Modeling Indirect Interaction in Open Computational Systems

David Keil and Dina Goldin University of Connecticut, USA

Workshop in Theory and Practice of Open Computational Systems WETICE '03, Linz, Austria, June 10, 2003

Our contribution

- Motivation: Design requires models
- Inspiration: Examples from natural systems
- We identify two distinct kinds of interaction
- *Indirect interaction* makes possible a richer set of behaviors in open computational systems than *direct interaction* alone
- *Models* that represent indirect interaction explicitly are more *expressive*

2

6

Keil-Goldin TAPOCS '03

Some interaction patterns in natural systems

- **1. Termites gathering chips into a pile** Protocol: Move at random, pick up chip when encountered, put down when another found
- **2.** Ants foraging for food forming trails Ants leave chemical trail, prefer existing trails, blaze shorter and shorter trails to and from food
- **3. Slime mold dividing and forming aggregate** These amoeba may aggregate by emitting chemical, migrating toward its greatest concentration

Keil-Goldin TAPOCS '03

What is the common feature?

In these systems, organisms communicate

- to coordinate their behavior
- in a decentralized way
- via their shared environment



Interactive computation

Ongoing exchange of data among computational entities, such that entities' outputs may causally influence their later inputs

- Open systems are interactive
- Interaction features interleaved, dynamically generated stream I/O (Goldin et al, 2001)
- Contrasts to *algorithmic computation*, such as that modeled by Turing machines, in which inputs are all *precomputed*

5

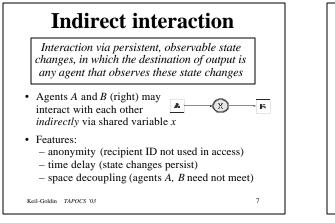
Keil-Goldin TAPOCS '03

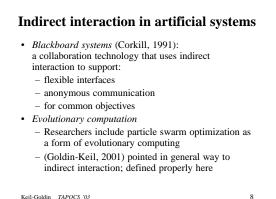
Direct interaction

Message passing, in which the destination is the agent specified in the message.

- This is the only kind of interaction modeled in concurrency theory
- Agent accesses to shared memory are represented by elevating shared memory to *process* status (Milner, 1993)

Keil-Goldin TAPOCS '03





Interaction via the real world

- Indirect interaction among computational entities introduces the possibility for the *real world* to be used as the interaction medium
- Examples:
 - Robot societies in which robots collaborate by moving physical objects
 - Sensor networks in which sensor motes move or leave markers

Keil-Goldin TAPOCS '03

Persistent state increases power

- Environment-agent relation features a *symmetry* in which each has persistent state
- Agents without persistent state (memory) are *reflex* (Russell-Norvig) and less adaptive, less powerful
- Indirect interaction uses the persistent state of the environment as a medium of communication

 It is what enables anonymity, time decoupling,
 - It is what enables anonymity, time decoupling, space decoupling
- · Formal modeling:
 - Semantically, persistent state is *data*, not a *process*
 - Persistent Turing Machines (Goldin et al, 2001) use persistent state to achieve greater expressiveness

Keil-Goldin TAPOCS '03

Self-organization and emergent behavior

- Persistent state enables a *wider range of behaviors*
- Self-organization: the interaction of a set of processes or structures at a lower level of a system to yield global structures or behavior at a higher level
- *Example:* Chemical reactions
- The higher-level system behavior is often called *emergent*

Keil-Goldin TAPOCS '03

11

Stigmergy

- *Definition:* A variety of self-organization in which agents are *mobile*, interacting via the environment
- Examples: from nature (slide 3)
- Stigmergy is an instance of indirect interaction that enables a wider behavior range due to emergence
- Design idea: Use stigmergy in design of collaboration technologies

Keil-Goldin TAPOCS '03

10

Theorem

Indirect interaction via the real world enables richer system behavior than is possible with direct interaction alone.

Proof: The real world may be assumed to be analog. Therefore its response to actions by computing agents may be uncomputable (Siegelmann).

Keil-Goldin TAPOCS '03

Power of indirect interaction

Corollary to above theorem: Models that explicitly represent indirect interaction, including via the real world, are more expressive than models that don't.

Conjecture: Even when modeling indirect interaction that does not use the real world as a medium, models that represent indirect interaction (via shared memory) explicitly are more expressive than those that don't.

14

Proof: future work

Keil-Goldin TAPOCS '03

Future work Prove the richness conjecture Define a formal semantics of indirect interaction Place collaboration technologies on a firm theoretical foundation by modeling indirect interaction explicitly

Keil-Goldin TAPOCS '03

15

13

Selected references Daniel D. Corkill. Blackboard systems. AI Expert 6 (9), pp. 40-47, 1991. Dina Q Goldin, David Keil. Evolution, interaction, and intelligence. Proceedings, Congress on Evolutionary Computation, Seoul, Korea, 2001. Dina Goldin, Scott Smolka, Peter Wegner. Turing Machines, Transition Systems, and Interaction. 8th Int'l Workshop on Expressiveness in Concurrency, Aarlborg, Denmark, August 2001. Robin Milner. Elements of interaction. Communications of the Association for Computing Machinery 36 (1), pp. 78-89, 1993. Stuart Russell, Peter Norvig. Artificial intelligence: A modern approach. Addison-Wesley, 1995. Hava Siegelmann. Neural networks and analog computation: Beyond the Turing limit. Birkhauser, 1999. Keil-Goldin TAPOCS '03 16