Making Business Decisions with AI

John Reinstra
jreinstr@stanford.edu

Abstract: Critical business decisions such as pricing and budget allocation have long been relegated to humans. In this paper, we show that a basic AI can make reasonable strategic business decisions too. Using concepts from neoclassical economics, we build a basic economic simulation expressed as an MDP, and build an agent to use Monte Carlo Tree Search to determine the best action to maximize profit. Given this, our agent is able to make sensible business decisions.

I. Introduction

Building an AI to actually be the CEO of a company would be enormously difficult. To make this problem more feasible as a final project, we simplify the problem substantially. While an actual business leader would need to manage people and make decisions across many areas of the business, we simplify our problem to three key areas: product pricing, production capacity, and R&D spending. Our agent can choose to increase or decrease any of these things. We also simplify the timeframe: each day our agent can choose one action before proceeding to the next day. Our agent attempts to choose the action that results in the highest expected profit.

We model this problem as an MDP, because the agent knows the state of its business with certainty. Each state represents one day of operation, and includes details on cash balance, unit price, production capacity, R&D spending, unit cost, and total units sold on that day. Units sold is determined by a monopolistic competition demand function and takes price as an input. Reward is the profit, meaning the difference in cash balance from the previous state to the current state.

II. Related Work

There was unfortunately not too much we could find in the area of fully automated business decisions. However, we are certainly not the first to apply concepts of decision making under uncertainty to business decisions. In one example, Rajenda Srivastava and Liping Liu examine the use of belief functions in business decisions, specifically to aid in the auditing process. Auditors can use a belief function to assess whether a certain financial statement should be examined more closely [1]. Taking a data driven approach to making business decisions is an increasingly common practice. This can be seen in the book Quantitative Methods for Business Decisions, which describes in detail how to use statistics, simulations and game theory to aid in making decisions [2].

A key distinction must be made between these related approaches and our proposed approach. In both of these approaches, a human is viewing the data and making the final decision. In our approach, our AI agent views the data, weighs different actions, and makes the final decision without human input.

III. Approach

Below, we go more in detail in modeling the problem as an MDP and our approach to choosing the optimal actions.

A. Economic Simulation as an MDP

We model our economic simulation as an MDP. We choose to go with an MDP because the agent knows the state of its own business (pricing, budget, etc) with certainty, so the observation is identical to the actual underlying state. The other reason we choose to model the simulation as an MDP is because there are known online methods that work well in practice for determining optimal actions of MDPs. Below, we outline the details of our simulation.

Definition of State and Reward

Each state in our MDP contains the following information:

- Cash balance
- Unit cost
- Unit price
- Production capacity
- R&D spend
- Units sold today

Based on the unit price, we use the following demand function to determine the number of units sold:

Demand = 30.0 - (Unit price / 30.0)
Units sold = max(Production capacity, Demand)

This is a very simple demand function but it is based on the theory of monopolistic competition, where the products are partially differentiated from each other rather than being perfect substitutes. Monopolistic competition is considered an appropriate approximate model for a number of industries, including clothing, restaurants and services [3].

We determine the unit cost by looking at the previous unit cost and, with a random chance, reduce the unit cost by an amount based on R&D spend. This is a rough approximation of the “innovation” that can come from R&D.

Once we have determined the number of units sold, we calculate the cash balance and reward using the previous state:

Cash balance = Previous cash balance - R&D spend + Units sold * (Unit price - Unit cost)
Reward = Profit = Cash balance - Previous cash balance

This calculation subtracts total revenue (units sold * unit price) from total costs (R&D spend + units sold * unit price) and adds the previous cash balance to get the new cash balance.

Defining Actions

At any state, we allow the agent to choose from the following seven actions:

- Do nothing
- Increase price
- Increase production capacity
- Increase R&D spend
- Decrease price
- Decrease production capacity
- Decrease R&D spend

For any decrease action, we do a no-op if the value is already at zero. Otherwise, for any increase OR decrease action, we increment by either a fixed constant value or a percentage, whichever is greater. We also set the cost of increasing the capacity at $10 per unit, and the payoff of reducing capacity at $10 per unit.

B. Choosing Optimal Actions

We use Monte Carlo Tree Search in conjunction with our MDP simulation to choose optimal actions. Part of the reason for choosing this method is that it allows us avoid fully exploring the tree (which has a branching factor of seven) to our desired depth. We carry over counts and utilities from step to step and use a rollout policy of random actions, because we do not have expert knowledge in this field. We highlight one interesting note on the configuration of MCTS below.

Discount Factor and Time Value of Money

The MCTS discount factor affects the present value of future rewards. While this parameter can often be thought of as simply a lever to improve the performance of the agent, it takes on special significance in an economic context. The discount factor, in our case, acts as a measure of the "time value of money," which is the concept that money today is worth more than the same amount of money in the future, because money today can be profitably invested during that time [4]. When we tweak this discount factor for our economic agent, we are effectively tweaking the agent's view of how much long term profits are worth compared with short term profits, which can affect the agent's propensity for making good short term decisions vs. good long term decisions.

IV. Results and Analysis

When we run our MCTS agent in our MDP simulation, we start with a state where values are randomly initialized within the following ranges:

Balance Range:	$1000 to $5000
Unit Cost Range:	$1 to $5
Price Range:	Cost - $0.50 to Cost + $2.00
R&D Spend Range:	$0 to $5
Capacity Range:	5 to 15 units

We run our MCTS with the following parameters:

Discount factor:	0.98
Explore constant:	1.0
Depth:	10
Iterations:	2,000

Under these parameters, our agent consistently follows a popular approach taken by private equity firms and publicly traded companies: raise prices as much as possible and cut costs as much as possible. It does this by first raising prices until it hits the optimal revenue maximizing point. Then, it cuts the R&D budget, boosting short term profits and failing to realize, like many business leaders, that there is a long term benefit to R&D that does not initially exceed the upfront cost of R&D. The agent's strategy is a successful approach that usually more than doubles the business's profits. Below are some sample runs with these parameters.

```
chose: price_increase
        profit: 1087.0
chose: price_increase
        profit: 1780.0
chose: rnd_decrease
        profit: 1830.0
chose: price_increase
        profit: 1997.0
chose: price_decrease
        profit: 1713.0
chose: price_increase
        profit: 2194.0
chose: rnd_decrease
        profit: 2244.0
chose: rnd_decrease
        profit: 2294.0
chose: rnd_decrease
        profit: 2344.0
chose: rnd_increase
        profit: 2294.0
chose: rnd_increase
        profit: 2244.0
chose: nothing
        profit: 2244.0
chose: nothing
        profit: 2244.0
chose: nothing
        profit: 2244.0
chose: rnd_increase
        profit: 2194.0
chose: nothing
        profit: 2194.0
```

```
chose: price_increase
        profit: 417.0
chose: price_increase
        profit: 1020.0
chose: price_increase
        profit: 1713.0
chose: price_increase
        profit: 1929.0
chose: nothing
        profit: 1929.0
chose: capacity_increase
        profit: 929.0
chose: rnd_decrease
        profit: 1979.0
chose: nothing
        profit: 1979.0
chose: nothing
        profit: 1979.0
chose: rnd_increase
        profit: 1929.0
chose: nothing
        profit: 1929.0
chose: rnd_decrease
        profit: 1979.0
chose: rnd_decrease
        profit: 2016.0
chose: rnd_decrease
        profit: 2016.0
chose: rnd_decrease
        profit: 2016.0
chose: nothing
        profit: 2016.0
```

```
chose: price_increase
        profit: 1512.0
chose: price_increase
        profit: 2296.0
chose: price_increase
        profit: 2950.0
chose: price_increase
        profit: 3252.0
chose: rnd_decrease
        profit: 3309.0
chose: rnd_decrease
        profit: 3359.0
chose: rnd_increase
        profit: 3309.0
chose: nothing
        profit: 3309.0
chose: rnd_decrease
        profit: 3359.0
chose: price_decrease
        profit: 3145.0
chose: price_increase
        profit: 3216.0
chose: nothing
        profit: 3216.0
chose: price_increase
        profit: 3338.0
chose: price_decrease
        profit: 3389.0
chose: rnd_decrease
        profit: 3439.0
chose: nothing
        profit: 3439.0
```

Three sample runs under the above parameters. In each case, the agent raises prices then cuts costs by decreasing R&D. This consistently more than doubles profits.

In an effort to get the agent to make better long term decisions, we also run our agent with a rollout policy of doing nothing, a depth of 100 and discount factor of 0.99, with the hope that this will give the agent a better sense of the long term impact of the actions it takes. Making these changes, however, did not cause a change in the agent's strategy. Perhaps this is because of how we modeled R&D costs and rewards. Potentially if we adjusted our economic model to be more realistic and and further adjusted our MCTS parameters, we could tease out better long-term strategies.

V. Conclusion and Future Work

In this paper, we seek to explore the possibility of using AI techniques to make optimal business decisions within a simplified economic simulation. We find that this is not only possible, but our agent makes decisions consistent with what most business leaders do when optimizing for the short term: raise immediate revenue and cut immediate costs. Our agent is consistently able to more than double its profit and tends to find a short term maximum profit. This is an interesting result, which suggests that more interesting insights could be obtained given a more realistic economic model.

For instance, the simulation could be expanded to include additional possible actions and budget items, such as marketing and equipment costs. Adding more details in this route could give the agent more control over the business and allow it to use more complex strategies. Another route could be adding competitors and multiple products to the mix. This could tease out interesting choices like when to enter or exit a market and how to win in a competitive market. It would also be interesting to have market conditions fluctuate over time to assess how an agent responds to changing conditions. Overall, we believe this is an exciting area that could benefit from further exploration.

References

[1] Srivastava, R., and Liu, L., “Applications of Belief Functions in Business Decisions: A Review,” *Information Systems Frontiers*, Vol. 5, No. 4, 2003, pp. 359, 378.
doi: 10.1023/B:ISFI.0000005651.93751.4b

[2] Curwin, J., and Slater, R., *Quantitative Methods for Business Decisions*, 6th ed., Thomson, 2008.

[3] “Monopolistic Competition,” *Boundless Economics*, URL:
https://courses.lumenlearning.com/boundless-economics/chapter/monopolistic-competition/ [retrieved 5 December 2019].

[4] “Time Value of Money (TVM),” *Investopedia*, URL: https://www.investopedia.com/terms/t/timevalueofmoney.asp [retrieved 5 December 2019].