



Artificial Life & Complex Systems

Lecture 9

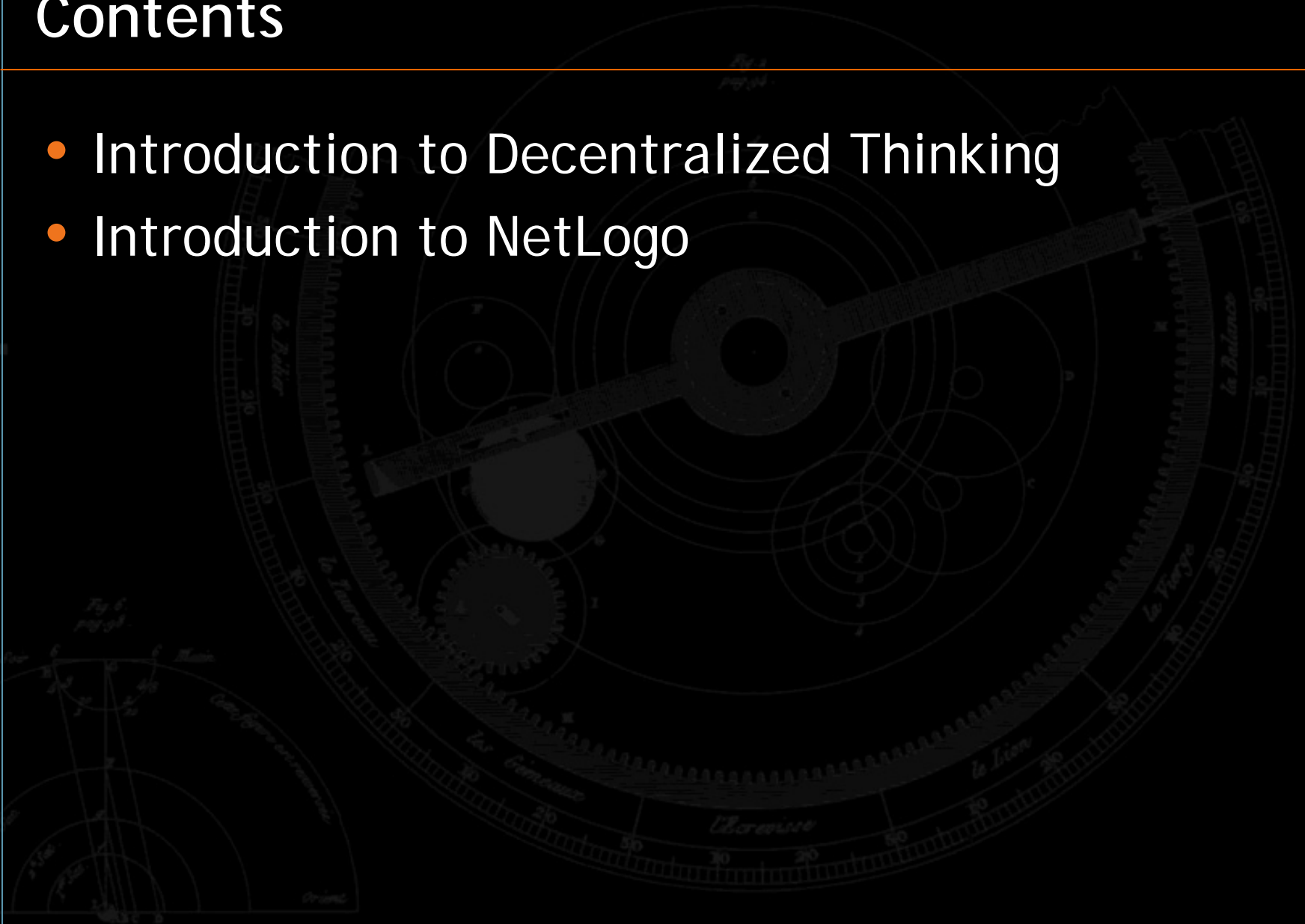
NetLogo

June 1, 2007

Max Lungarella

Contents

- Introduction to Decentralized Thinking
- Introduction to NetLogo

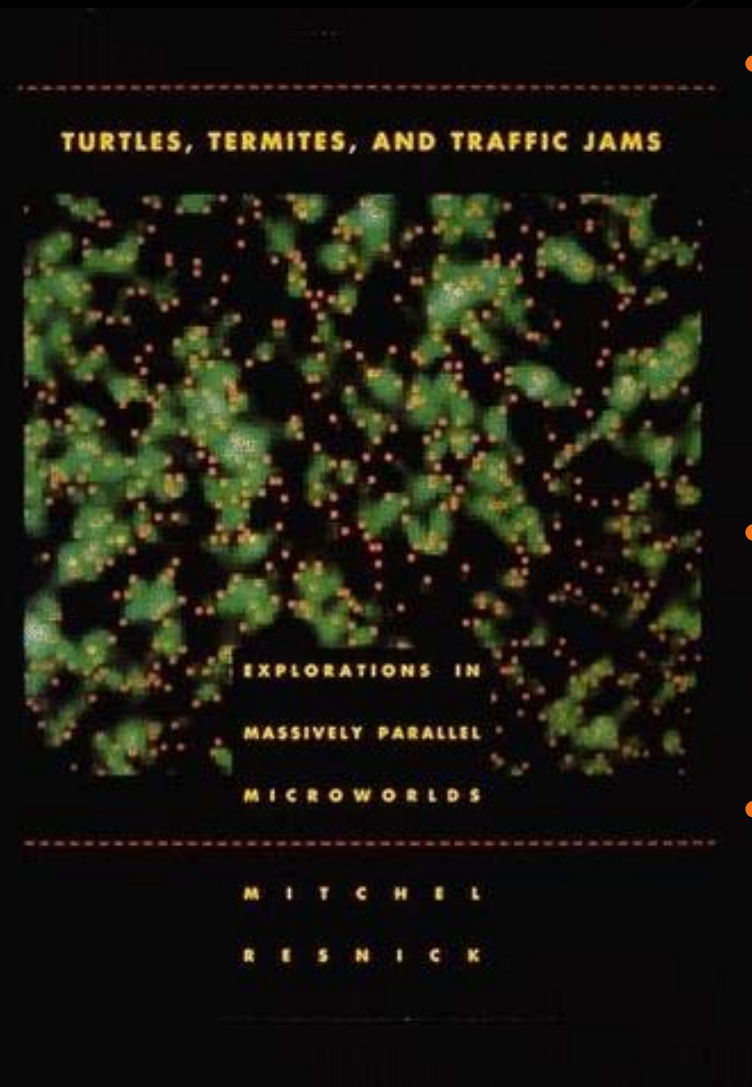


Centralized Mindset

Everything must have a single cause, an ultimate controlling factor

“When people see patterns in the world (like a flock of birds), they often assume that there is some type of centralized control (leader of the flock).”
(Resnick, 1997)

Manifesto of Decentralized Thinking



Resnick, 1997

- *Adam Smith (1776):* "Each individual in a society neither intends to promote the public interest, nor knows how much he is promoting it ... he intends only his own gain, and he is in this led by an invisible hand to promote an end which was no part of his invention."
- Image of "invisible hand" (i.e. economic order and justice can be achieved without centralized control of the economy)
- *Charles Darwin (1860):* Who or what is responsible for organized complexity of living systems? Order and complexity arise from decentralized processes of variation and selection.

Manifesto of Decentralized Thinking

Decentralization in AI and ALife

- Early AI (1950s-1970s) was centralized; model-based, relying on inference engine (sense-model-plan-act)
- Early ALife (1950s) was, however, decentralized (example: cellular automata)
- Less early AI (1980s-1990s) was decentralized
 - Resurgence of neural networks with Parallel Distributed Processing (1986): high-level symbolic representations are the emergent result of interactions among sub-symbolic elements
 - Marvin Minsky (1987): "Society of Mind" (society of mental agents work and compete to produce cognition)
 - Daniel Hillis (1987): "Connection Machine" (massively parallel computer: > 64000 processors)
 - Rodney Brooks (1991): "Subsumption architecture" (robots are controlled by collection of simple behavioral modules)
- Contemporary AI and ALife: many examples given in this course! ☺

NetLogo

Grew out of three ideas

- *Probing people's thinking*: How do people think about self-organizing behaviors?
- *Developing new conceptual tools*: Developing heuristics and qualitative tools to help people thinking about decentralized systems in new ways ("learning by building"; "to understand is to invent"; "constructionist approach to learning and education")
- *Developing new computational tools*: NetLogo is an massively parallel programming language that lets people control the actions of (and interactions among) thousands of computational objects

What is NetLogo? Modeling Complex Systems

Programmable modeling environment for simulating natural and social phenomena

- Well suited for modeling complex systems evolving over time
- Hundreds or thousands of independent agents operating concurrently
- Exploring the connection between the micro-level behavior of individuals and the macro-level patterns that emerge from the interaction of many individuals

What is NetLogo? Modeling Complex Systems

Easy-to-use application development environment

- Opening simulations and playing with them
- Creating custom models: quickly testing hypotheses about self-organized systems
- Models library: large collection of pre-written simulations in natural and social sciences that can be used and modified
- Simple scripting language
- User-friendly graphical interface

What is NetLogo? Flash History



LOGO (Papert & Minsky, 1967)

- Theory of education based on Piaget's constructionism ("hands-on" creating and test of concepts)
- Simple language derived from LISP
- Turtle graphics and exploration of "microworlds"

StarLogo (Resnick, 1991), MacStarLogo, StarLogoT

- Agent-based simulation language
- Exploring the behavior of decentralized systems through concurrent programming of 100s of turtles

NetLogo (Wilensky, 1999)

- Further extending StarLogo (continuous turtle coordinates, cross-platform, network, etc.)
- Most popular today (growing cooperative library models)

The World of NetLogo

NetLogo is a 2-D world made of 3 kinds of agents:

- Patches – make up the background or “landscape”
- Turtles – move around on top of the patches
- Observer – oversees everything going on in the world

Examples patch-only models



BZ-reaction



Fur

The World of NetLogo

Examples turtle-only models

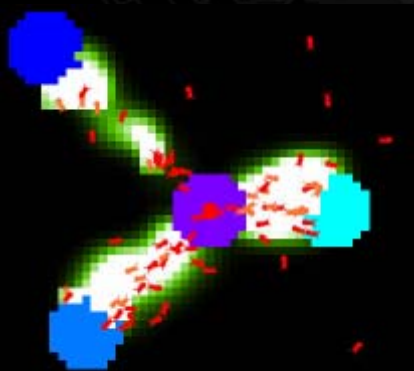


Flocking



Fireflies

Examples patch-&-turtle models



Ants



Termites

Sample session



```
> create-turtle 100
> forward 200
> forward random 200
> set-color blue
> if ypos < 0 [set-color green]
> if color = green [repeat 36 [forward 1 right 10]]
> turtles-have step-size
> set-step-size random 100
> forward step-size
> clear-all
```

Procedures

- executed by turtles
- executed by patches
- executed by observer (monitors overall activity of NetLogo world)



Observer

```
to setup
clear-all
create-turtle 100
patch-setup
end
```

Turtle

```
to walk
random-step
test-for-chemical
end
to random-step
right random 40
left random 40
forward 1
end
to test-for-chemical
if any-chemical here?
[set-color yellow]
end
to any-chemical-here?
ask patch-here [chemical>0]
end
```

Patch

```
to patch-setup
patches-have chemical
ifelse (distance 0 0) < 20
[set-chemical 10 set-patchcolor
blue]
[set-chemical 0]
end
```


To run NetLogo program:

```
> setup  
> repeat 1000 [walk]
```

Alternatively, use demon mechanism:

```
> activate-demon walk
```

Demon executes in the background

Multiple asynchronous demons are possible



Links:

<http://ccl.northwestern.edu/netlogo/>

http://doursat.free.fr/docs/CS790R_S05/CS790R_S05_Lecture_4_NetLogo.pdf

