Investigating the Value of Information and Computational Capabilities by Applying Genetic Programming to Supply Chain Management

Scott A. Moore[1] and Kurt Demaagd[2]

[1] University of Michigan Business School, Ann Arbor, MI, USA, samoore@umich.edu
[2] University of Michigan Business School, Ann Arbor, MI, USA, demaagdk@umich.edu

Abstract. In this paper we describe a research project centering on experiments in which game-playing evolving agents are used to investigate the value of information. Specifically, in these experiments we define populations of agents whose strategies evolve towards those that have better restocking strategies for their supply chain. The agents evolve their strategies in order to minimize costs (either for themselves or for their value chain). We describe several different experiments in which we will vary the abilities of agents both to gather and to store more information. Part of the results of this project will be related to the value of information and computational capabilities: Is it always better to have more information? If not, what are the conditions under which less information is better? The culminating experiment is one in which evolving agents compete to sell information to other evolving agents playing their roles in a supply chain.

1 Introduction

It is generally believed that having more information available to make a decision is a good thing; however, in today's computing-saturated business world information overload eventually occurs. Further, given that it costs money to manage information, care should be taken when the decision is made to gather an additional piece of information. The general strategy we are using in the research program described in this paper is to simulate a business problem, varying parameters so that players are placed under differing requirements for, or conditions of, information and computational capabilities. (For the rest of the paper, when it will not cause confusion, we will use the term "information" to refer to "information and computational capabilities.") The players who participate in these simulations will be created computationally through the use of a genetic program [Koz00]. Genetic programming is a computational approach to managing the evolution of agents whose fitness to live is based on their ability, relative to other agents in the

population, to perform in a specified way, for example to navigate a space or to compute a particular function. We are using genetic programming in this experiment as a means for investigating the information needs of the participants — the abilities both to gather and to make use of information — as they vary with the simulation's complexity. At the end of the series of experiments we will also look into how effectively evolving agents are able to sell information to the agents competing in the supply chain. A secondary result of this research program will be an increased understanding of how sensitive the genetic program is to its initial parameters, evidence showing how varying certain genetic program-related parameters affect the experimental results, and how well a genetic program can navigate a large search space. The specific business situation we will use in this research program is a multi-level supply chain game as exemplified by Sterman's beer game [Ste89]. Throughout the rest of the paper we use the term *beer game* to refer to the simulation we are investigating. A final result of this project will be insight into how well solutions found by a genetic program compare with solutions found by other methods in the management science and information systems literature. This paper describes the research program we are undertaking on this subject.

In the initial experiments described in this paper, agents evolve in several separate populations that correspond to the different roles in the beer game, namely retailer, wholesaler, distributor, and manufacturer. (See Figure 1.) The standard scenario is one in which an agent can only send orders to

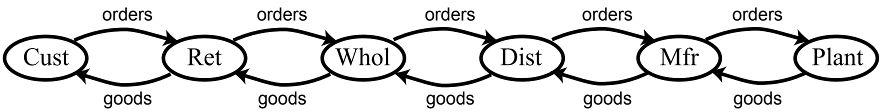

Fig. 1. Basic set-up of supply chain

one agent who is one step upstream (referring to the flow of goods) and can only send goods to one agent who is one step downstream. Demand is determined by an exogenous customer who both sends orders to the retailer and ultimately receives final delivery of the goods. The times for orders and goods to arrive at their destinations are taken from distributions that are not known by the players. The goods are manufactured at the plant, another exogenous player.

Each week each player has a standard series of tasks that it completes (see Figure 2). First, goods arrive from the player upstream; these goods are put into inventory. Second, orders arrive from the player downstream; these orders are added to any previous backorders (i.e., orders from past weeks that have not been filled). Third, the player ships as many goods as it can in order to fulfill the orders it has received. Fourth, the player places an order