

Intro to Complex Systems

Course Content

Schedule

Intro	■ week 1 - Overview, introduction
OO	■ week 2 - Ordinal optimization, the fundamentals, the unconstrained case; ■ week 3 - Ordinal optimization, the extensions to VOO, COO, selection rules;
MDP	■ week 4 - MDP models, VI, PI (model-based) ■ week 5 - MDP algorithms (large state space, large action space) - unknown models?
RL	■ week 6 - RL elementary (model-free) ■ week 7 - RL supplementary ■ week 8 - RL algorithms (TRPO, GAE, PPO, DDPG)
EBO	■ week 9 - Event-based optimization (models) ■ week 10 - EBO key equations (performance difference, derivative) ■ week 11 - EBO algorithms (policy iteration, policy gradient)
SBO	■ week 12 - No-Free-Lunch-Theorem ■ week 13 - Elements of Simulation, OCBA ■ week 14 - Nested Partitions, memoryless triangle ■ week 15 - Queueing theory fundamentals (Little's Law) + introduction to final quiz. ■ week 16 - Student presentation (may need to schedule a different time)

Intro to DECS

1. What is Complex System:
 - simulation-based
 - takes long time to simulate
2. Time cost of simulation
 - a computer network, an electronic grid ...
 - several days --> optimize
3. What can we do?
 - **Model** the DEDS
 - **Simulate** the DEDS
 - **Evaluate** the DEDS
 - **Optimize** the DEDS
4. What is DEDS:
 - Discrete event dynamic system
 - state s_i is concrete
 - state transition mechanism is **event-driven**
e.g. Airport, The Internet
5. Nature of DEDS:
 - A set of tasks

- A set of resources
- Routing (among resources)
- Scheduling (among tasks)

Just like an optimization problem.....

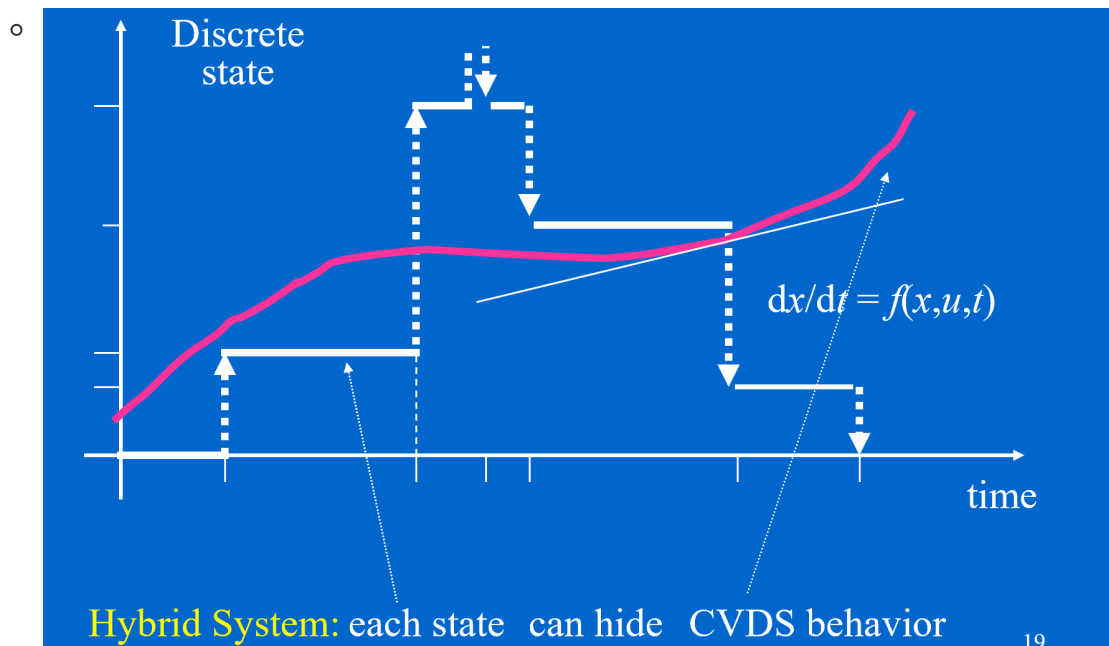
6. Comparison with a CVDS Trajectory:

- y axis --> event --> no value

Q: If we integrate the DEDS trajectory several times, we shall also get a smooth trajectory. What is wrong with this?

A: Because there are no metric assigned on the state space of a DEDS.

Integration of the DEDS trajectory makes no sense, because the units of vertical axes has no meaning only graphical.



7. Modeling Ingredients:

- Discrete State:
combinatorial explosion
- Stochastic Effects
- Hierarchical:
depending on what level is needed to optimize the performance of interest
- Computational
computationally feasible algorithms

8. Mathematical Specification:

- state approach
 - X : finite state space
 - A : event space, e.g. arrival
 - $\Gamma(x)$: enabled event space
 - f : state transition function
- input / output approach:
 - string & language

- clock mechanism:
 - $c_n(\alpha)$: the n-th lifetime of event α
 - $\tau_n(\alpha)$: the time of the n-th occurrence of event α
- Different Models:

	FSM (Markov Chains)	Queuing Network	Min-Max Algebra	Petri Nets	Language & Processes	GSMP
STATE	yes	yes	yes	graphical	no	yes
EVENT	input	yes	yes	yes	yes	yes
FEASIBLE EVENT	yes	yes	yes	yes	not really	yes
TIME	no	yes	(yes)	yes	no	yes
TRANSITION	yes	yes	yes	yes	no	yes
RANDOM- NESS	no/yes	yes	no	yes	no	yes

25

9. The workflow:

- building models
- validation and analysis
- **evaluating the model**
- **optimization and tuning**

10. Performance Evaluation:

- sensitivity analysis: $J(\theta + \Delta\theta)$
- performance surface: $J(\theta)$ at $\theta^{(i)}$
- find optimal parameter $\theta_{optimal}(t)$

11. Three Approaches to a Simple Control Problem:

- open loop control
- feedback control
- stochastic control