

Negociação em um Modelo Multiagente

Breno W. S. R. de Carvalho

27 de junho de 2016

1 Introdução

Motivação Negociações são uma parte vital da experiência humana. Desde compras em lojas, virtuais ou físicas até definir o que jantar em família. Um mundo tão vasto e rico de possíveis interações foge completamente do escopo deste trabalho. Aqui observamos um sistema extremamente simples.

Neste trabalho é criado um modelo simples de agentes que negociam entre si. Cada agente produz um determinado recurso, mas para sobreviver ele precisa de outros recursos que possam estar disponíveis em sua sociedade, assim sendo ele precisa oferecer parte dos recursos que ele tem em troca do que lhe falta.

2 O Modelo

Existem duas formas de se enxergar o modelo construído neste trabalho. Como qualquer outro sistema multiagente [2]. Você pode observar o grafo que define o relacionamento entre os agentes e outras propriedades do mundo onde eles estão inseridos. Além disso podemos observar os agentes individualmente, como eles se comunicam e como tomam decisões.

A modelagem de sistemas multiagentes é importante pois permite criar sistemas robustos e que se recuperam de falhas, além de permitir uma visão mais natural das entidades representadas [1, 2].

O Mundo

O mundo dos agentes consiste em um grafo que contém a noção de vizinhança entre os agentes, bem como algumas propriedades fundamentais e externas aos agentes, como por exemplo, o custo de produção de determinado recurso.

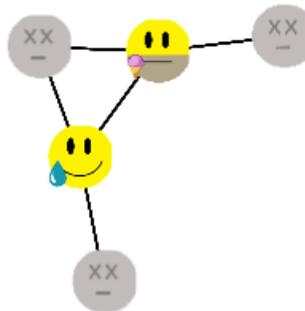


Figura 1: Um mundo com 5 agentes.

Agentes

Negociação Os agentes possuem dois tipos de recursos negociáveis, água e comida. Um agente precisa de água e de comida para sobreviver e só consegue produzir um destes por vez. Perceba que apesar de um agente poder trocar o tipo de recurso que ele produz, esta ação é muito custosa para ele, o que pode fazer com que ele morra muito mais cedo do que o esperado.

No processo de negociação, um agente pode fazer uma oferta para um de seus vizinhos, que pode aceitar ou não. Este processo gera competição entre os agentes e algo importante a ser considerado é a ordem que essas negociações podem ocorrer¹.

Necessidades Os agentes implementados neste trabalho necessitam de água e comida para sobreviver. No entanto eles também precisam *descansar* e têm um marcador de humor que eles tentam maximizar. A cada *clock* do mundo os agentes consomem uma unidade de água e de comida, ou caso não tenham esses recursos, eles aumentam seu nível de sede e fome. Quando estes níveis chegam a um determinado limite, i.e. 100%, eles mudam de estado e morrem.

Cada agente pode escolher o tipo de recurso que deseja produzir, e mudar esta escolha em qualquer *clock*, no entanto fazer essa mudança implica que o agente não produzirá neste *clock*. Ao produzir, um agente consome uma determinada quantidade do seu fator de energia que só pode ser recuperada quando o agente decide utilizar o tempo que ele teria para produzir para descansar.

Humor Cada agente neste trabalho tem um indicador de humor, $h(a)$. Este indicador de humor é alterado através de um sistema de bonificação. Além disso o agente tem associado a ele um subconjunto das seguintes características que determina como essa bonificação é ganha.

Poupador de água recebe bônus se a sua quantidade atual de água é maior que a sua quantidade de água no *clock* anterior.

Poupador de comida recebe bônus se a sua quantidade atual de comida é maior que a sua quantidade de comida no *clock* anterior.

Poupador Recebe metade da soma dos bônus de poupadores de água e poupadores de comida.

¹Não foi implementada uma fase de negociações síncrona pela dificuldade de modelar quais ofertas são aceitas ou rejeitadas sem criar a possibilidade de *dead lock*. Por exemplo, imagine um grafo que é um triângulo formado pelos agentes A , B , C . Sendo que não importa qual a oferta, cada nó só tem recursos suficientes para aceitar uma oferta. Considere que todos os agentes enviaram suas ofertas para os outros. Se A e B aceitarem ambos as ofertas de C teremos um conflito. Este conflito poderia ser resolvido de diversas formas, uma delas é atribuindo prioridades a certos nós, um fator aleatório de desempate, comparação de qual transação é mais vantajosa para o ofertante, ...

Hedonista Recebe bonus se a quantidade de recursos for menor que no *clock* anterior.

Esnobe recebe bônus proporcional a quantidade de vizinhos com menos recursos que ele dividida pela quantidade total de nós.

Minimalista recebe bônus proporcional a quantidade de vizinhos com mais recursos que ele dividida pela quantidade total de nós.

Benevolente recebe bônus proporcional a quantidade de vizinhos vivos e felizes dividida pela quantidade total de nós.

Maléfico recebe bônus proporcional a quantidade de vizinhos mortos ou infelizes dividida pela quantidade total de nós.

Os agentes da figura 1, por exemplo são poupadores de água.

Agente Simples

O agente simples é a primeira versão de agente analisada neste trabalho. No futuro seria interessante utilizar agentes com regras mais complexas, inclusive com aprendizado (um modelo de regressão, talvez).

Humor Um agente simples a tem um fator de emotividade, $0 \leq \lambda_a \leq 1$. Ao tomar decisões eles podem utilizar uma heurística para decidir qual escolha é a mais proveitosa. A emotividade permite incluir, de certa forma, nesta heurística os objetivos dos agentes. Assim sendo o agente simples utiliza o seu tempo estimado de vida após a escolha, $t(e(a))$, e o seu humor, $h(a)$, para avaliar o valor de uma escolha: $v_a(e) = (1 - \lambda_a)t(e(a)) + \lambda_a h(e(a))$.

Negociação Para sobreviver os agentes trocam ofertas entre si. A forma como este agente calcula a sua oferta é subtrair o produto do recurso com maior quantidade r^+ com o seu custo $c(r^+)$ do produto do recurso com menor quantidade r^- com o seu custo $c(r^-)$, i.e. $d(r^+, r^-) = |r^+c(r^+) - r^-c(r^-)|$. Isto permite calcular o quanto se pedir e o quanto oferecer de cada recurso. Assim, ele oferecerá $\frac{d(r^+, r^-) * c(r^+)}{2}$ de r^+ por $\frac{d(r^+, r^-) * c(r^-)}{2}$ de r^- .

De certa forma estamos tentando tornar dois recursos distintos comparáveis. Isto foi feito em termos do custo de produção, mas poderia ter sido feito em termos de quão importante o recurso é para a sobrevivência do agente.

A forma como um agente simples decide aceitar uma oferta o é se o seu novo estado após escolher a oferta, $e_o(a)$ for superior ao seu, $e(a)$. Por tanto a inequação $e_o(a) > e(a)$ deve ser satisfeita para que ele aceite a oferta e a troca de recursos seja realizada.

Produção Além das necessidades por recursos e negociações o agente deve respeitar o próprio nível de energia que é decrescido a cada vez que o agente produz recurso e só é restaurado se o agente se abster de produzir e descansar. A escolha de qual ação tomar (escolher produto, produzir, descansar), é feita segundo o algoritmo 1.

Algoritmo 1: *Escolha da próxima ação*

```
Dados: Agente  $a$ 
se  $a$  não tem um produto para produzir então
|  $a$ .produto = escolha_aleatória(água, comida)
senão
| se  $e_p(a) \geq e(a)$  então
| | // Se o estado após  $a$  produzir é maior que o atual
| |  $a$ .produz()
| senão
| |  $a$ .descansa()
```

3 Conclusão

Este trabalho implementa um sistema simples de agentes onde os agente precisam negociar para sobreviver. Com esta implementação foi possível notar comportamentos interessantes e características deste ambiente.

Uma análise mais formal do sistema criado bem como a utilização de novos tipos de agentes (possivelmente com aprendizado) são alguns dos trabalhos futuros possíveis.

Um fenomeno interessante quando temos um caminho entre duas componentes conexas de um grafo (conectadas apenas por este caminho) é o surgimento de mercados quando os produtos têm custos distintos. Outro fenômeno interessante é que comunidades de nós benevolentes não parece atingir um equilíbrio com mais nós vivos do que uma comunidade neutra.

Referências

- [1] Stuart Jonathan Russell, Peter Norvig, John F Canny, Jitendra M Malik, and Douglas D Edwards. *Artificial intelligence: a modern approach*, volume 2. Prentice hall Upper Saddle River, 2003.
- [2] Michael Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2009.