Proxy-Bidding Strategies for Intelligent Agent Negotiations

Hitesh Attri **Amol Goel**
F. Frank Chen **Subhash C. Sarin**

Center for High Performance Manufacturing, Virginia Polytechnic Institute and State University
{hattri,ffchen,amgoel,sarins}@vt.edu

Keywords: Intelligent Agent, Multi-Agent System, Distributed Artificial Intelligence, Proxy Bidding, Negotiation Protocol

Abstract
Software Agents provide an efficient way to model complex interactions between intelligent entities present in a system. The contract net protocol based negotiation system coupled with intelligent agents is becoming a popular way to model complex and dynamic problem domain of manufacturing planning and scheduling. These systems are usually based on the free-market concept of economics and depend on the emergent behavior of the components of the system in order for them to act efficiently in diverse situations. In this paper, we propose two new bidding schemes which can be used by a Facilitator agent that controls such auctions (negotiations) between intelligent agents, to proxy-bid for an agent depending on its degree of risk acceptance behavior. Proxy-bidding is desirable in intelligent systems to control network traffic and unnecessary client computations. Simulation based results show that a system with built-in proxy bidding is more robust and efficient when compared with a bidding system without it. We also provide an example where this bidding system is used as real-time job scheduler for an automated Job-Shop system.

INTRODUCTION
According to [Woolridge 2001], an agent is a computer system placed in an environment, which is capable of autonomous action in the environment to reach its goal. Most software agents used in contemporary agent–based systems loosely follow this definition. Agent–based systems have been gaining popularity for modeling of real – world problems in various domains which are complex and distributed and can be solved more efficiently by a set of intelligent autonomous entities rather than one big monolithic solver.

In the last decade there have been a number of agent-based applications developed for manufacturing systems control, planning and scheduling. [Shen et al 2000] reviews some of the research projects in the area. Many of these applications use the Contract Net Protocol [Smith 1980, Smith and Davis 1981] or its variants for allocation of resources to tasks. Market-based approaches which involve buying/selling of services between agents are used in almost all new multi-agent systems. [Clearwater 1996] discusses various projects where Market-based approach is used in distributed applications. While we now have some very flexible and robust systems available, most of these systems do not optimize the use of backbone networking. The cost of sending and receiving messages between agents is assumed to be zero or negligible. In this paper we propose a mechanism called proxy-bidding which reduces network traffic in terms of messages sent between agents, likewise it reduces computation involved in handling of these messages at the receiving agent.

This paper is structured as follows. In section 2 we discuss proxy-bidding, its application in an agent – based system and agent risk profiles. In section 3 we discuss the implementation of an agent – based real-time scheduler for an automated job – shop. We ran the scheduler without proxy – bidding and with it using a simulation model. In section 4 we tabulate the results obtained in both cases and interpret them. Finally, in section 5 we discuss conclusions which can be drawn from this study and future work.

PROXY-BIDDING
In a contract net based multi-agent system, we usually have proposing agents which propose new tasks in the system, an auction coordinator agent which posts these tasks on a publicly accessible location and accepts bids to carry out these tasks and bidding agents which bid for tasks of their interest. The agents communicate with each other by sending messages which have a standard format and use some network bandwidth for their transfer based on actual size. In an auction based system, a lot of messages are exchanged between the auction coordinator and bidding agents. Whenever a particular bidding agent is outbid by another agent it has to be communicated of the change in status by the auction coordinator, and then bidding agent may send a new bid for the task. Proxy bidding is the mechanism in which the auction coordinator bids on behalf of the bidding agents. This saves the transfer of messages between auction coordinator and bidding agent every time the bidding agent is outbid. This technique has been successfully applied by online auction stores like eBay

[eBay] for human bidders. The main difference between eBay's approach and ours is that eBay proxy-bidder assumes a single risk taking profile for all users which is not generally true. In our system, the coordinator models the bidding behavior of individual agents based on their risk – taking profile and automatically bids on behalf of the agent. There are two behaviors that a bidding agent can adopt depending on its level of *risk desirability*. For subsequent discussion we assume that the higher bid amount is favorable to the bidding agent (lower bid amount favorable is analogous). If the agent's behavior is more risk taking i.e. it is *risk neutral*, it can end up receiving a lower amount for that task. The objective of the agent in this case is to win the bid at any cost which is greater than or equal to its minimum acceptable bid amount. On the other hand for an agent showing *risk averse* behavior, the objective is to win the bid at highest amount possible even if there is a possibility that it might not win the bid. The bidding agent informs the auction coordinator what kind of proxy bidder (risk neutral or risk averse) it should create and the minimum amount it is willing to receive to carry out that task.

Risk – Averse Agent Profile

The risk averse proxy bidder fits a straight line between the starting bid and the minimum amount the agent is willing to receive. If while the auction is running, the bid amount goes below the minimum that was specified by the bidding agent, the proxy bidder asks the agent if would like to lower its minimum. If the bidding agent does lower its minimum then the process starts again else it withdraws from the auction.

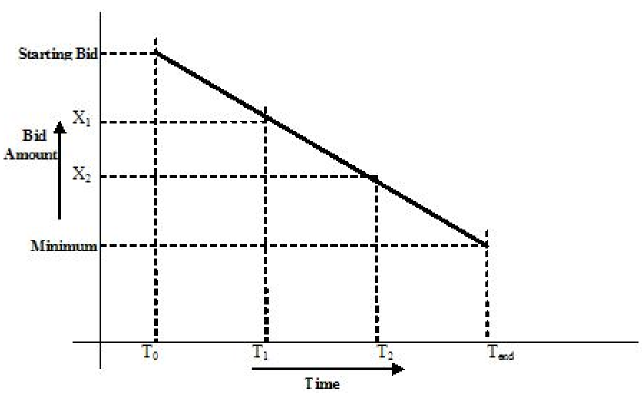

Figure 1. Risk Averse Proxy Bidder

The algorithm for risk averse proxy bidder:

```
At each time interval
 if (my.current = best bid)
     do nothing
 else if(best bid < my.interval.bid)
     my.current = my.interval.bid
 else
     do nothing
```

Risk – Neutral Agent Profile

The risk neutral proxy bidder calculates the starting bid amount of the agent it is representing based on the minimum value given by the agent. At the penultimate time interval of the auction it checks if its current bid is the best bid, if it is then it does nothing else it bids the minimum bid amount as its current bid. If while the auction is running, the bid amount goes below the minimum that was specified by the bidding agent, the proxy bidder asks the agent if would like to lower its minimum. If the bidding agent does lower its minimum then the process starts again else it withdraws from the auction.

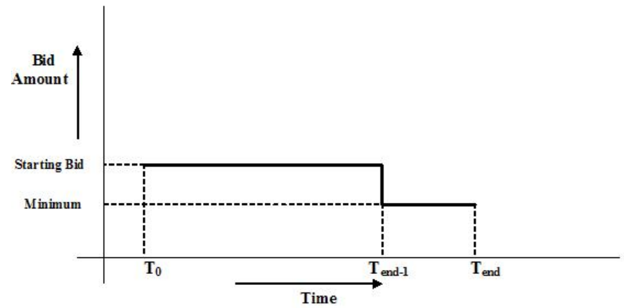

Figure 2. Risk Neutral Proxy Bidder

The algorithm for risk neutral proxy bidder:

```
      At penultimate time interval
            if (my.current = best bid)
                  do nothing
            else
                  my.current =
            my.minimum
```

IMPLEMENTATION

We use an automated job shop to compare the performance of an agent-based system with proxy bidding mechanism and without it.

The Job Shop

We considered a hypothetical job-shop which closely resembles the FMS Lab setup at Virginia Tech. The job shop consists of new parts arriving through a Conveyor (In), they are then picked up by a Robot (1) and placed on any of three identical Machining centers, M1, M2 and M3. The robot also moves the parts between the machines depending on their processing requirements. Once all operations for the part are over, another Robot (2) picks the part and places it on a Conveyor (Out) and the part exits the system. The parts have pre-determined due dates and a set of operations that have to be performed on them. All operations can be performed on any of the three machines but processing time required on each machine for the same operation is different. All processing times are deterministic and known to machines. There are two types of parts which enter the system – normal and rush orders. Rush orders have higher priority than normal parts. We also assume that there are no queues in the system.

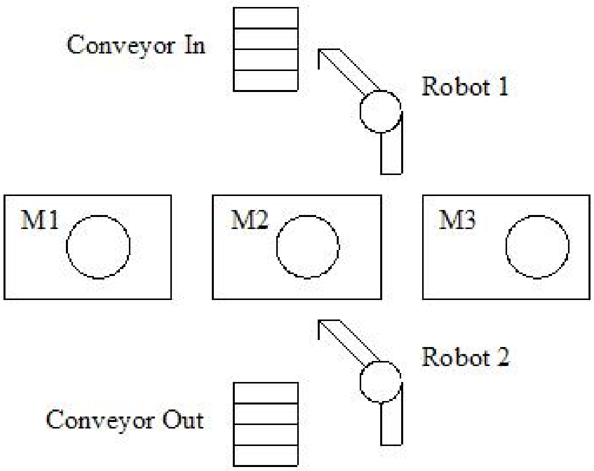

Figure 3. The automated Job-shop

Agent – Based Model

We model the above job-shop using intelligent agents. We use three different types of agents – Part agents, Facilitator agent and Machine agents. We assume robots and conveyors are always available when required and they just add a constant delay to a part's processing time. As we don't have to schedule operations on these components of the job-shop we exclude them from the agent-based model. Part agents bring new tasks to the system when they arrive, the machine agents bid for these tasks based on their availability and value of particular operation to the machine agent. The Facilitator agent supervises the auction process and awards processing contracts to winning machine agents. Parts start with a fixed budget to be used towards paying the machines that would perform the required operation(s) on the part. When a new part enters the system, it sends a message to the Facilitator agent announcing its arrival and type of operations needed by it and its deadline. The Facilitator agent creates a new blackboard (publicly accessible object) for the part and announces creation of blackboard to all known machine agents. The bidding process is initiated and the bidder with the lowest bid at the end of the auction time is awarded the contract. Once a machine has been awarded a contract to carry out an operation, the next operation of the part (if any) becomes available for bidding. When a machine is working on a part it would bid for new parts entering the system only if it would be able to finish the new operation before its due date, after finishing the operation it is currently doing. Once all operations for a part are finished, the Facilitator agent destroys the associated blackboard object, the part agent dies and the part is assumed to have exited the system.

In normal agent-based system, the Facilitator agent informs currently winning agent each time they are outbid by some other agent. The agent then re-submits a bid by lowering its original bid by a random amount (but still greater than or equal to its minimum valuation for the deal). In the proxy-bidding system, the agent when submitting their bid for a task inform the Facilitator agent the risk profile they wish to use and their minimum valuation for the task. The Facilitator agent contacts the Machine agent only when it wins the contract or bid amount goes lower than its minimum valuation while auction is active. The activity diagrams of various agents are as follows:

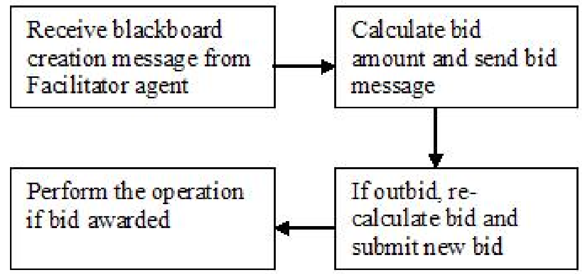

Figure 4a. Activity diagram for Machine agent without proxy-bidding

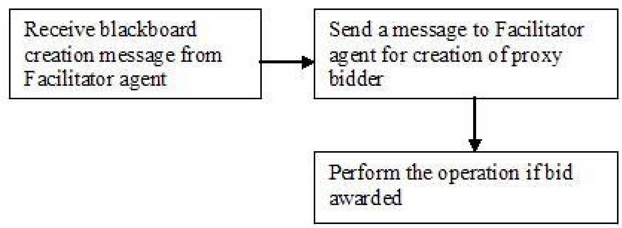

Figure 4b. Activity diagram for Machine agent with proxy-bidding

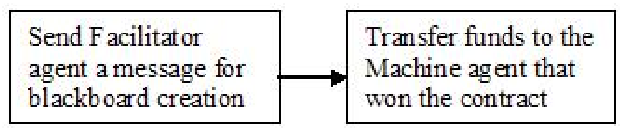

Figure 5. Activity diagram for Part agent

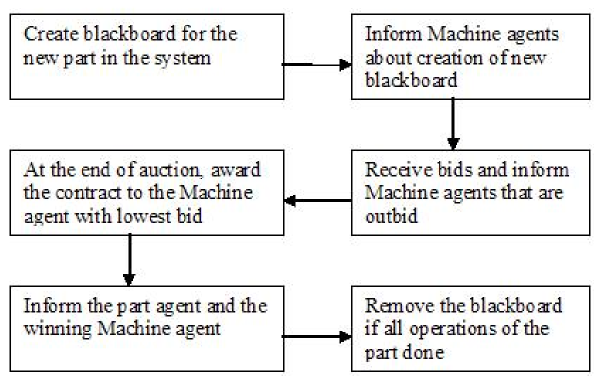

Figure 6a. Activity diagram for Facilitator agent without proxy-bidding

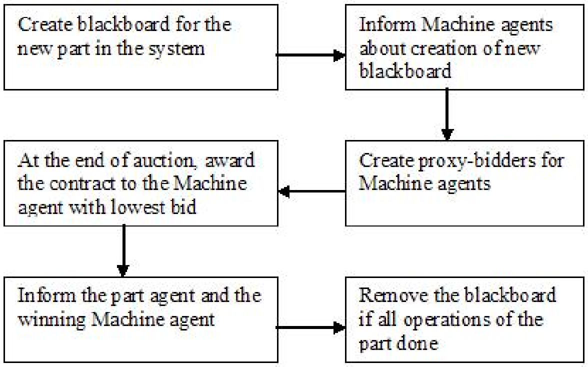

Figure 6b. Activity diagram for Facilitator agent with proxy-bidding

The Simulation Model

The word-wide body for agent standards is FIPA (Foundation for Intelligent Physical Agents) [FIPA]. It is a non-profit organization which produces software standards for interoperable agent based systems. Most contemporary agent – based systems are FIPA compliant. JADE (Java Agent Development Environment) is a software toolkit [JADE] which can be used for creation of FIPA compliant agents. It has been successfully used for development of agent based systems in prior research projects [JADE]. We built our agent system using JADE as the development environment. The agents communicate using FIPA-compliant messages encoded in XML format [W3C]. We ran both normal and proxy-bidding versions of our agents on JADE and noted down values of relevant parameters. The results which we obtained are summarized in the next section.

RESULTS

We assumed Poisson arrival process for parts entering the system. Parts were assumed to be of three different types having different operation sequences. We ran the system for eight different scenarios based on number and type of proxy – bidding agents present. Results obtained for two different data sets are summarized in Table 1 and Table 2 in Appendix. Figure 7a and Figure 7b show the average number of messages in the system for the two data sets. Figure 8 shows the profile of percent earning for Machine 1 for different scenarios for the two data sets.

CONCLUSIONS AND FUTURE WORK

It is shown that the number of messages reduce in a system with introduction of proxy bidders. The reduction is not linearly related with the number of proxy – bidders. On average the payoff for bidding agents remains same and does not depend whether they use proxy – bidding mechanism or not. Thus introduction of proxy bidding in an agent based resource allocation system would improve system performance in terms of efficiency of network bandwidth utilization without adversely affecting earning potential of individual agents. This study just introduces the concept of proxy – bidding as applied to multiagent systems. Further analysis is required for determining the potential pros and cons of applying proxy-bidding in larger scale system and some other problem domains. Some suggestions for modifying the current system to make it more complex and closer to real world situations are

- Non-reliable machine centers.
- Stochastic part processing times.
- Finite capacity server queues.
- More resource types modeled into the system (e.g. material handling components, tools etc.)
- Preemption of tasks on servers.

For making proxy – bidding more applicable to real world situations, we need to model more agent risk profiles to capture actual behavior of diverse autonomous agents in a large multiagent system.

REFERENCES

[Clearwater 1996] Ed. Clearwater, Scott H. 1996. *Market – Based Control, A paradigm for distributed resource allocation*. World Scientific Publishing Ltd., Singapore.

[eBay] The eBay online store http://www.ebay.com

[FIPA] The FIPA web-site http://www.fipa.org

[JADE] The JADE web-site http://jade.cselt.it

[Shen et al 2000] Shen, Weiming;Norrie, Douglas H.;Barthes, Jean-Paul A. 2000. *Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing*. Taylor & Francis, England.

[Smith 1980] Smith, Reid G. 1980. “The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver”. *IEEE Trans. on Computers* C-29(12):1104-1113.

[Smith and Davis 1981] Smith, Reid G.; Davis, Randall. 1981. “Frameworks for Cooperation in Distributed Problem Solving”. *IEEE Trans. on Systems, Man, and Cybernetics* 11(1):61-70.

[W3C] The W3C XML web-site http://www.w3.org/XML

[Woolridge 2001] Woolridge, Michael. 2001. *An introduction to Multi Agent Systems*. John Wiley and sons Ltd., England.

APPENDIX

Table 1. System performance for different scenarios (Data Set I)

Scenario	Data Set 1						
	Number of Messages	Machine 1		Machine 2		Machine 3	
		Operations	Earnings	Operations	Earnings	Operations	Earnings
All NP	2480	7.5	14896	8	15891	8	15922
1 A , 2 NP	927	8.5	10524	9.5	11649	7	11785.5
1 N , 2 NP	863	6	10240	10.5	9658	8.5	11728
2 A , 1 NP	735	10	5995	8	11118	5	7150
2 N , 1 NP	860	9	5348	10	4363	4	6930
1 A , 1 N , 1 N	834	9.5	8357.5	8.5	5695	5.5	9590
2 A , 1 N	262	3	9101.5	6.5	8788	10.5	7704
2 N , 1 A	233	7.5	7084	6.5	5189.5	3	5679
* NP : Non-Proxy Bidders , A : Risk - Averse Bidder , N : Risk - Neutral Bidder							

Table 2. System performance for different scenarios (Data Set II)

Scenario	Data Set 2						
	Number of Messages	Machine 1		Machine 2		Machine 3	
		Operations	Earnings	Operations	Earnings	Operations	Earnings
All NP	1908	6.5	12874	4	7976	4.5	8928
1 A , 2 NP	861	5.5	5912.5	7.5	8680	5	8960
1 N , 2 NP	1350	7	7818	5.5	7083.5	6	9391
2 A , 1 NP	387	7	9947	5	6892	6	9236
2 N , 1 NP	382	7	7833	6	6762	5.5	9765
1 A , 1 N , 1 N	551	7.5	8754.5	6	8498	4	7210
2 A , 1 N	260	6	6777	6	4497.5	7	5733
2 N , 1 A	273	7	6657.5	5.5	3915.5	5.5	6195.5
* NP : Non-Proxy Bidders , A : Risk - Averse Bidder , N : Risk - Neutral Bidder							

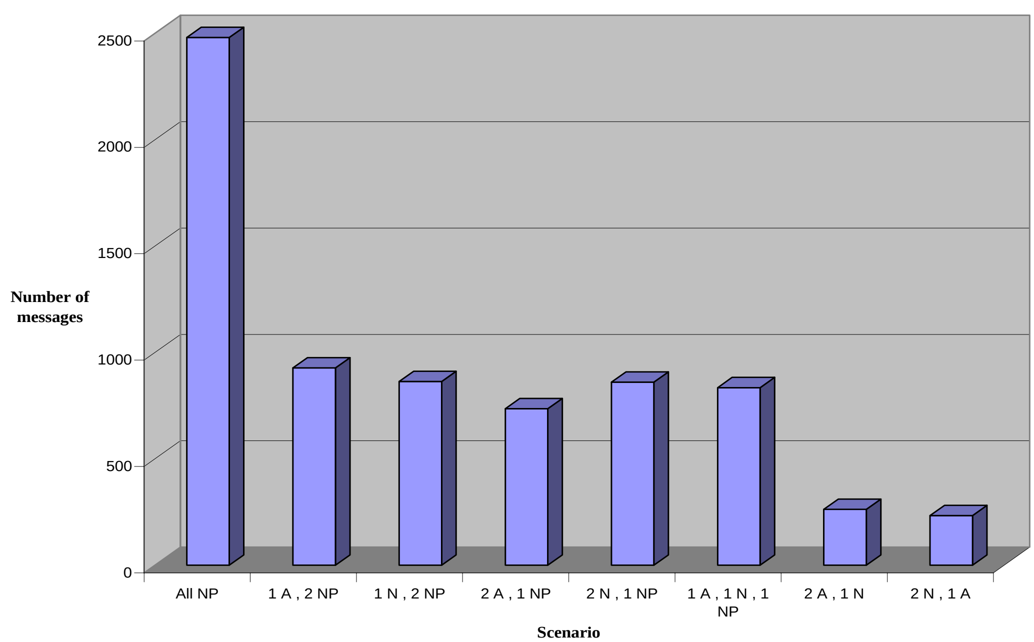

Figure 7a Average number of messages in system (Data Set I)

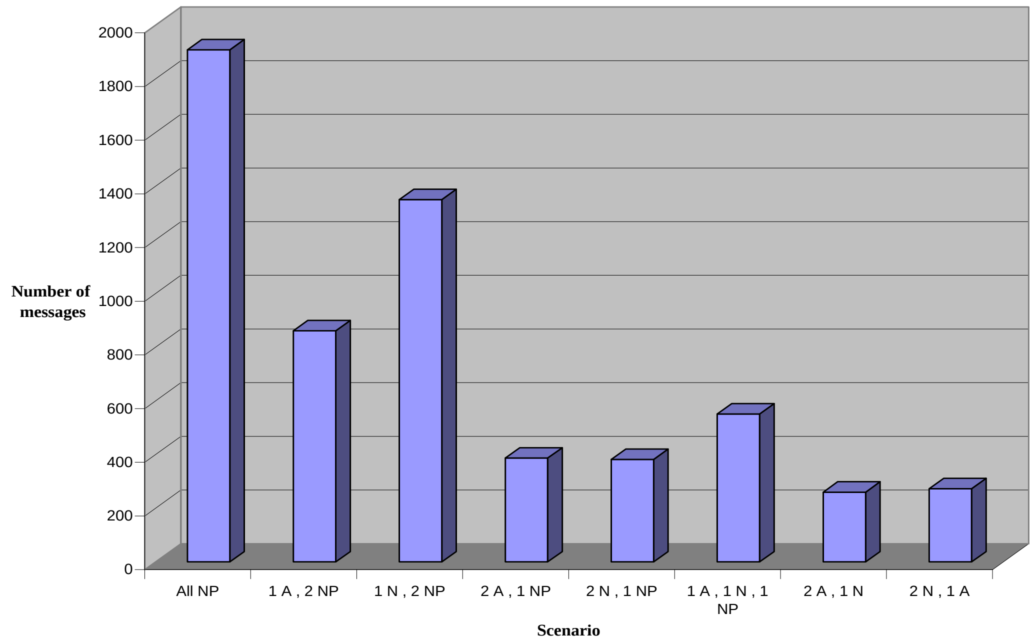

Figure 7b Average number of messages in system (Data Set II)

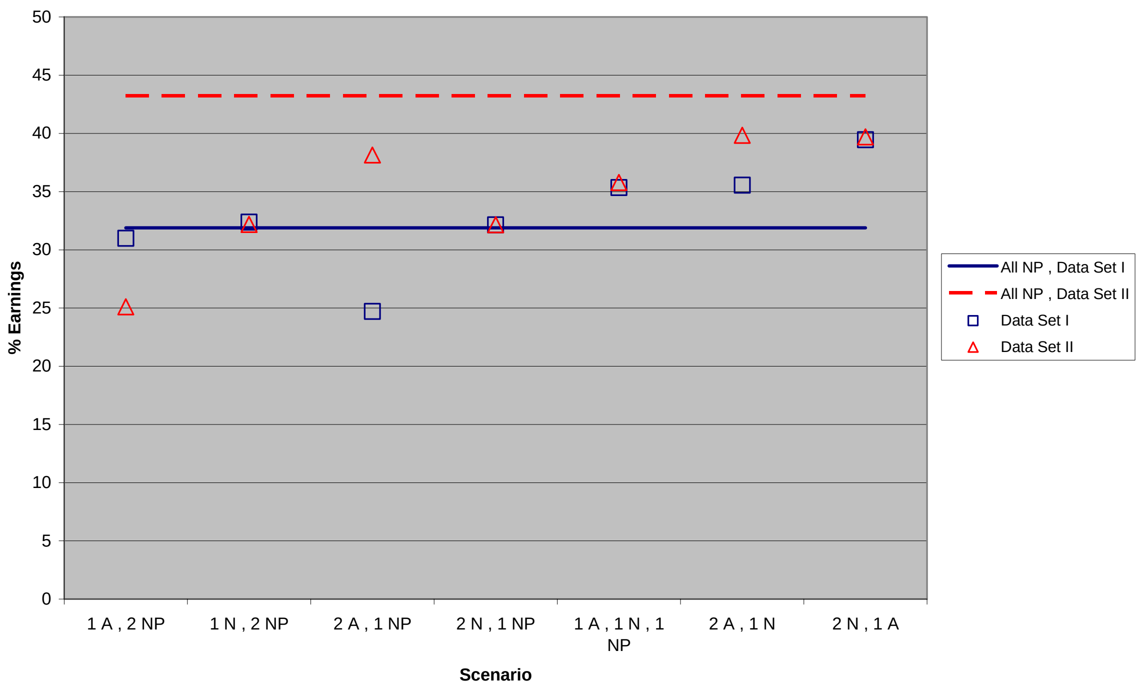

Figure 8. Earnings of Machine 1