Reconciling Autonomic networking with agent-based modeling

Dimitri Papadimitriou
Alcatel-Lucent - Bell Labs
dimitri.papadimitriou@alcatel-lucent.com

IETF 94 (Yokohama) – NMRG meeting

Multi-agent system

- Assume cooperative environment

- Agent: function (map input to action)
  - **Autonomous agent**: capable of making decisions on behalf of users or other programs with some degree of independence – without external intervention/direction in response to situations it encounters

    -> hence, free to choose between different actions

    Note: also referred to as intelligent agent

  - **Learning agent**: capable of making decisions about how it acts based on experience (capable of improvement over time)

A first observation…

- Adaptivity : capacity to react and/or change decisions in a timely and cost-effective manner when internal or external events occur **that affects value delivery**

  => the same input doesn't produce the same output and thus decision(s) that would be detrimental to the system

  Characteristic property of Complex Adaptive System (vs. multi-agent systems)

- Often used in its weakest form: different input produces different output (decisions)

Agent-based modeling

- Agent-based model consists of
  - Set of agents, their (spatial, temporal, or functional) attributes and behaviours
  - Set of agents relationships and methods of interaction
    - Determine processes (rules) governing the interactions among agents
- Key properties
  - Modularity and self-containment
  - Offers capability to model potential asynchrony of (heterogeneous) interactions among agents and between agents and their environments
  - Autonomy: agents individually decide when, how and with what to interact

Agent-based modeling

- Agent-based model (ABM): computational models of complex adaptive systems
  - Primary goal: search for explanatory insight into the collective behavior of agents (which don't necessarily need to be "intelligent")

  + Ability of agents to be autonomous fits (a-priori) self-X principles
  - Hide real complexity to the “user” and handle more and more complexity on their own
- Technique to model decentralized optimization/decision problem solving with agents having individual goals and constraints, negotiate to reach a global satisfactory/ acceptable state
  - Fits policy framework (map state to action and enforcement)
  - Example: DCOP, ACO, particle swarm optimization

Autonomics

- Focused on self-optimization – self-configuration: ability of the system to automatically (and adaptively) configure/optimize local operation parameters/decisions according to global objectives

- Design pattern:
  - Operator/user specifies high-level objectives
  - Automatic translation of high-level objectives into low-level policies (state to actions) and configurations
  - System (autonomously) optimizes one or a number of goals

- Challenges:
  - Heterogeneity (node and network-level)
  - Distribution
  - Reliability
  - Complexity

Heterogeneity

- At first glance, software agents provide mean to “abstract” from underlying hardware, OS and provide engine-implementation specifics

- This benefit rapidly leads (when applied to network systems) to questions such as
  - Who becomes the master/deciding entity: the agent or the underlying engine(s) ?
  - Do they supplement or complement each other ?
  - When an agent performs on behalf of multiple engines shall it also orchestrate the decisions/actions (if these engines are inter-dependent) ?

Distribution

- Agent-based communication
  - Self-containment property of agents (modularity) enables to determine if something is part of an agent or not (boundaries)
  - Agents have easily identifiable attributes
  - Capability to model potential asynchrony of interactions among agents and between agents and their environments
- Alternatives: specific (1:1) or generic (N:1)
- There is no clear “insight” (agent-based modeling is not a system design but a modeling technique)

Reliability

- When agent act on behalf of
  - (How to) verify if agent decisions are applied correctly ?
  - How to determine that the “agent-based system” performs as expected ?
  - How to detect counter-acting/antagonistic decisions ?
  - How to identify events on which agent have to act on (or not) ? And timeliness of (expected) response ?

- Note the same questions apply also when modeling autonomic networking without ABM – the major difference stems from granularity (potentially many agents per node element vs. at best single autonomic manager per element)

Complexity

- The overall goal of autonomics is to ***hide/reduce complexity*** of network (micro-)management and related operations

- … but original goal of ABM is to (computationally) model emergence of complex behavior from (simple) behavioral rules specifying interactions executed on the basis of purely local information, without reference to the global pattern

  → thus, a good model ***produces complexity***

- Not trivial … may be clarification of what "hiding/reducing complexity" implies/means in CS-terms would help (does it refer to abstraction technique ?) and which complexity metric is being used

Reconciling models (tentative)

Autonomics

- Top-down (holistic)
- Functional
- Targeted (network control)

- No central authority (but some form of coordination)
- Procedure-driven (main task: specify the “autonomic loop(s)”), orchestration follows

Agent-based modeling

- Bottom-up
- Behavioral
- Multi-purpose/generic (model of technical and non-technical systems)
- Decentralized (cooperative but also non-cooperative)
- Interaction-driven (main task: specify interaction methods)

… if something would remain

- Agent-based model (ABM): computational models of complex adaptive systems
  - Primary goal: search for explanatory insight into the collective behavior of agents (which don't necessarily need to be "intelligent")

  + Ability of agents to be autonomous fits (a-priori) self-X principles
  - Hide real complexity to the “user” and handle more and more complexity on their own
- Technique to model decentralized optimization/decision problem solving with agents having individual goals and constraints, negotiate to reach a global satisfactory/ acceptable state
  - Fits policy framework (map state to action and enforcement)
  - Example: DCOP, ACO, particle swarm optimization

But…

- Only if some level of cooperation can be considered

  then autonomic management framework (and objectives) coupled to agent-based model leads to distributed version of constraint optimization problems as main research goal

- Otherwise, if selfish behavior (self-interested agents) considered as main characteristic of Internet environment

  then (non-cooperative) game theoretic models for distributed optimization would better fit this objective (but does it still fit the autonomic framework ?)