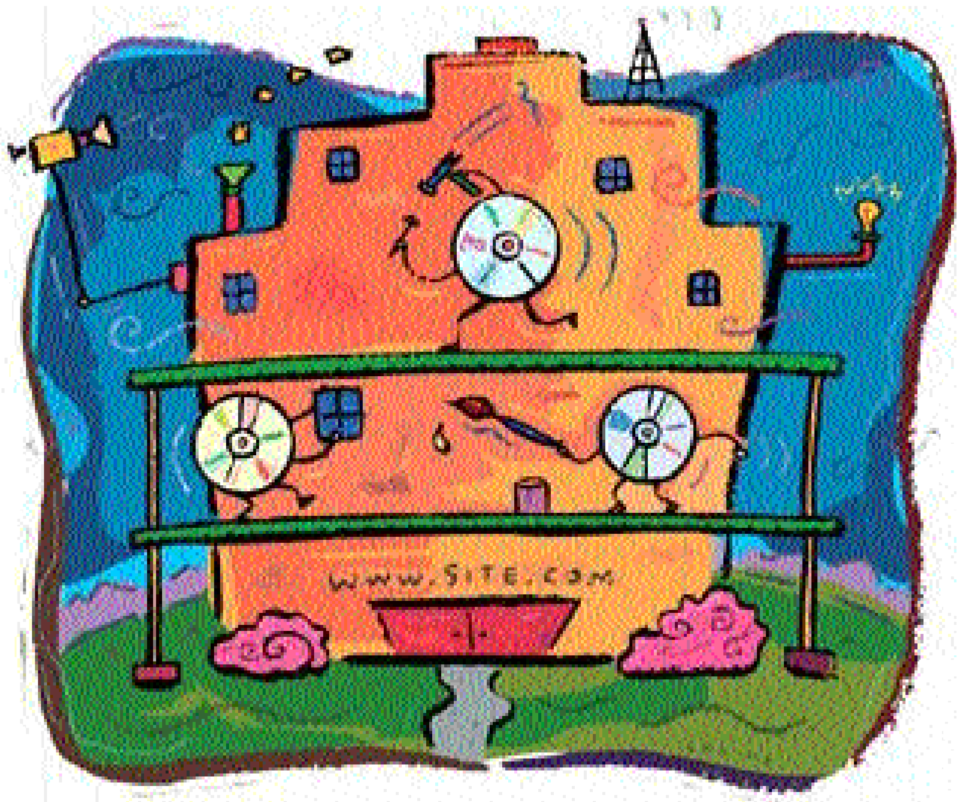
Agent-based Manufacturing

Part 2 of 2

Putting agents to work in the real world

David Greenstein & James J. Odell
dgreenst@tir.com / jodell@compuserve.com



Agent-based systems can borrow many concepts from such natural systems as ant colonies, wolf packs, and bird flocks. Even human systems, such as taxicab dispatching, auction houses, peace negotiations, and market economies, have highlighted parallels between natural systems and agent-based systems. In particular, manufacturing systems are one such area that can benefit from the metaphors of natural systems. When designed in this way, manufacturing systems become as flexible, dynamic, and adaptive as are systems in the natural world.

Adaptation in Natural Systems

Adaptation is no stranger to manufacturing operations. Manufacturers that fail to adapt rapidly to the ever-changing world become extinct. They go out of business. Adaptation enables the system to react to changes in the market or in the manufacturing environment. When designed properly, the individual parts of the system can be empowered to change based on their environment and market conditions.

An adaptive agent responds to its environment. The simplest form of adaptation is reaction—that is, a direct, predetermined response to a particular event or environmental signal usually expressed by an IF-THEN form. From atoms to ants, the reactive mode is quite evident. A carbon atom would have a rule that states in effect, "If I am alone, I will only bond with oxygen atoms." An ant would have a rule that if it finds food, it should return the food to its colony, while leaving a trail of pheromones. By themselves, reaction rules do not change. Instead, change comes through other mechanisms such as learning and evolution. Without learning and evolution, ants and atoms are still quite able to support complex "societies." *With* learning and evolution, however, the rules can be changed based on experience—resulting in new and perhaps improved results.

Learning

Learning is change that occurs during the lifetime of an agent and can take many forms. The most common techniques enable rules and decisions to be weighted based on positive or negative reinforcement. For example, in a basic bidding system, a bid could be selected simply on the basis of bid price. However, other considerations might also be appropriate, such as the bidder's ability to deliver its goods in the quantity, quality, and time frame requested. Over time, a Purchasing agent can learn to choose from reliable Vendor agents instead of just choosing the lowest bid. If a vendor's performance improves (or declines), the purchaser's decisions are modified accordingly. In other words, the agent continues to learn. Popular learning techniques that employ reinforcement learning include credit assignment, Bayesian and classifier rules, and neural networks.

Evolution

Evolution is change that occurs over successive generations of agents. For example, Cell agents[1] in the Agile Manufacturing Information System (AMIS)[2] continually evolve to address changing market and business needs. Here, the mix of resources within a Cell dynamically evolves and changes so the cell can produce the products demanded in the marketplace. Each Resource agent in a Cell must continue to win jobs and maintain a positive bottom line, thereby contributing to the overall profitability of the cell.

A resource that consistently fails to win jobs will eventually have a negative cash balance. If a resource maintains a negative cash balance long enough, the cell may decide to replace that resource. The nonproductive resource can either "die" due to malnutrition of cash or can be sold to another cell. The original cell can then buy a replacement resource, with different capabilities better suited to the products produced in the cell. In other words, there is a "survival of the fittest" quality to the mechanism, where each internal change represents a new generation of cell. By evolving in this way, the cell maintains a set of resources that allows it to remain profitable and survive in a dynamic marketplace.

Best Interest

Whether adaptation is by learning or evolution, each agent is responsible for acting in its own best interests to ensure that its goals are met.

The Cell agent's best interest is to win as many jobs as possible and keep the cell busy fulfilling customer orders. The cell also generates as much profit as possible, ensuring its continued viability as a virtual business enterprise.

The Resource agent's best interest is to win as many jobs as possible and keep busy processing jobs. The Resource also generates as much profit as possible, guaranteeing that it remains a viable member of the cell.

The Job agent's best interest is to complete the job quickly, to make certain that it is finished by the customer's due date. The Job agent also looks for the cell and resource that can complete the job at the lowest possible cost.

The best interest concept embodies the metaphor of free market behavior, as the cell, resource, and job interact and compromise to reach a solution that balances each agent's best interests. This balancing of best interests between these three entities and their dynamic interaction allows a dynamic, adaptive, and productive structure to emerge in agent-based manufacturing systems.

Seven-Step Negotiation Process

When you decide to purchase a product, your decision is influenced by certain requirements. For instance, the cost of the product must be within your budget. Another requirement might be how long it takes to perform the work. Once all the elements of your decision criteria are met, you award the work to the manufacturer that best fits your needs.

AMIS uses a standard seven-step bidding process to form an agreement to provide a product. This bidding process allows customers to obtain products through a common market process, ensuring that all products produced are at fair market prices. All seven steps of the negotiation process must be completed successfully in order to complete the transaction.

1. ***Request for quote***. The Request for Quote (RFQ) is the first step in the AMIS bidding process. A customer (or Customer agent) creates an RFQ that specifies the desired products or services, along with a response date, and sends it to a Broker agent. The Broker acts as a liaison, forwarding the RFQ to each Cell that has "subscribed" to provide the requested product.

2. ***Receive quotes.*** Each Cell agent determines its ability to complete the job according to the customer's RFQ specifications. If a cell is able to meet the customer's requirements, it creates a quote for the job. Quotes contain estimated information on the delivery date, price, quality, and special characteristics related to completing the job. Cells return their quotes to the broker, which holds the quotes until the RFQ response date has been reached and then returns all the quotes to the customer.
3. ***Select winner.*** When time has expired for the cells to submit quotes, the customer (or Customer agent) begins the selection process. The customer selects the winner from the submitted quotes by finding the most desirable mix of cost, time, quality, and special characteristics, based on its requirements for the job. The customer sends an award notification to the cell with the best quote.
4. ***Winner confirms.*** The winning cell accepts or rejects the job depending on whether it still has the capacity to do it. The cell might reject the job, if it has accepted other jobs between the times it prepared the quote and received the award. If the winner rejects the job, the customer offers it to the cell with the next best quote. This continues until a cell accepts the job. After the winner accepts the job, the other cells that submitted quotes are informed of the decision. At that time, losing cells are able to access data on the quotes, which help them evaluate why they lost the job—and perhaps learn by modifying their behavior for future quotes.
5. ***Issue purchase order.*** After a cell has confirmed the customer order, the customer authorizes the Cell to begin production by issuing a purchase order to the cell.
6. ***Generate product.*** The cell completes the work on the product, delivers the product to the customer, and sends an invoice to the customer.
7. ***Make payment.*** The customer ends the process by paying the cell for the work done.

The seven-step process establishes a common approach for business interaction between cells. The same process is followed when a cell wants to subcontract part of a job to another cell.

Agent Classifications

In a multiagent system, agents can play various roles. For instance in the examples above, an agent can play the role of job, cell, broker, and so on. Furthermore, each agent can play one or more roles. However, an agent with many roles can be difficult to manage and maintain. For example, an agent playing job, cell, and broker roles would be quite "fat." While there is no rule against fat agents even in the human world, we would find this too much for the average individual. Taking our lessons from nature, then, can be quite useful when designing agent systems. (After all, if nature has had a few billion years of solving hard combinatoric problems, maybe we could learn something from its results.)

In addition to roles, agents can be classified in other ways. The list below includes some of the major classifications employed by agent-based systems.

Atomic Agents

Atomic agents are the smallest and simplest agents. They are the building blocks upon which the entire system is built because they typically represent a single physical entity or function. Examples of atomic agents are Resource, Job, and Broker agents.

Aggregated Agents

These are collections of agents working together as one entity. They can contain atomic agents or even other aggregated agents. The Cell agent is an example of an aggregated agent because it consists of single and aggregated agents, such as Resource, Common Utility, and Negotiator agents.[2]

Cooperative Agents

These agents work together to serve a common set of goals. Typically, they must work together to function correctly. For example, the Job, Cell, and Broker agents must work together to support the negotiation process outlined above. If they do not, product development could not occur in any meaningful way, and agents such as the Cell would "die" because they could not make enough money on which to "live."

Intelligent and Nonintelligent Agents

Currently, no standard definition of intelligent and nonintelligent agents exists. However, it is probably safe to say that nonintelligent agents consist of a set of rules for their behavior and possibly DNA-like code (described below), though they do not contain a strategy or an evolutionary mechanism. It is, in effect, hard-wired into a certain strategy and state of evolution as set forth by the designer. In other words, the nonintelligent agent has no capacity to change in response to its environment or to make a noncodified decision.

Intelligent agents, on the other hand, can have mechanisms for evolution and strategy—along with a rule set and DNA-like code. The agent can therefore make intelligent decisions based on a best-interest strategy. It would adapt to its environment to make itself more profitable, efficient, or adaptable. The evolution mechanism allows the agent to modify its rule set and DNA-like codes according to its own strategy mechanism. The following elements define how the agent operates:

- Rule set- A set of rules and constraints under which the agent operates. It governs the agent's interaction with other agents and the environment.
- Goal set- A set of desired end states under which the agent prefers to operate. These are used to define the best interests under which the agent operates.
- DNA-like code- This represents the current state of the

evolution of an agent, including the physical capabilities of the agent, such as a cell or resource.
- Strategy mechanism- The tool which is used to pursue the best interest of the agent. When an agent must make a decision by weighing risks and rewards, it applies the strategy mechanism—including the best interest parameters of the rule set—to recommend a course of action, including modification of the rule set, if appropriate, based on the constraints imposed on the agent.
- Evolution mechanism- The evolution mechanism continually evaluates the behavior of the cell. Based on a behavioral evaluation, the evolution mechanism may modify cell behavior and capability by adjusting the DNA-like code of the agent. If a Darwinian-style evolution is employed, the most "fit" cells "live" and are crossbred; the least fit "die" and are removed from the system.

Summary

Distributed Organizational Control

To be agile, large centralized manufacturing organizations must be decomposed into simpler, smaller business units that are responsible for their own business, financial, and production success. This distributed organizational control allows these smaller units to reorganize and react quickly to changing market conditions. These smaller units—cells—can easily be reconfigured to maximize efficiency or to respond to a change in the market. Distributed organizational control enables the system to respond locally to unexpected failures or shutdowns by quickly reallocating the necessary resources.

Furthermore, distributed organizational control can be based on the concept of survival of the fittest. Therefore, if a cell within an organization consistently fails to contribute to the greater well-being of that organization, that cell ceases to exist. On the other hand, if every cell is successful, the entire operation is successful. Distributed organizational control allows a successful manufacturing operation to emerge from the interaction of smaller units.

Another benefit of a distributed organization is the ability to quickly form *ad hoc* formations of business units that achieve common business goals. Here, individual cells cooperate as a unit for a common benefit, and then dissolve when no longer needed. The components of such a virtual organization do not have to be aligned with a physical organization, adding another degree of flexibility not found in traditional systems.

Capacity Management

The more unpredictable the manufacturing environment, the more significant the problems associated with advanced scheduling. For that reason, AMIS does not use the concept of scheduling. Instead, it manages the capacity of the resources.

As a business entity, each cell has limited resources that have limited capacity. All jobs in a cell are temporarily put into a holding capacity queue of a cell. Then, just before a job starts, each resource in a cell bids on the job in the queue of the cell. Because the bidding is done right before the job starts, the chance of an unexpected event affecting the completion of the job is significantly reduced.

If, however, a problem occurs during the production process, the system is not disabled. This is an important benefit of capacity management. Because the resources in the cell are self-loading and balance the load among themselves, a job that cannot be completed by a resource is returned to the cell's queue for rebidding and reallocation. This dynamic allocation of jobs to resources greatly reduces the effects of the unpredictable nature of the shop floor. While this is not the only technique for capacity management, it works well in the automotive industry.

Market-Driven Economy

In a market-driven economy, manufacturers build products in *response* to market demand, rather than in *anticipation* of demand. Businesses compete for limited resources and customers but cooperate when it is beneficial. Change is constant as new products emerge and customer demands evolve.

AMIS relies heavily on the economic laws of supply and demand. Rather than trying to forecast market demand and schedule production based on rigid plans, AMIS provides an architecture that adapts to the dynamic marketplace. Both inside and outside the cell, agents operate in a profit-driven economy. The competition between cells or resources will drive the market to an equilibrium or market-clearing price. The producer's need for higher profit and faster production times interact with the customer's need for lower prices and higher quality. These opposing forces result in the best prices and products for everyone involved.

Conclusion

Agent-based manufacturing is a new way of thinking about and applying information. The primary benefits of the agent-based approach are that they provide dynamic, reliable, and agile systems. As such, it will enable organizations of the future to accommodate rapidly changing business conditions, increase market responsiveness, lower cycle times, increase productivity, and better utilize its resources—and most importantly, it will benefit the bottom line. In other words, the agent-based approach will be the way modern manufacturers develop their systems to compete in the twenty-first century.

References

1) Odell, James, and David Greenstein, "Agent-Based Manufacturing: Part I," Distributed Computing, 2(4), May, 1999.

2) Greenstein, David, and Kelly Thomas, "Intelligent Agents for an Emergent Industrial Ecology," *Intelligent Manufacturing Systems*, Proceedings of IJCAI, 1995.

David Greenstein is a project manager in the Manufacturing Information Systems department at GM.

James Odell <<www.jamesodell.com>> is a consultant, educator, and author.