Can System Dynamics Models Learn?

George P. Richardson
Rockefeller College of Public Affairs and Policy
University at Albany - State University of New York
G.P.Richardson@Albany.edu

Can System Dynamics Models Learn and Adapt?

- Of course not! Models don't learn; people do.
- Learning and adaptation require internal changes,
- And system dynamics models have fixed structure (equations),
- So system dynamics models can't learn.
- But how close can they come?

Can System Dynamics Models Learn and Adapt?

- System dynamics models can change dominant structure:
  - Nonlinearities are our source of endogenous system change
- So the question becomes
  - How close can endogenous shifts in loop dominance, generated by nonlinearities, come to resemble learning?
- And the answer is: Pretty close!

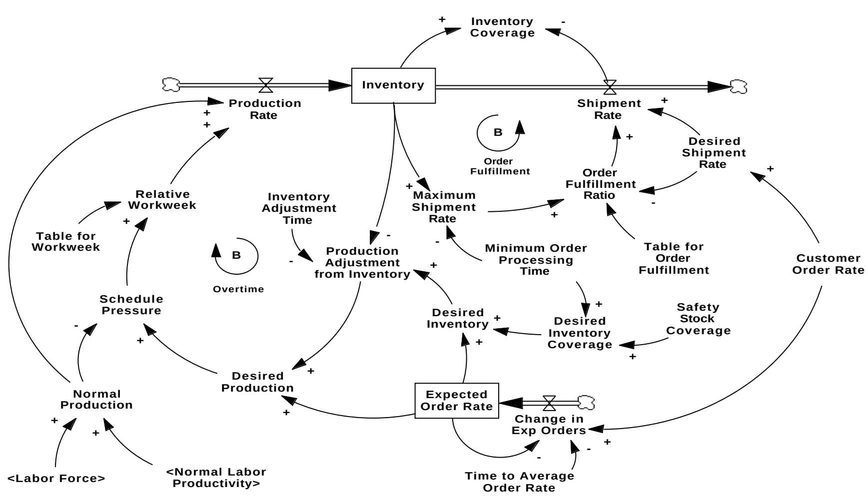
What is Required?

- Model structure for the *self-perception of model behavior*
- Model structure for *adaptation and 'learning'*
- And perhaps, model structure for endogenously generated *experimentation leading to 'learning'*

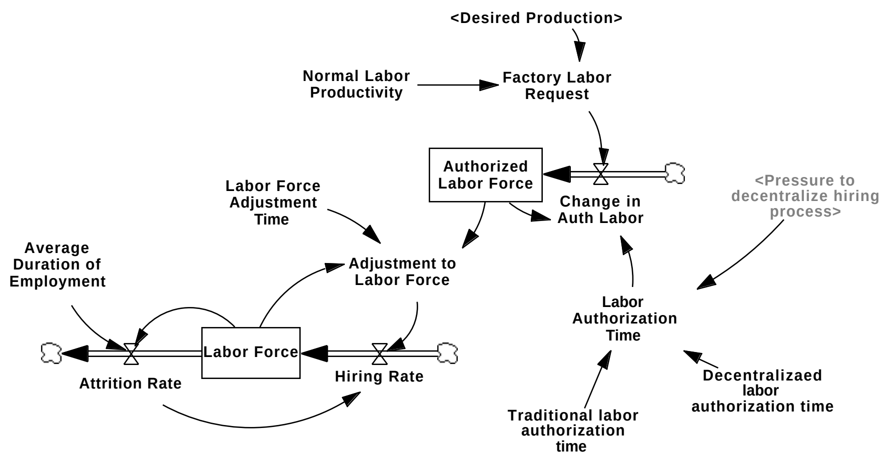
An example:
Inventory / Workforce Instabilities

- Classic structure of Inventory / Workforce oscillations in response to randomness in customer orders
- Oscillations stem from delays in adjusting the workforce to changes in the desired production schedule
- Two policies dampen the oscillations: undertime/overtime and re-engineering hiring

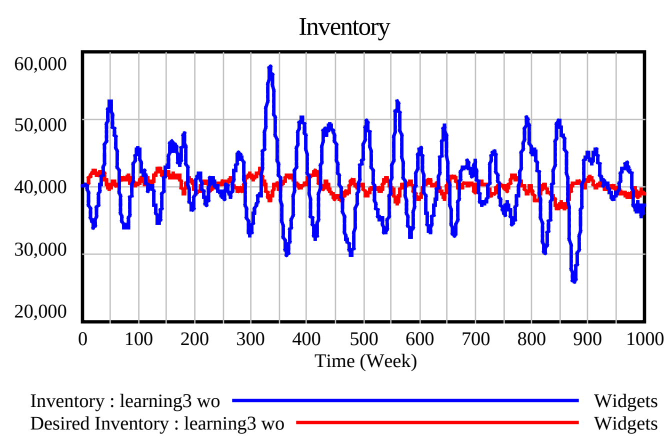
Inventory, Production, & Shipments



Hiring Delays

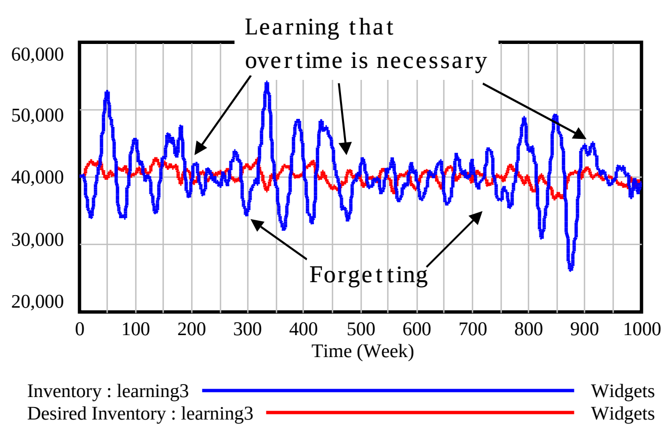


Persistent oscillations in Inventories

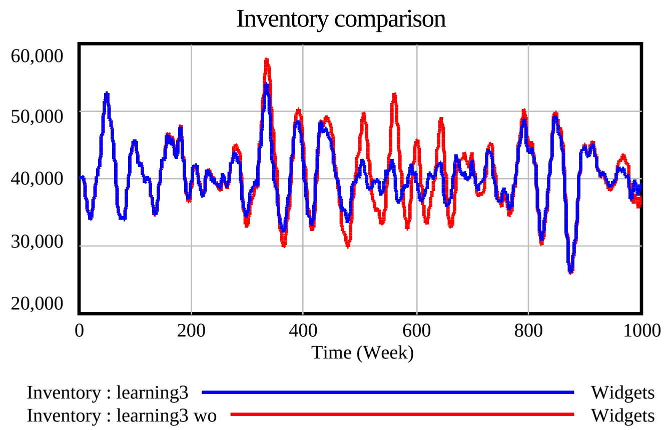


…and the Laborforce

Learning and forgetting



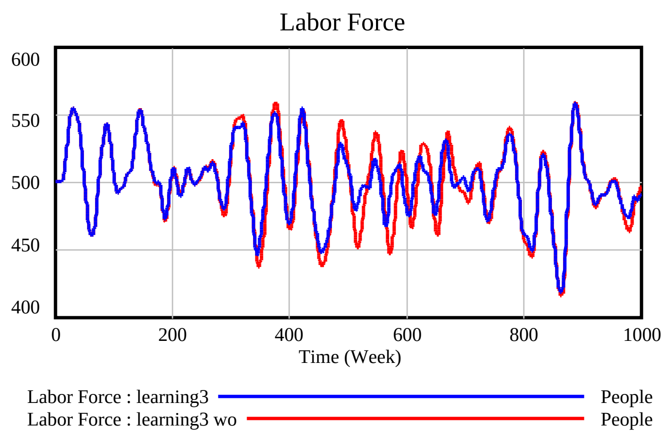
Inventory, with and without learning



G.P.Richardson, Rockefeller College, University at Albany

11

Labor, with and without learning



G.P.Richardson, Rockefeller College, University at Albany

12

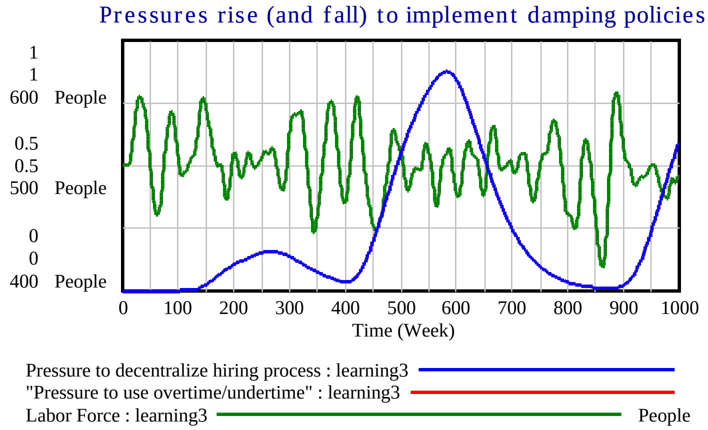
How does the model ‘learn’?

- It ‘perceives’ that it, itself, is oscillating,
  - In particular, that the Laborforce is oscillating.
- It ‘perceives’ the period and amplitude of its own oscillations, much as a person would.
- It comes to perceive that the amplitude of its oscillations is above its tolerance,
- So ‘pressures’ mount to phase in undertime & overtime and decentralize hiring

How does the model ‘forget’?

- The model ‘perceives’ that its Laborforce variations are within its acceptable tolerance,
- So the pressures for undertime & overtime and removing hiring delays subside,
- And the damping policies are phased out -- the model ‘forgets’ that it ever had a problem.

The dynamics of pressures to adjust



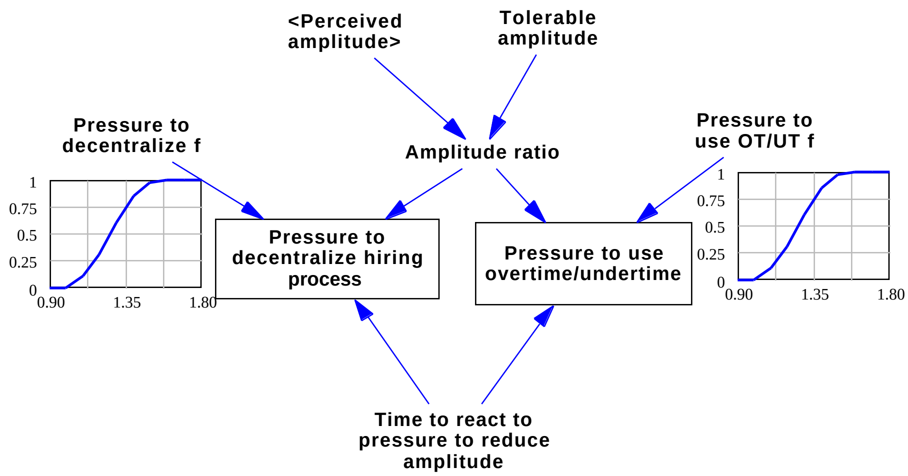
Modeling pressures to adjust

- Perceive peaks and valleys of the oscillations
- Estimate period and amplitude
- Pressure to adjust = f(Amplitude/Tolerance)
- Pressures to adjust are then applied to hiring delays and the undertime/overtime policy
- Relative workweek =

  Pressure to use overtime/undertime*Table for Workweek(Schedule Pressure) + (1-Pressure to use overtime/undertime)*1
- Labor authorization time =

  Pressure to decentralize hiring process*Decentralizaed labor authorization time + (1-Pressure to decentralize hiring process)*Traditional labor authorization time

Modeling pressures to adjust



G.P.Richardson, Rockefeller College, University at Albany

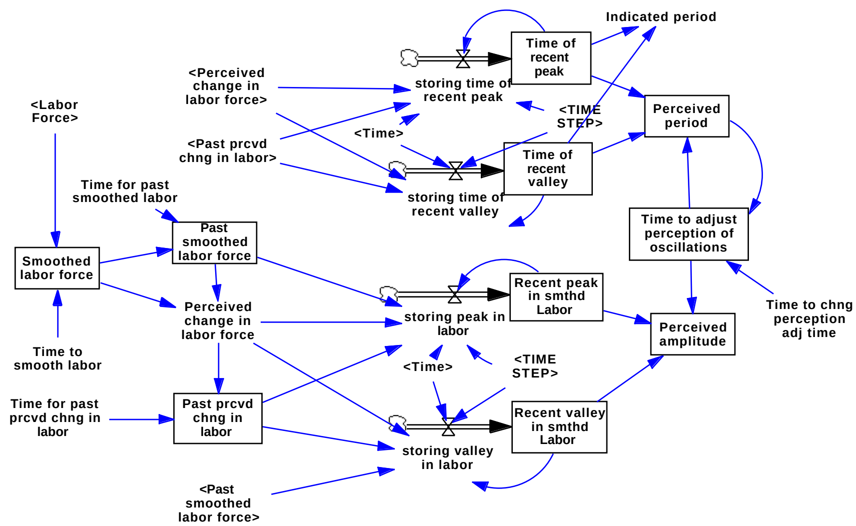
17

Perceiving peaks, valleys, period, and amplitude

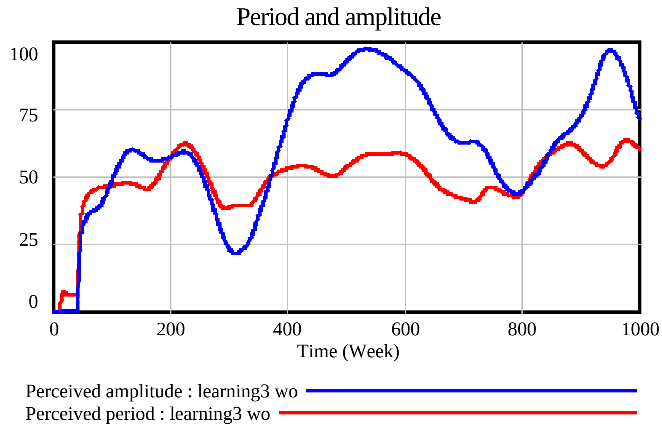
G.P.Richardson, Rockefeller College, University at Albany

18

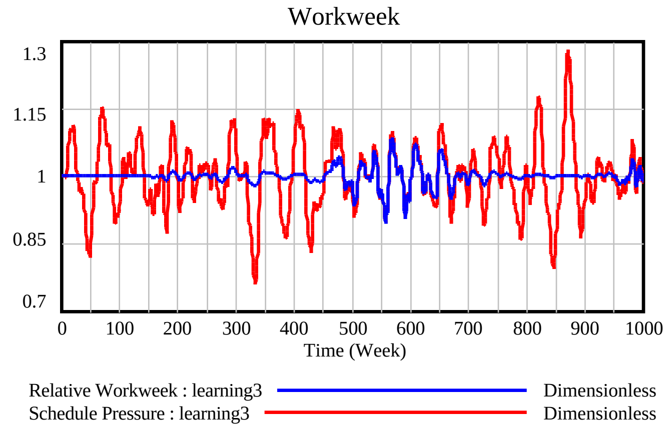
The hard part is modeling the perception of peaks and valleys...



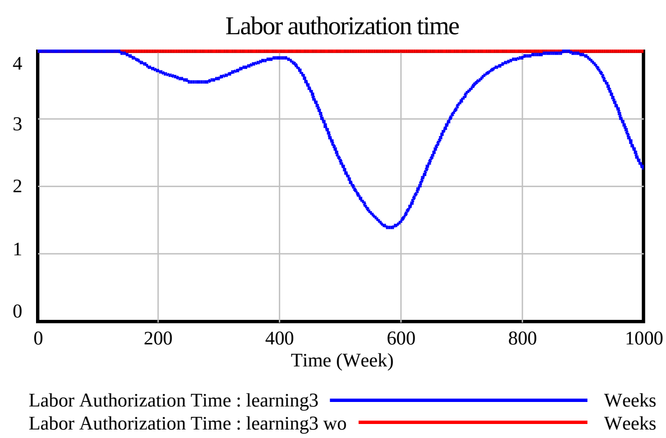
Perception of period and amplitude



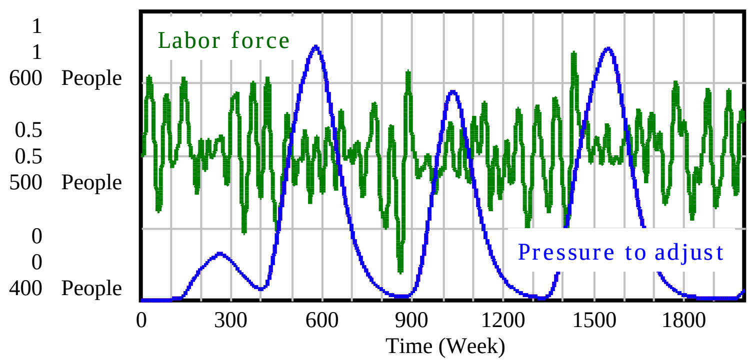
Learning to adjust the workweek policy



Learning to re-engineer the hiring policy



Long run dynamics

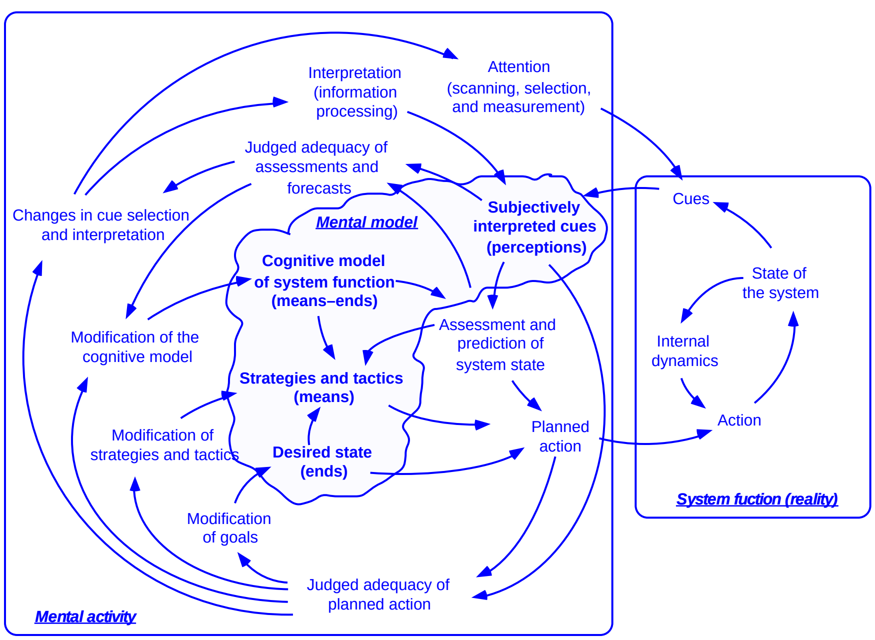


What have we learned?

- If the model didn't forget, the problem would be solved.
- Nonlinearities enable continuous models to adapt and change over time.
- Modeling the model's perception of its own dynamics is tricky.
- We are thinking of 'learning' as *purposeful adaptation in response to system behavior to come closer to goals.*
- So far, the model can 'learn' and 'forget' only what we tell it to.

Can a model *explore* multiple policies and *select* on its own the most advantageous?

- Why not?
- In addition to what we've seen, the model would require:
  - an *Exploration* sector that sets out the structure for explorations;
  - A *Selection* sector that contains criteria for evaluation of the model's own dynamic behavior.
- None of that seems impossible, but it could be daunting...



Why build models that learn?

- To achieve real-time adaptive control at the policy level
- To compress human learning time
  - Ask models to show us what we can’t learn without them
- To prove that we can do it
- Because we’ve solved all the easier dynamic problems
  - Not bloody likely!
- Because it’s New Year’s Eve and we’re looking for something to do...

Further reading

- Self-learning policies in Urban Dynamics, *Readings in Urban Dynamics II* (1975).
- DeJong, Learning to plan in continuous domains. *Artificial Intelligence* 65 (1994).
- Ram & Santamaria, Continous case-based reasoning. *Artifical Intelligence* 90 (1997)
- Richardson, Andersen, Maxwell & Stewart, Foundations of Mental Model Research (1994)
- Powers, *Behavior, the Control of Perception* (1973)