On distributed control and price mechanisms

Anders Rantzer

Automatic Control LTH, Lund University

With special thanks to Ather Gattami and Michael Rotkowitz
for stimulating discussions

Inspiration from other work

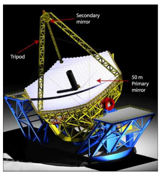
- Spatially invariant systems
  (Bamieh, Paganini, Dahleh, ...)
- Distributed control using dissipativity
  (Langbort, Chandra, d'Andrea, ...)
- Network congestion control
  (Kelly, Paganini, Doyle, Low, ...)
- Stability of Kuramoto's coupled nonlinear oscillators
  (Jadbabaie, Barahona, Motee, ...)
- Distributed average computation
  (Xiao, Boyd, ...)
- The saddle point algorithm
  (Arrow, Hurwitz, Uzawa, ...)

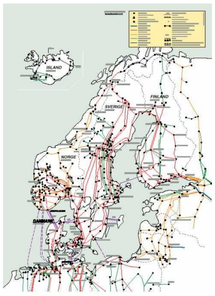
Thanks to Peter Alriksson, Toivo Henningsson, Andreas Wernrud
Vijay Gupta, Cedrik Langbort, Pablo Parrilo, Jeff Shamma, ...

Building theoretical foundations for distributed control

Three major challenges:

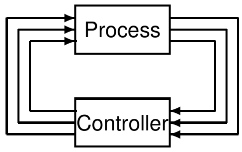
- Rapidly increasing complexity
- Dynamic interaction and couplings
- Information is decentralized





Building theoretical foundations for distributed control

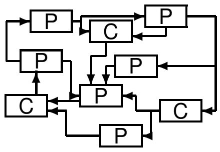
A centralized paradigm dominates theory and curriculum today



$\Downarrow$

We urgently need methodology for

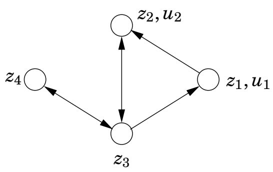
- Decentralized specifications
- Decentralized design
- Verification of global behavior



Outline

- **Distributed information: Team theory**
- Example: Vehicle formation control
- Distributed design: The saddle point algorithm
- Example: Trading a commodity
- Conclusions

A Linear Quadratic Dynamic Team Problem



Find μ_1, μ_2 to minimize the stationary variance

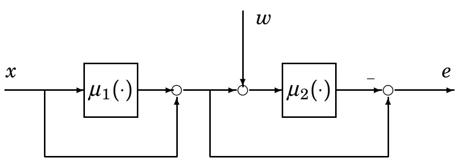
$$\mathbf{E} \sum_{i,j} (|z_i|^2 + |u_j|^2)$$

$$\begin{bmatrix} z_1(k+1) \\ z_2(k+1) \\ z_3(k+1) \\ z_4(k+1) \end{bmatrix} = \begin{bmatrix} \Phi_{11} & 0 & \Phi_{13} & 0 \\ \Phi_{21} & \Phi_{22} & \Phi_{23} & 0 \\ 0 & \Phi_{32} & \Phi_{33} & \Phi_{34} \\ 0 & 0 & \Phi_{43} & \Phi_{44} \end{bmatrix} \begin{bmatrix} z_1(k) \\ z_2(k) \\ z_3(k) \\ z_4(k) \end{bmatrix} + \begin{bmatrix} \Gamma_1 u_1(k) + w_1(k) \\ \Gamma_2 u_2(k) + w_2(k) \\ w_3(k) \\ w_4(k) \end{bmatrix}$$

$$u_1(k) = \mu_1\Big(\bar{y}_1(k), \bar{y}_2(k-2), \bar{y}_3(k-1), \bar{y}_4(k-2)\Big)$$
$$u_2(k) = \mu_2\Big(\bar{y}_1(k-1), \bar{y}_2(k), \bar{y}_3(k-1), \bar{y}_4(k-2)\Big)$$

$$\bar{y}_i(k) = \begin{bmatrix} C_i z_i(k) \\ C_i z_i(k-1) \\ C_i z_i(k-2) \\ \vdots \end{bmatrix}$$

The Witsenhausen counterexample



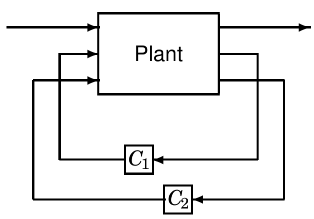
Minimize $\mathbf{E}\left(\left|x + \mu_1(x) - \mu_2(x + \mu_1(x) + w)\right|^2 + \left|\mu_1(x)\right|^2\right)$

when x and w are given Gaussian variables.

The best controllers are not linear, because for a fixed output variance of μ_1, a non-Gaussian signal can transfer more information than a Gaussian one.

[Witsenhausen (1968) A counterexample in stochastic control]

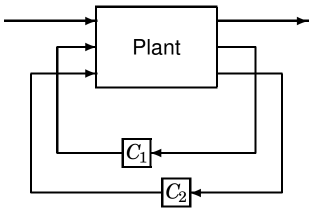
An incentive for signalling



If one controller has information useful for the other, then there is an incentive to encode this information in the control inputs.

This "signalling" creates complicated nonlinear control laws.

The signalling incentive sometimes disappears!



[Yu-Chi Ho and K'ai-Ching Chu (1972)]:

If a decision-makers action affects our information, then knowing what he knows will yield linear optimal solutions

Convexity in distributed control

A recent observation by [Bamieh, Voulgaris (2002)] and [Rotkowitz, Lall (2002)]:

When the Ho-Chu condition holds, i.e. *communication links propagate information at least as fast as the process does.*, the distributed control synthesis problem becomes convex:

$$\text{Minimize } \|T_{11} + T_{12}QT_{21}\|^2_{\mathbf{H}_2}$$

where

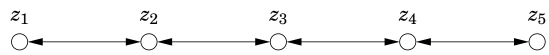
$$Q(z) = \begin{bmatrix} q_{11}(z) & z^{-2}q_{12}(z) & z^{-1}q_{13}(z) & z^{-2}q_{14}(z) \\ z^{-1}q_{21}(z) & q_{22}(z) & z^{-1}q_{23}(z) & z^{-2}q_{24}(z) \end{bmatrix}$$

The objective function is a quadratic function of the transfer functions $q_{ij}(z)$, so in principle this is a classical LQG problem!

Outline

- Distributed information: Team theory
- **Example: Vehicle formation control**
- Distributed design: The saddle point algorithm
- Example: Trading a commodity
- Conclusions

A vehicle formation



The objective is to make $\mathbf{E}|z_{i+1} - z_i|^2$ small for $i = 1, \dots, 4$.

Each vehicle obeys the independent dynamics

$$z_i(k+1) = z_i(k) + u_i(k) + w_i(k)$$

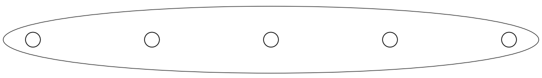
We will do this by minimizing

$$\mathbf{E}\sum_{i=1}^{4}|z_{i+1} - z_i|^2 + \mathbf{E}|z_1|^2 + \mathbf{E}|z_5|^2 + 10\,\mathbf{E}\sum_{i=1}^{5}|u_i|^2$$

subject to different information constraints

A vehicle formation

Full state information everywhere:

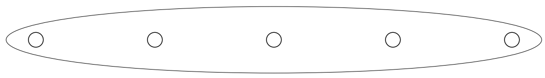


$\mathbf{E}\,|z_1 - z_2|^2 = 3.13$ $\mathbf{E}\,|z_3 - z_4|^2 = 3.16$

$\mathbf{E}\,|z_2 - z_3|^2 = 3.16$ $\mathbf{E}\,|z_4 - z_5|^2 = 3.13$

A vehicle formation

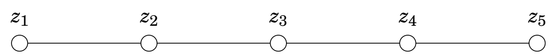
Full state information everywhere:



$\mathbf{E}\,|z_1 - z_2|^2 = 3.13$ $\mathbf{E}\,|z_3 - z_4|^2 = 3.16$

$\mathbf{E}\,|z_2 - z_3|^2 = 3.16$ $\mathbf{E}\,|z_4 - z_5|^2 = 3.13$

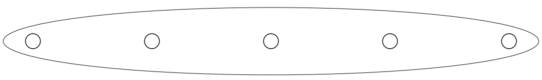
No communication:



$\mathbf{E}\,|z_1 - z_2|^2 = 3.40$ $\mathbf{E}\,|z_3 - z_4|^2 = 3.40$

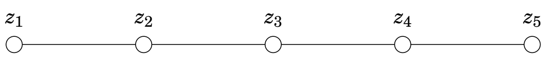
$\mathbf{E}\,|z_2 - z_3|^2 = 3.40$ $\mathbf{E}\,|z_4 - z_5|^2 = 3.40$

A vehicle formation



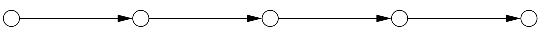
$\mathbf{E}\,|z_1 - z_2|^2 = 3.13$ $\mathbf{E}\,|z_3 - z_4|^2 = 3.16$

$\mathbf{E}\,|z_2 - z_3|^2 = 3.16$ $\mathbf{E}\,|z_4 - z_5|^2 = 3.13$



$\mathbf{E}\,|z_1 - z_2|^2 = 3.40$ $\mathbf{E}\,|z_3 - z_4|^2 = 3.40$

$\mathbf{E}\,|z_2 - z_3|^2 = 3.40$ $\mathbf{E}\,|z_4 - z_5|^2 = 3.40$



$\mathbf{E}\,|z_1 - z_2|^2 = 3.30$ $\mathbf{E}\,|z_3 - z_4|^2 = 3.33$

$\mathbf{E}\,|z_2 - z_3|^2 = 3.32$ $\mathbf{E}\,|z_4 - z_5|^2 = 3.25$

Outline

- Distributed information: Team theory
- Example: Vehicle formation control
- **Distributed design: The saddle point algorithm**
- Example: Trading a commodity
- Conclusions

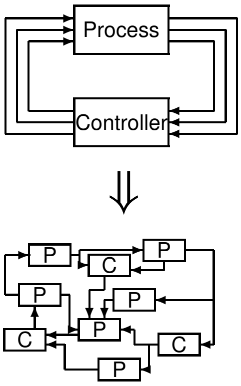
Building theoretical foundations for distributed control

Team theory can design a set of controllers to cooperate as a team with a common objective

But ...

What if the global model is too complex to handle?

What if each controller has its own objective?



Dual decomposition — The saddle point algorithm

$$\min_{x,y,z,w}[\phi_1(x,y)+\phi_2(y,z)+\phi_3(z,w)]$$
$$=\max_{p,q}\min_{x,y_1,y_2,z_2,z_3,w}[\phi_1(x,y_1)+\phi_2(y_2,z_2)+\phi_3(z_3,w)+p(y_1-y_2)+q(z_2-z_3)]$$

Update in gradient direction:

Computer 1: $\begin{cases}\dot{x} &= -\partial\phi_1/\partial x\\ \dot{y}_1 &= -\partial\phi_1/\partial y_1 - p\end{cases}$

Computer 1 and 2: $\dot{p} = y_1 - y_2$

Computer 2: $\begin{cases}\dot{y}_2 = -\partial\phi_2/\partial y_2 + p\\ \dot{z}_2 = -\partial\phi_2/\partial z_2 - q\end{cases}$

Computer 2 and 3: $\dot{q} = z_2 - z_3$

Computer 3: $\begin{cases}\dot{z}_3 = -\partial\phi_3/\partial z_3 + q\\ \dot{w} = -\partial\phi_3/\partial w\end{cases}$

Converges whenever ϕ_i are convex! [Arrow, Hurwicz, Usawa 1958]

Global stability of saddle algorithm

$$\min_{Cx=0}\phi(x)=\max_p\min_x[\phi(x)+p^TCx]$$

$$\begin{bmatrix}\dot{p}\\ \dot{x}\end{bmatrix}=\begin{bmatrix}Cx\\ -C^Tp-\partial\phi/\partial x\end{bmatrix}$$

$$\begin{bmatrix}\ddot{p}\\ \ddot{x}\end{bmatrix}=\begin{bmatrix}0 & C\\ -C^T & -\partial^2\phi/\partial x^2\end{bmatrix}\begin{bmatrix}\dot{p}\\ \dot{x}\end{bmatrix}$$

$$V=|\dot{p}|^2+|\dot{x}|^2$$

$$\begin{aligned}\dot{V}&=\dot{p}^T\ddot{p}+\dot{x}^T\ddot{x}\\ &=\dot{p}^T(C\dot{x})+\dot{x}^T(-C^T\dot{p}-(\partial^2\phi/\partial x^2)\dot{x}\\ &=-\dot{x}^T(\partial^2\phi/\partial x^2)\dot{x}\end{aligned}$$

Network congestion control

Maximize $U_i(x)$ over $x_i \geq 0$ subject to $\sum_i R_{li}x_i \leq c_l$

Introduce prices p_l for the links

$$\min_{p_l\geq 0}\max_{x_i\geq 0}\sum_i\left[U_i(x_i)-\sum_l p_l\left(R_{li}x_i-c_l\right)\right]$$
$$=\min_{p_l\geq 0}\max_{x_i\geq 0}\sum_i\left[U_i(x_i)-x_i\sum_l p_lR_{li}\right]+\sum_l p_lc_l$$

The gradient algorithm is used for updates of prices and flows

Synchronization of oscillators

Oscillators with nonlinear coupling

$$\ddot{\theta}_i=-\textstyle\sum_j\left[\sin(\theta_i-\theta_j)-d(\dot{\theta}_i-\dot{\theta}_j)\right]$$

In state space form

$$\begin{aligned}\dot{\theta}_i&=\omega_i\\ \dot{\omega}_i&=-\textstyle\sum_j\sin(\theta_i-\theta_j)-d\omega_i\end{aligned}$$

Lyapunov stability analogous to the saddle algorithm

$$V=\sum_{i,j}\left[\left(\sin\frac{\theta_i-\theta_j}{2}\right)^2+\left(\frac{\omega_i-\omega_j}{2}\right)^2\right]$$
$$\dot{V}=-\frac{d}{2}\sum_{i,j}|\omega_i-\omega_j|^2$$

Outline

- Distributed information: Team theory
- Example: Vehicle formation control
- Distributed design: The saddle point algorithm
- **Example: Trading a commodity**
- Conclusions

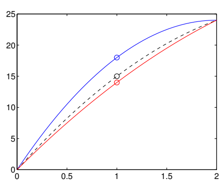
Example: Three Trading Units (The Beer Game)

Given endowments w_1, w_2, w_3, optimize trade quantities u_1, u_2:

$$\max_{u_1,u_2}[U_1(w_1+u_1)+U_2(w_2-u_1+u_2)+U_3(w_3-u_2)]$$

Example:

$$\begin{aligned}U_1(x_1)&=24-6(x_1-2)^2\\ U_2(x_2)&=27-3(x_2-3)^2\\ U_3(x_2)&=32-2(x_3-4)^2\end{aligned}$$



Dual Decomposition

Given endowments w_1, w_2, w_3, optimize trade quantities u_1, u_2:

$$\max_{u_1,u_2}[U_1(w_1+u_1)+U_2(w_2-u_1+u_2)+U_3(w_3-u_2)]$$
$$=\min_{\lambda_{jk}}\max_{u_{jk}}\Big[U_1(w_1+u_1)+U_2(w_2-v_1+u_2)+U_3(w_3-v_2)$$
$$+\lambda_1(u_1-v_1)+\lambda_2(u_2-v_2)\Big]$$

For fixed prices λ_{jk}, the inner maximization decomposes into three separate optimization problems

$$\max_{u_1}\left[U_1(w_1+u_1)-\lambda_1u_1\right]$$
$$+\max_{v_1,u_2}\left[U_2(w_2-v_1+u_2)+\lambda_1v_1-\lambda_2u_2\right]$$
$$+\max_{v_2}\left[U_3(w_3-v_2)+\lambda_2v_2\right]$$

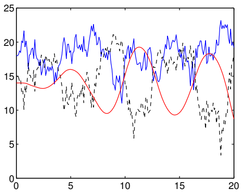
Gradient dynamics become decentralized!

Saddle Dynamics for the Three Trading Units

$$U_1(x_1) = 24 - 6(x_1 - 2)^2$$
$$U_2(x_2) = 27 - 3(x_2 - 3)^2$$
$$U_3(x_2) = 32 - 2(x_3 - 4)^2$$

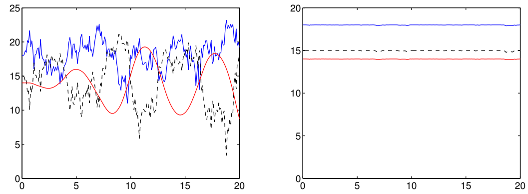
$$\lambda(t + h) = \lambda(t) - (\partial U/\partial\lambda)h$$
$$x(t + h) = x(t) + (\partial U/\partial x)h$$

$U_i(t)$ saddle dynamics have been simulated when w_1 is noisy with spectral density $|e^{0.1i\omega} - e^{-0.1}|^{-2}$
(a beer game with stochastic model for customer demand)



Why so oscillatory? Can stochastic team theory help?

Optimal Team Solution Drastically Better



The saddle algorithm is decentralized, but is hard to tune.

The stochastic team theory allows for optimal performance, provided that a global model is available.

Can we get the best of both worlds?

Dual decomposition on stochastic processes

$$\max_{u_1,u_2} \mathbf{E}\left[U_1(w_1 + u_1) + U_2(w_2 - u_1 + u_2) + U_3(w_3 - u_2)\right]$$
$$= \min_{\lambda_{jk}} \max_{u_{jk}} \mathbf{E}\Big[U_1(w_1 + u_1) + U_2(w_2 - v_1 + u_2) + U_3(w_3 - v_2)$$
$$+ \lambda_1(u_1 - v_1) + \lambda_2(u_2 - v_2)\Big]$$

- Prices and decisions are stochasic processes
- Information delays imposed as correlation constraints
- A sequence of negotiations converges to optimum
- No global model is needed
- Optimal team solution is multi-player Nash equilibrium

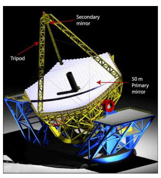
Outline

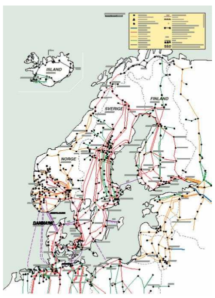
- Distributed information: Team theory
- Example: Vehicle formation control
- Distributed design: The saddle point algorithm
- Example: Trading a commodity
- **Conclusions**

Conclusions

Three major challenges:

- Rapidly increasing complexity
- Dynamic interaction and couplings
- Information is decentralized





Today's message: The key to handle complexity is decomposition. A powerful theory is starting to emerge!

Unlimited research needs

- Implement the negotiation procedure for design
- Who loses when one player behaves suboptimally?
- Utilize graph structure
- Nonlinear versions
- ...