

Uso do *framework* motivacional de Breazeal em NPCs

Marcos Côrtes

Rodrigo Richa

Esteban Clua

Anselmo Montenegro

marcos.v.p.cortes@gmail.comrodrigoricla@gmail.comesteban@ic.uff.branselmo@ic.uff.br

Universidade Federal Fluminense – Instituto de Computação, Brasil

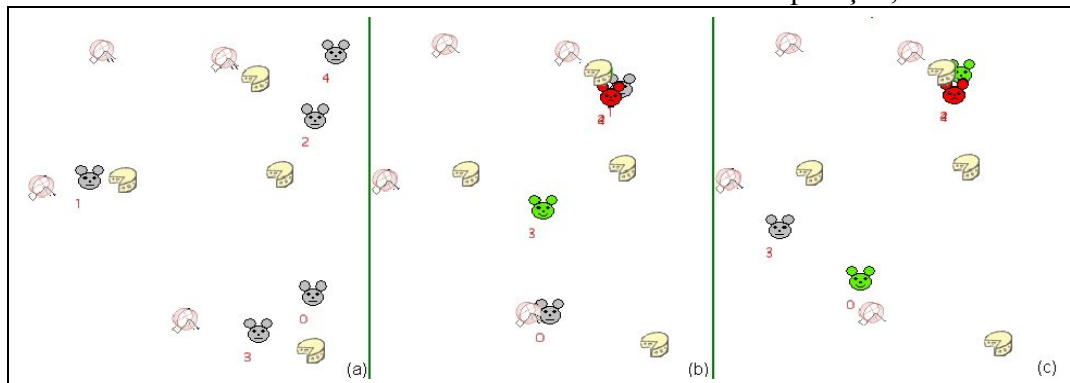


Figura 1: Imagem do protótipo de simulador de NPCs usando o *framework* de Breazeal exibindo instantes (a) 6 segundos, (b) 19 segundos e (c) 23 segundos da mesma simulação do gráfico 1

Resumo

Atualmente, tecnologias de entretenimento como games e TV digital possuem maior complexidade e exigem maior realismo dos seus personagens (maior “ilusão de vida”). Métodos tradicionais em IA de jogos para controle de NPCs, como máquina de estados, embora sejam suficientes para algumas tarefas de vida artificial, são pouco escaláveis e flexíveis, pois precisam de um modelo rígido e lógico do objeto simulado. Este artigo descreve o uso de um sistema motivacional baseado no *framework* de Breazeal na inferência de movimento e comunicação de NPCs, sendo uma abordagem alternativa e mais flexível no escopo dos jogos digitais.

Palavras-chave: Atores sintéticos, Inteligência Artificial, Vida Artificial, Modelo de Emoção, NPC, Agentes.

1. Introdução

A simulação de NPCs em jogos digitais são normalmente implementados através de máquina de estados e agentes. [POZZER 03] argumenta que com os sistemas de simulação interativos exigem cada vez mais sistemas de IA mais escaláveis e complexos. [SILVA 02] também sugere a necessidade de buscar alternativas aos métodos clássicos de controle de NPCs.

Com isto em mente, propomos neste trabalho um agente/NPC controlado sistema motivacional com emoções que influem no seu comportamento e gerar a comunicação entre eles. Usamos como base o

framework de [BREAZEAL 98].

O protótipo foi construído para ser genérico, permitindo a um game designer ou especialista de domínio possa especificar os requisitos deste agente e possa implementá-los no motor motivacional.

2. Framework de Breazeal

O sistema de [BREAZEAL 98] consiste em um robô chamado *Kimet* que simula a interação de uma criança com seu tutor (*infant-caretaker dyad*). O objetivo do agente robô é aprender uma forma eficiente de obter suas demandas junto ao tutor humano através de suas expressões faciais, que são representações externas de suas emoções.

O *framework* é composto por três classes de entidades: *drives* (pulsões), emoções e comportamentos. Todas estas entidades são consideradas processos (*process*) implementados como um operador (um *transducer*) como na equação 1. O processo recebe n entradas i , sendo que cada uma possui um peso w_i , podendo ser este positivo ou negativo. O somatório destes valores somado a uma interferência b (*bias*) gera o valor de saída. Este valor de saída é chamado de **carga**, **energia** ou simplesmente **valor**. As entradas i de um processo são normalmente valores de carga de outro processo. A ligação de um processo a outro é chamada de **influência**.

Cada componente está sujeito a “ser ativado” quando extrapola um limiar (*threshold*) superior (ou inferior). Quando isto ocorre, o processo ativa uma computação para tal evento, como por exemplo influenciar determinado comportamento [MAES 90].

$$x = \sum_{j=0}^n w_j \cdot i_j + b$$

Equação 1: Operador (*transducer*)

3. Implementação proposta

A implementação proposta considerou que todas as três entidades originais derivam de um processo. Este é discretizado de tal forma que a cada tempo t , seja executada a Equação 1 considerando os valores de i do tempo anterior ($t-1$).

O processo deve possuir um limiar superior e um inferior. Quando um desses eventos ocorre, o processo passa a influenciar outro. Isto é: Se o processo “fome” extrapolar o limiar superior, ele passa a influenciar outro processo, o “comer”.

Além disso, adicionamos mais uma classe de itens: “Sensor”. Este permite a ligação do sistema com o mundo externo. Basicamente eles são versões simplificadas de um processo que tem seus valores dependentes do estado do ambiente.

No final, teremos uma “rede” (grafo) de processos que se influenciam partindo dos sensores e indo até os comportamentos, que gerarão o controle necessário ao NPC.

A Tabela 1 demonstra mais detalhadamente quais os processos definidos e as relações adotadas. Ela foi constituída pensando nos requisitos de simular nosso problema abordado no protótipo (uma comunidade de camundongos).

Devemos o *framework* é **simbólico** e modela o funcionamento fisiológico de seres vivos. Não devemos confundí-lo como um modelo conexionista (ex. rede neural). A teoria adotada por ambos os modelos são diferentes e incompatíveis.

3.1 Drive

Um *drive* como um impulso natural (essencial) **que deve sempre** ser satisfeito pelo agente. Sua função é simular necessidades fisiológicas naturais de um ser vivo que aumentam com o decorrer do tempo devido ao consumo do organismo de seus recursos (exemplo: fome e sede).

Quando o valor de um *drive* atinge um limiar, ele entra para o estado ativo e precisa ser satisfeito. Isto equivale influenciar um comportamento que o satisfará.

Um exemplo de *drive* seria a fadiga: A locomoção do agente e gastos cognitivos incrementariam este *drive*. Quando ele excede um limiar é ativado, influenciando o “desejo de dormir” do agente para satisfazer o *drive* ativado.

Para os *drives*, os limiares de ativação podem ser inferiores e/ou superiores.

3.2 Comportamento

Comportamentos são como atividades (ações, reações, padrões) observáveis num ser vivo e também considerados como uma mensagem do agente para todos os elementos do ambiente. Um comportamento que satisfaz um *drive* é um comportamento consumidor (*consummatory behavior*) deste *drive*.

Existe uma outra classe de comportamento: os facilitadores (*appetitive behaviors*). Cada comportamento consumidor possui um ou mais comportamentos facilitadores. Quando um comportamento consumidor não pode ser realizado por algum motivo, tal como a ausência de variáveis para ele se realizar ou algum impedimento, o sistema de inferência tem em mãos os comportamentos facilitadores para mudar o estado do ambiente e do agente para um outro estado apto a execução do comportamento consumidor.

Um comportamento varia seu valor de zero até infinito e possui somente o limiar superior. Quando o seu valor extrapola o limiar superior, ele é ativado e o comportamento passa a ser efetuado, se possível. E um comportamento pode ser executado junto com outro, pelo simples fato de que foram especificados de tal forma que um não anula a presença do outro (não existe um comportamento andar e outro correr), salvo o comportamento de dormir, que inibe todos os outros.

Por último, os comportamentos são implementados associados a um algoritmo de “tentar executar” (*try play*) que sempre é chamada quando ele extrapola o limiar superior. Esta função realiza a ação do agente no ambiente e determina uma especialização do comportamento. Caso ele não consiga executar por alguma razão (como por exemplo, comportamento de comer ser inviável, pois não há comida perto o suficiente), o comportamento tenta executar os seus comportamentos facilitadores.

3.3 Emoção

A emoção também é um processo. Entretanto em cada passo de cálculo é decrementado de um valor definido. Sempre que for extrapolado o seu limiar superior ela é ativada. Caso mais de uma emoção seja ativada, apenas será exibida a que possuir maior valor. Ela é exibida através da mudança do desenho do NPC.

A implementação proposta utiliza um modelo baseado em Elkan e Plutchik, tal como o framework de [BREAZEAL 98] (na verdade usamos um subconjunto de itens do modelo original de [BREAZEAL 98]). Estes modelos são classificados como modelos de “emoções puras” ou “emoções discretas”. Estes modelos determinam que as emoções sejam compostas por um número limitado de emoções primitivas discretas e disjuntas.

3.4 Sensor

Para que o agente possa ter percepção do mundo e gerar a comunicação entre os agentes, foi adicionado ao sistema o conceito de sensores. Estes sensores podem servir de entrada aos processos e possuem duas funções básicas: podem inferir algo em relação ao mundo (função ativa) e receber e armazenar alguma mensagem enviada ao agente (função passiva). O primeiro caso corresponde ao sensor que busca itens no ambiente, como comida ou outro agente. O segundo são receptores que podem receber mensagens de outros agentes. Assim, qualquer alteração provocada de um agente para o outro deve ser propagada por uma mensagem do agente causador ao agente afetado.

No segundo caso, estes sensores podem servir de influência a processos. O valor desta influência é dado como na equação 1, porém a incógnita i é a intensidade de uma mensagem j . Em outras palavras, a influência de um sensor passivo é dada pelo somatório das intensidades das mensagens capturadas.

Com isto, foi possível gerar interações entre os NPCs, onde eles podiam conversar e brigar um com os outros.

4. Calibração e Resultados

Depois de implementado, o sistema precisou ser calibrado. Isto consiste em mudar os valores de pesos do sistema de tal forma que apresentasse um comportamento satisfatório, entenda-se como satisfatório o fato de transmitir ilusão de vida, tema já discutido no início do artigo.

Para calibrações ruins o problema mais comum que ocorreu é uma tendência aos agentes a se comportarem de uma mesma forma, convergindo para um único tipo de emoção, como por exemplo, todos ficarem com raiva. Ou ainda apenas alguns *drives* ou emoções serem ativados em detrimento de outras.

Com algumas tentativas foi possível chegar em uma situação balanceada, onde todos os processos puderam ser ativados e os agentes apresentaram situações de correlação entre suas ações, tais como:

1. Um agente A fica com raiva por que não achou um brinquedo e começa a agredir o agente B. O agente B por sua vez fica triste ou com medo;
2. Um grupo de agentes fica feliz, pois estão mais próximos de um brinquedo e uma fonte de comida;
3. Agentes alegres ficam juntos, pois estão realizando o comportamento de conversar.

4.1 Resultados

Para fins de análise, se acompanhou o sistema motivacional de um NPC (o agente 0), conforme o

gráfico 1.

Desde o início do tempo, percebe-se o *drive* “vontade de brincar” em crescimento. Chegando no instante 6 (figura 1a) ele é ativado e passa a influenciar o comportamento brincar. Este em algum momento é ativado e força o agente a ir atrás de um brinquedo para satisfazer o *drive*. No instante 19 (figura 1b) isto ocorre e o *drive* “vontade de brincar” é satisfeito: como está negativo, extrapola seu limiar inferior e passa a influenciar a emoção “Feliz”. Esta é incrementada e no pico da curva, no instante 25 (figura 1c), o agente esta exibindo uma caricatura feliz.

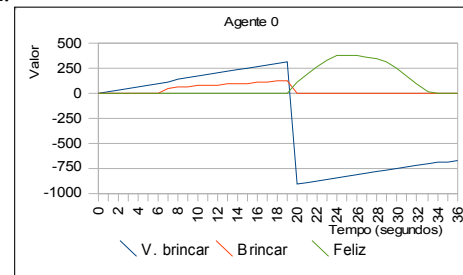


Gráfico 1: Gráfico de alguns processos em do agente 0 em determinada simulação

5. Considerações finais

O presente trabalho demonstra uma possível implementação para simulação de NPCs com inferência sobre suas necessidades através da adaptação do *framework* de Breazeal para este problema. Este método é genérico o suficiente para que se possa aplicá-lo a várias naturezas de agentes, desde animais, NPCs de RPG ou outros.

Verificou-se que a técnica proposta por [BREAZEAL 98] permite uma boa solução para simulação de NPCs. O grande ganho deste trabalho é usar o modelo na geração de movimento dos NPCs e sua interação (comunicação). Obtendo-se razoável “ilusão de vida” com baixo custo computacional e um algoritmo relativamente simples.

Referências:

- [BREAZEAL 98] Breazeal, C., 1998. A motivational system for regulating human-robot interaction. Proceedings of the fifteenth National Conference on Artificial Intelligence, Madison, EUA, pp54-61.
- [BULLOWA 79] Bullowa, M., 1979. Before Speech: The Beginning of Interpersonal Communication, Cambridge University Press, Cambridge, London [in RODRIGUES].
- [CAÑAMERO 97] Cañamero, L., 1997. A Hormonal Model of Emotions for Behavior Control. Proceedings of the 4th European Conference on Artificial Life (ECAL'97). [in MORGADO 05].
- [DAMASIO 94] Damásio, A., 1994. O Erro de Descartes. Europa-América [in ZAGALO 04].

- [FURTADO 02] Furtado, A. L.; Ciarlini, A. E. M. 2002. Cognitive and affective motivation in conceptual modelling. Avaliado em: http://editorial.unab.edu.co/revistas/rcc/pdfs/r32_art1_c.pdf [em 01 de maio de 2008].
- [GUDWIN 08] Gudwin, R.. Agentes com Emoções. Avaliado em: <http://www.dca.fee.unicamp.br/~gudwin/courses/IA009/> em 20/05/2008.
- [MAES 90] Maes, P., 1990. Learning Behavior Networks from Experience, ECAL90 [in BREAZEAL].
- [MORGADO 05] Morgado, Luís Filipe Graça, 205. Integração de Emoção e Raciocínio em Agentes Inteligentes. Universidade de Lisboa, Lisboa, Portugal.
- [POZZER 03] Pozzer, C. Furtado, A.F.; Ciarlini, A. E. M., 2003. Agentes e emoções em histórias interativas. Departamento de Informática da PUC-RJ, Rio de Janeiro, Brasil. Avaliado em: ftp://ftp.inf.puc-rio.br/pub/docs/techreports/03_39_pozzer.pdf [em 20 de abril de 2008].
- [ORTONY 88 *at al*] Ortony, A.; Clore, G.; Collins, A., 1988. The Cognitive Structure of Emotions Cambridge University Press. [in MORGADO].
- [SEP 07] SEP, 2007. Emotion. Artigo da Stanford Encyclopedia of Philosophy. Acessado em: <http://plato.stanford.edu/entries/emotion> [em 05 de maio de 2008].
- [SILVA 02] Silva, D. R. D.; Tedesco, P. R. T.; Ramalho G. L., 2006. Usando Atores Sintéticos em Jogos Sérios: O Case SmartSim. SBGames 2006, Pernambuco, Brasil. Avaliado em: <http://cin.ufpe.br/~sbgames/proceedings/files/Usando%20Atores.pdf> [em 01 de maio 2008].
- [RODRIGUES 07], Rodrigues P. S. L., 2007. Um sistema de geração de expressões faciais dinâmicas em animações faciais 3D com processamento de fala. Tese de doutorado, Departamento de Informática da PUC-RJ, Rio de Janeiro, Brasil.
- [VELÁSQUEZ 98] Velásquez, J., 1998. Modeling Emotion-Based Decision-Making. Proceedings of the 1998 AAAI Fall Symposium Emotional and Intelligent: The Tangled Knot of Cognition, 164-169. AAAI Press.
- [ZAGALO 04] Zagalo, N., Branco, V., Barker, A., 2004, Emoção e Suspense no Storytelling Interactivo, in Actas, Games2004 - Workshop Entretenimento Digital e Jogos Interactivos, Lisboa, Portugal, (pp.75-84)

Influência / Influenciado	Medo	Raiva	Feliz	Triste	Fome	V. de brincar	V. de conversar	Dor de barriga	Cansaço	Comer	Brincar	Dormir	Conversar	Bater	Defecar	Move Random
Medo	0	-,O	0	0	0	0	0	0	0	0	0	0	0	-,O	0	0
Raiva	0	0	0	-,O	0	0	0	0	0	0	0	0	0	+,O	0	0
Feliz	0	0	0	0	0	0	0	0	0	0	0	0	0	-,O	0	0
Triste	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fome	+,O	0	0	0	0	0	0	0	0	+,O	0	0	0	0	0	0
V. de brincar	0	0	0	-,U	0	0	0	0	0	0	+,O	0	0	0	0	0
V. de conversar	0	-,O	-,U	0	0	0	0	0	0	0	0	0	+,O	-,O	0	0
Dor de barriga	0	+,O	-,U	+,O	0	0	0	0	0	0	0	0	0	0	+,O	0
Cansaço	0	0	0	0	0	0	0	0	0	0	0	+,O	0	0	0	0
Comer	0	+,O	0	0	0	0	0	0	+,O	0	0	0	0	0	0	-,E
Brincar	0	0	0	0	0	0	0	0	+,O	0	0	0	0	0	0	-,E
Dormir	0	0	+,U	0	0	0	0	0	0	0	0	0	0	0	0	-,E
Conversar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-,E
Bater	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-,E
Defecar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Move Random	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Legenda																
0	Não há relação															
+	A influência é positiva															
-	A influência é negativa															
U	A influência ocorre quando o influenciador extrapolar um limiar inferior (under_state)															
O	A influência ocorre quando o influenciador extrapolar um limiar superior (over_state)															
E	A influência ocorre em ambos os limiares															

Tabela 1: Tabela do grafo de relação entre processos do sistema motivacional