Reflection in Action:
Meta-Reasoning for Goal-Directed Autonomy

Ashok K. Goel

Design & Intelligence Laboratory, School of Interactive Computing,
Georgia Institute of Technology, Atlanta, Georgia 30308, USA
goel@cc.gatech.edu; http://home.cc.gatech.edu/dil/3

Abstract
In this talk, I will review my laboratory’s work on meta-
reasoning for goal-directed autonomy. We are exploring
how meta-reasoning in intelligent agents enables self-
adaptation and how self-adaptation promotes goal-directed
autonomy. In particular, we are investigating how an agent’s
model of its own design enables to the agent to adapt its
design to recover from failures (failure-driven retrospective
adaptation) and to achieve new goals (proactive, goal-
directed adaptation). I will illustrate these issues with
examples from the interactive turn-based strategy game
called Freeciv.

Meta-Reasoning for Goal-Directed Autonomy

Autonomy is a central characteristic of intelligent agents.
Since autonomous agents live in dynamic environments,
they must adapt themselves to changes in the world. It is
useful to make a few distinctions here. Firstly, adaptations
to an agent can be retrospective (i.e., when the agent fails
to achieve a goal in its given environment), or proactive
(i.e., when the agent is asked to operate in a new task
environment). Secondly, adaptations can be either to the
deliberative element in the agent architecture, or the
reactive element, or both. Thirdly, adaptations to the
deliberative element may be modifications to its reasoning
process (i.e., to its task structure, selection of methods, or
control of reasoning), or to its domain knowledge (i.e., the
content, representation and organization of its knowledge),
or both.

For the past two decades, my laboratory has been exploring
the thesis that an agent’s model of its own design may
enable the agent to adapt itself both to recover from failure
(failure-driven, retrospective adaptation) and to achieve
new goals (goal-directed, proactive adaptation). In
particular, we have been investigating the hypothesis that

an agent self-model that specifies the teleology of
the agent’s design (i.e., a model that describes how the agent’s
methods use knowledge to accomplish its tasks) affords
localization of the modifications needed for goal-directed
as well as failure-driven. We have developed an agent
description language called TMKL (Murdock & Goel
2008) for specifying the teleology of an agent’s design,
i.e., for specifying the goals and subgoals of the agent,
the domain knowledge of the agent, and the methods of the
agent that use domain knowledge and compose the
subgoals into the goals. TMKL is as expressive but more
explicit than HTN (Hoang, Lee-Urban & Munoz-Avila
2005).

In addition, we have developed a series of knowledge
systems that use agent self-models for enabling different
kinds of model-based adaptations in different conditions.
Autognostic (Strouf & Goel 1996, 1997, 1999), for
example, performs failure-driven, retrospective
reconfiguration of the reasoning strategies in the
deliberative component of a planning agent. Autognostic
actually is a knowledge-based shell in which different
agents can be encoded and adapted. Reflects (Goel et al.
1998) is an encoding of a reactive agent in Autognostic; it
modifies the design of a reactive agent to recover from a
class of failures. We found that agent self-models enable
self-monitoring of the agent’s behaviors as well as self-
diagnosis of the agent’s design when the agent’s behaviors
failed to achieve a goal.

REM (Murdock & Goel 200, 2003, 2008) performs goal-
directed proactive adaptation, for example adapting a
planning agent designed to assemble a device from its
components to achieve the new goal of disassembling the
device. We discovered that TMKL self-models enable an
agent to transfer and adapt its task-method structure for the
task it was designed to new similar, closely related tasks
assigned to it. REM too is a knowledge-based shell for
encoding different planning agents. It uses the TMKL
description of the planning agent to localize the
modifications needed to achieve new goals, and then uses
Reflection in Action

In recent work, my colleagues and I are investigating the above ideas in the context of the interactive turn-based strategy games. We know that game designs evolve rapidly but incrementally. Further, when a game evolves, the software agents that play the game should evolve as well. However, the current practice of adapting game-playing agents to incremental changes in game designs is ad hoc because at present there is little systemization of designs of game-playing agents or understanding of mechanisms for modifying their designs. We hypothesize that scaling up game-agent adaptation to end game-users requires interactive tools that make the designs of game-playing agents transparent to game users, help generate design modifications, and automatically translate modifications to agent designs into modifications to their codes. My software engineering colleague Spencer Rugaber and I are developing an interactive technology (called GAIA) for supporting adaptation of game-playing software agents as game designs evolve. GAIA contains a TMKL model of software agents that play Freeciv, an open-source variant of the turn-based strategy game called Civilization (www.freeciv.org). As the game design evolves (e.g., a new rule is introduced into the game), GAIA uses the TMK model to analyze the modifications needed to the agent design and propagates the changes all the way down to the program code (Jones et al. 2009). This project is enabling us to build on and pull together various elements of our previous work on meta-reasoning for self-adaptation and goal-directed autonomy.

Acknowledgements: My collaborators in this research over the years have included Eleni Stroulia, J. William Murdock and Joshua Jones. The ongoing GAIA project is in collaboration with Spencer Rugaber and Lee Martie, and supported by an NSF (SoD) Grant (#0613744) on “Teleological Reasoning in Adaptive Software Design.”

References


