X)dX$ $\langle \rho, 1\rangle = 1$

Cost function $J(u) = \langle \phi, \rho(T)\rangle + \int_0^T \langle f_0(X, u, t), \rho(t)\rangle dt$

Find the control sequence u (t) that minimizes $J(u)$

Open-loop control problem

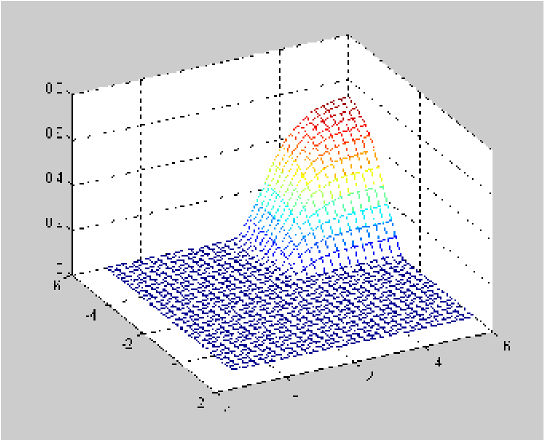
- Taking into account the scalar product definition

$$\langle \phi, \rho(T)\rangle = \int \phi(X)\rho(X, T)dX = E_{\rho(T)}\{\phi(X)\}$$

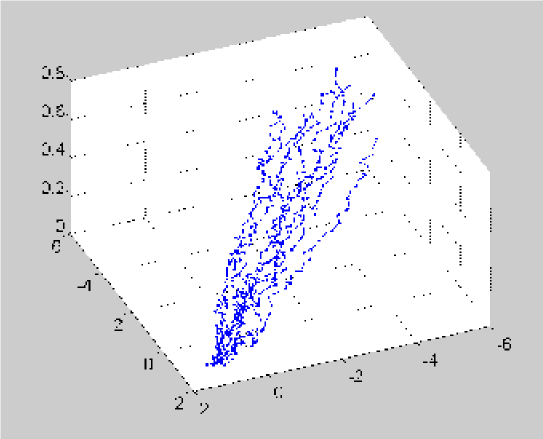
the cost function interpretation is

$$J(u) = E_{\rho(T)}\{\phi(X)\} + \int_0^T E_{\rho(t)}\{f_0(X, u, t)\}dt$$

PDEs vs. Stochastic Processes



PDE-based solution

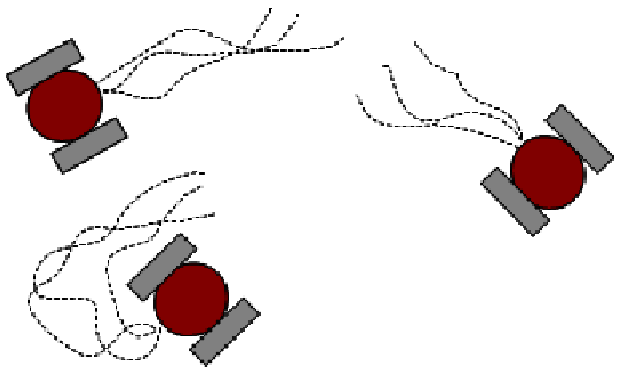


Stochastic process based solution

Milutinović, D., Garg, D. P., A Sampling Approach to Modeling and Control of a Large-size Robot Population, *Proceedings of the 2010 ASME Dynamic Systems and Control Conference (DSCC), Boston, MA*

Milutinović, D., Utilizing Stochastic Processes for Computing Distributions of Large-Size Robot Population Optimal Centralized Control, *Proceeding of the 10th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, Lausanne, Switzerland

Multi-robot systems

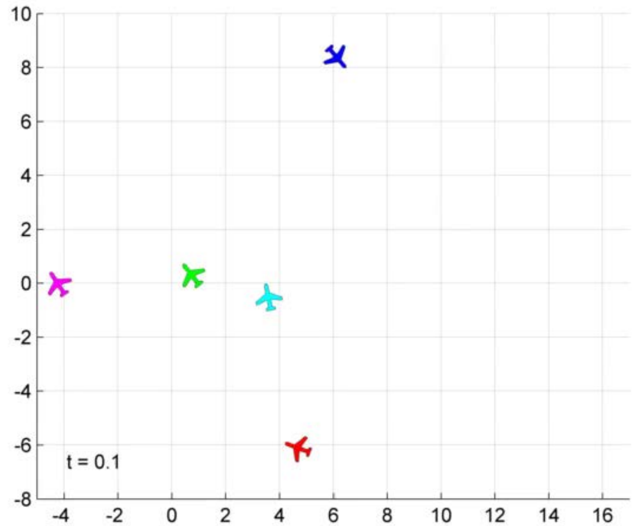


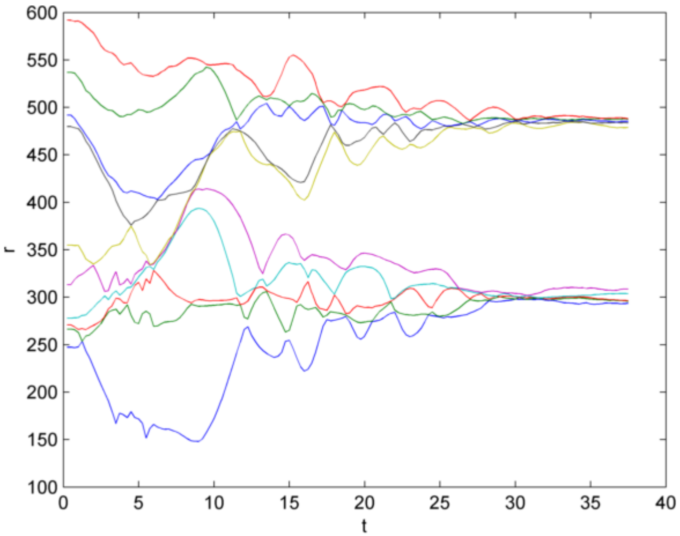
- Each agent adds new degrees of freedom
- More (options) stochastic processes to consider
- Combinatorial expansion of possible ways to control the overall system, due to redundant degrees of freedom

Multi-robot systems

- Robot swarms (control in probability density space)
  - Partial differential equations
  - Trajectory samples

- Robot teams (~10 robots)
  - Path Integral approach + Kalman smoother

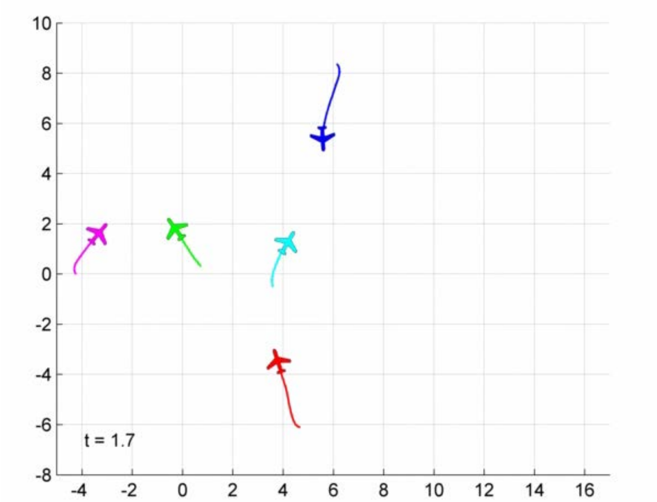




Path Integral Approach: Kappen, H.: Linear Theory for Control of Nonlinear Stochastic Systems. Physical Review, Letters 95(20), 1–4 , 2005

Multi-robot systems

- Robot swarms (control in probability density space)
  - Partial differential equations
  - Trajectory samples
- Robot teams (~10 robots)
  - Path Integral approach + Kalman smoother



The best student paper award:
Anderson, R., Milutinović D., A Stochastic Optimal Enhancement of Feedback Control for Unicycle Formations, *Proc. of the 11th International Symposium on Distributed Autonomous Robotic Systems (DARS'12)*, Baltimore, MD

The Dubins Traveling Salesperson Problem with Stochastic Dynamics (TuAT2.1)

http://users.soe.ucsc.edu/~anderson/

anderson@soe.ucsc.edu

Call for Papers: Special Issue on Stochastic Models, Control and Algorithms in Robotics Submission deadline: November 15, 2013

Guest Editors: Jongeun Choi (MSU), Dejan Milutinović

Editor: Karl Hedrick

Dejan Milutinović, UC Santa Cruz

http://people.ucsc.edu/~dmilutin/

dejan@soe.ucsc.edu

Thank you for your attention !