Aprendizagem por Reforço com Rede Neural no Desenvolvimento de Jogos Digitais

Edival F. S. Assis[1], Maury M. Gouvêa Jr.[2]
[1]*Instituto de Ciências Exatas e Informática*
[2]*Instituto Politécnico*
Laboratório de Robótica e Inteligência Artificial
Pontificia Universidade Católica de Minas Gerais
Belo Horizonte, Brasil
edival.assis@sga.pucminas.br, maury@pucminas.br

***Abstract*—Computer games are very popular today and have given rise to a multi-billion dollar industry. One of the milestones that permitted this progress was Artificial Intelligence, which provided more realism and immersion. The use of learning methods has brought computer games to a new level, which enables them to be adapted while they are being played. This paper presents a reinforcement learning method for a non-player character of a computer game. This character is represented by a feedforward neural network and its learning is performed by a backpropagation algorithm. The tests showed that the non-player character is able to increase its performance by using the proposed method.**

Keywords— Non-player Character; Machine Learning; Artificial Neural Network; Reinforcement Learning

I. INTRODUÇÃO

O mercado de desenvolvimento de jogos digitais é um dos maiores de entretenimento do mundo. Na época do seu surgimento, os jogos digitais apresentavam como principal característica a possibilidade de dois jogadores humanos se enfrentarem. Com o passar dos anos, os jogos digitais necessitaram apresentar desafios aos seus usuários. Os personagens, aliados ou não, e o ambiente deveriam interagir com os jogadores de forma mais inteligente, proporcionando maior realismo e imersão. Com a inclusão da Inteligência Artificial (IA) nos jogos digitais, isso se tornou mais factível, pois esta área da Ciência da Computação busca reproduzir a inteligência humana, apresentando um amplo campo para desenvolvimento [1].

Um dos principais patamares da IA para jogos digitais é a aprendizagem, permitindo que os jogos possam evoluir e adaptar-se quanto mais forem jogados [2]. Nos últimos anos, diferentes técnicas de IA vem sendo utilizadas em jogos digitais [3]. No entanto, as redes neurais ainda são pouco aplicadas em jogos digitais, principalmente utilizando aprendizagem por reforço [4]. Os desenvolvedores de jogos digitais apresentam certo receio em utilizar redes neurais para o aprendizado on-line, pois os agentes podem apresentar comportamentos indesejáveis.

No paradigma de aprendizagem por reforço, o agente aprende por tentativa e erro, interagindo com o ambiente. O agente é recompensado por executar ações eficientes ou punido caso contrário. A aprendizagem por reforço vem sendo aplicada em várias pesquisas, como robótica móvel [5], times de robôs [6], jogos de tabuleiro [7], dentre outros.

Este trabalho apresenta um método de aprendizagem por reforço com redes neurais aplicado a jogos digitais de ação em terceira pessoa. O modelo proposto deve permitir que um personagem não controlado pelo jogador (NPC, do inglês *non-player character*) explore o ambiente interativamente, i.e., o agente executa uma ação e, conforme a sua eficiência, um reforço positivo ou negativo é dado ao agente. O ambiente possui itens, que devem ser recolhidos, e inimigos, que atacam e devem ser atacados. O NPC é constituído por uma rede neural *feedforward* que utiliza aprendizagem por reforço para buscar suas melhores ações. Para treinamento da rede neural utiliza-se o algoritmo *backpropagation* baseado em aprendizado por reforço. O modelo é validado em um estudo experimental onde o agente percorre o ambiente por várias iterações e seu desempenho é avaliado ao longo do tempo.

O restante deste artigo está organizado como segue. A Seção II apresenta os conceitos básicos da aprendizagem por reforço. A Seção III descreve o método proposto para o NPC baseado em aprendizagem conexionista por reforço. Na Seção IV, são apresentados os experimentos que validam o método proposto. A Seção V conclui o trabalho e apresenta propostas para trabalhos futuros.

II. APRENDIZAGEM POR REFORÇO

A aprendizagem por reforço permite a um determinado agente aprender interagindo com o ambiente sem a presença de um tutor [5]. Dessa forma, o agente tem que decidir qual ação tomar buscando uma política ótima ou quase ótima por meio das recompensas observadas [1].

A Figura 1 mostra que em cada passo da interação o agente observa o estado corrente do ambiente, st, e escolhe uma determinada ação, at. Realizada essa ação, modifica-se o estado do ambiente, s_{t+1}, que retorna ao agente um sinal de reforço, $r_{s,a}$, que pode ser uma recompensa ou penalização [8].

Por meio de tentativa e erro o agente aprende em relação ao ambiente, criando um mapa estado-ação. O objetivo do

aprendizado é definir qual ação tomar a cada iteração que maximize o valor da recompensa [9].

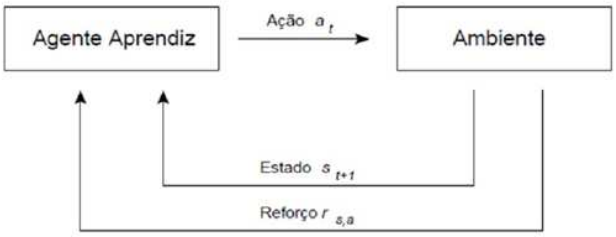

Figura 1. Ciclo Percepção-Ação

III. NPC Baseado em Aprendizagem por Reforço Conexionista

O NPC, constituído por uma rede neural *feedforward* multicamada [10], tem como entradas o estado atual do ambiente e uma possível ação a ser tomada. Como saída, o agente fornece o sinal de reforço, a ser maximizado. Assim, para o estado atual, o NPC avalia qual ação produz o maior reforço. A cada ação executada, a rede neural utiliza o algoritmo *backpropagation* para ajuste dos seus pesos. As próximas subseções apresentam as partes constituintes do jogo digital utilizado para implementação do método proposto.

A. Ambiente

O ambiente do NPC é um espaço 3D, criado no Unity 3D versão 2.6.1, que contém um tabuleiro 10x10, totalizando 100 posições. O personagem pode movimentar-se para cada posição adjacente do tabuleiro nos sentidos norte, sul, leste e oeste. No ambiente, existem itens a serem capturados pelo NPC e inimigos que podem ser atacados e atacam o NPC. Inicialmente, todos esses elementos são dispostos em posições aleatórias do tabuleiro. A Figura 2 mostra o ambiente do jogo com o NPC, alguns itens e inimigos.

Figura 2. Ambiente 3D com o NPC, itens a serem recolhidos e inimigos.

B. Campo de Visão do Personagem

O NPC visualiza o que está ao seu redor por meio de uma matriz 3x3 correspondente às posições adjacentes. Os inimigos e itens, quando entraram no alcance da visualização do NPC, produzem um valor positivo na matriz de visualização, sendo 1 para inimigo, 2 para item a ser capturado e 0 para nenhum item ou inimigo presente na vizinhança. A Figura 3 mostra uma matriz de visualização que representa um NPC com dois inimigos à frente e um item atrás.

1	1	0
0		0
0	0	2

Figura 3. Matriz de visualização do NPC

A matriz de visualização do personagem, a ser apresentada para a rede neural, é representada por uma sequência de bits, sendo cada par de bits uma adjacência; assim, 00 = 0, 01 = 1 e 10 = 2. A leitura é feita começando do canto superior esquerdo para a direita até o final da linha, depois recomeçando à esquerda da linha abaixo, e assim por diante. Por exemplo, a matriz de visualização representada na Figura 3 tem a seguinte sequência de bits: { 01, 01, 00, 00, 00, 00, 00, 10 }.

C. Ação

O NPC pode executar duas ações básicas, a saber, explorar ou atacar um inimigo em um dos quatro sentidos possíveis. Se, ao explorar, houver um item na nova posição, este item é capturado. A Tabela 1 mostra as ações e seus equivalentes binários.

Tabela 1. Ações executadas pelo NPC

Ação	**Função**	**Equivalente Binário**
1	Ataque norte	000
2	Ataque sul	001
3	Ataque leste	010
4	Ataque oeste	011
5	Explorar norte	100
6	Explorar sul	101
7	Explorar leste	110
8	Explorar oeste	111

A tomada de decisão, isto é, a escolha da ação a ser executada, é descrita na Seção III.G.

D. Estado

O estado é representado pela localização do NPC no tabuleiro e pela matriz de visualização. As 100 posições do tabuleiro são representadas por uma sequência binária equivalente ao intervalo decimal de 0 a 99.

E. *Função Retorno*

Um retorno é sempre produzido após a execução de uma ação do NPC, podendo ser positivo, negativo ou nulo. Um retorno positivo representa um estímulo ao agente, decorrente de uma ação bem sucedida; um retorno negativo uma repreensão, por uma ação mal sucedida; e retorno nulo significa que o agente executou uma ação que não lhe trouxe nenhum ônus ou bônus. A Tabela 2 mostra os valores de retornos que o NPC pode receber do ambiente. Recolher um item é a ação que trás maior recompensa para o agente, pois esta é a sua principal função.

Tabela 2. Retorno do ambiente

Consequência da Ação do NPC	Retorno
Dano	-3
Derrota um inimigo	1
Recolhe um item	5
Movimento simples	0

F. *Estrutura do NPC*

Para representar o estado do NPC, foi utilizado uma rede neural feedforward de três camadas, com 27 entradas binárias, 14 neurônios na camada oculta e uma saída. Dentre as 27 entradas da rede neural, 8 delas representam a posição do personagem, 16 a matriz de visualização e 3 a ação do NPC. A saída fornece o sinal de reforço. Os 14 neurônios da camada oculta foram estabelecidos empiricamente. A ação do NPC é determinada em função do estado atual – posição, borda e visualização – e da ação que produz a maior saída da rede neural.

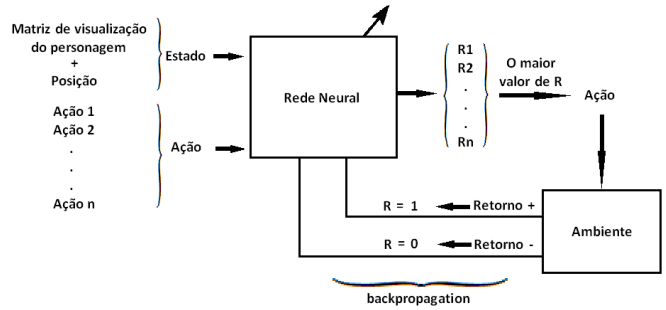

Figura 4. Sistema de aprendizagem por reforço com rede neural adotado

G. *Aprendizagem do NPC*

Em cada iteração, o NPC percebe o estado, posição e visualização da vizinhança, e verifica qual ação produzirá o maior reforço. Para cada estado + ação, a rede neural produz um sinal de 0 a 1, que representa uma estimativa do sinal de reforço. As saídas que tendem para 1 são produzidas por estado + ações que deverão ser bem sucedida e aquelas saídas que tendem para 0, ao contrário, são produzidas por estado + ações que deverão ser mal sucedidas. A ação escolhida é aquela que produzir a maior saída da rede neural para uma dada entrada, i.e., estado + ação.

Depois de executar a ação sugerida, o algoritmo de aprendizagem ajusta, em tempo real, os pesos da rede neural como segue: com um retorno positivo do ambiente, o elemento de aprendizagem considera que a saída desejada para a entrada atual (estado + ação que produziu maior saída) será 1 (ação bem sucedida); caso contrário, com um retorno negativo do ambiente, a saída desejada para a entrada atual será 0. O algoritmo de aprendizagem por reforço conexionista proposto é executado como segue:

1. Para um dado estado, aplicar na RN as entradas correspondentes a todas as ações possíveis, i.e., { p, v, ai }, i = 1, ..., 8;
2. Escolher a ação que produzir a maior saída da RN;
3. Movimentar o agente e obter o retorno do ambiente;
4. Se o retorno for positivo, então a saída desejada será 1; caso contrário, retorno negativo, saída desejada será 0;
5. Ajustar os pesos da RN com n interações utilizando o algoritmo backpropagation;
6. Voltar ao passo 1.

IV. Estudo Esperimental

Foram utilizados dois conjuntos de testes, o primeiro com algumas alterações no modelo de aprendizagem e o segundo referente às ações do personagem. O primeiro conjunto de teste está relacionado ao retorno que o personagem recebe do ambiente a cada interação: se ocorrer um retorno positivo o algoritmo de aprendizagem ajusta sua saída para 1; caso contrário, o ajuste é feito para 0. O segundo conjunto de teste está relacionado à disponibilidade da execução das ações pelo NPC. As próximas subseções apresentam detalhes dos dois conjuntos de testes e seus respectivos resultados.

A. *Primeiro Conjunto de Testes*

Foram efetuados 10 testes para cada configuração, executando-se 20.000 interações. Os grupos foram estruturados como segue:

Grupo 1: Os ajustes dos pesos são feitos de forma gradual após reforços positivos para 1 e negativos para 0;

Grupo 2: Os ajustes dos pesos são feitos após reforços positivos para 1 e negativos para 0;

Grupo 3: O personagem ao receber um retorno negativo ajusta a saída da rede neural para 0 apenas para esta ação, porém se o retorno é positivo o ajuste da saída é feito para 1 referente a esta ação. As demais ações que não foram eleitas para este estado são ajustadas como se o retorno para elas fosse negativo, sendo assim para 0.

Nos testes iniciais, grupos 1, 2 e 3, o personagem não aprende as ações ideais diante do estado que se encontra, ficando preso apenas a um determinado número de ações. O personagem inicia o teste com um determinado número de ações (explorar ou atacar) e, no decorrer do teste, persiste com estas ações, como mostra a Figura 5. Observa-se que não ocorre uma alternância entre as ações e o personagem apenas ataca. Neste caso não morre, mas também não recolhe nenhum item, ou ainda, ele consegue recolher os itens porém não consegue atacar, morrendo sempre que entra em contato

com algum inimigo. Também foi observado que a maioria das ações do personagem foi direcionada para as bordas passando raramente pelo centro da plataforma.

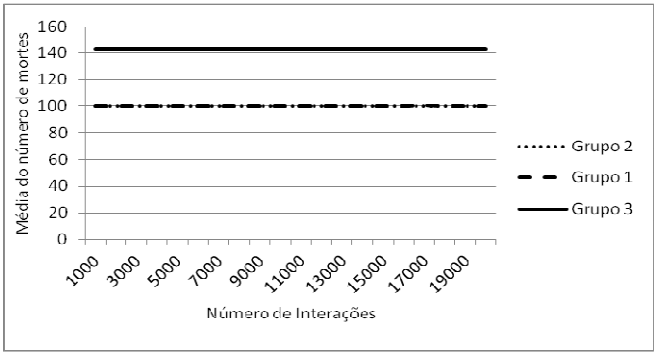

Figura 5. Médias dos grupos 1, 2 e 3

Nesta configuração, o personagem fica preso entre posições que não possuem um retorno do ambiente, evidenciando a falta da influência exercida pela aprendizagem por reforço da rede neural. É necessário algum retorno do ambiente para que o personagem possa ter um direcionamento, i.e., se está agindo adequadamente ou não.

Passando por posições que não apresentam inimigos ou itens a ser recolhidos, o personagem executa a ação que possui o maior valor de saída, e aguarda o retorno. Entretanto, se não ocorrer qualquer retorno do ambiente para direcionar o personagem, este fica preso entre duas posições vazias do tabuleiro.

Não foram notadas mudanças significativas para as simulações dos grupos 1, 2 e 3. O número médio de mortes do personagem manteve-se constante nas três simulações, e a configuração adotada para o primeiro grupo de testes mostrou-se ineficiente. Essa configuração não permitiu que o agente aprendesse qual a melhor ação em função do estado – mesmo com um número elevado de interações realizadas.

B. Segundo Conjunto de Testes

Para o segundo grupo de testes foi proposto que o NPC deveria atingir um ponto determinado do tabuleiro, obrigando-o a passar por regiões que não estavam sendo exploradas na primeira configuração. Para que isto fosse possível, sem indicar qual a nova posição o NPC deveria tomar, foi disponibilizado apenas as posições mais próximas a que o NPC deveria alcançar. Assim, o NPC deveria decidir entre explorar ou atacar – comprometendo parcialmente a função de explorar.

Neste grupo de testes, foram efetuados 10 testes para cada configuração, cada teste executando 20.000 interações. Os grupos foram estruturados como segue:

- Grupo 4: o ajuste é feito de forma gradual após reforços positivos para 1 e negativos para 0;
- Grupo 5: o ajuste é realizado diretamente após os reforços positivos para 1 e negativos para 0;
- Grupo 6: o personagem ao receber um retorno negativo, ajusta a saída da rede neural para 0 apenas para esta ação. Se o retorno for positivo, o ajuste da saída será feito para 1, e as demais ações que não foram eleitas para este estado são ajustadas como se o retorno para elas fosse negativo.

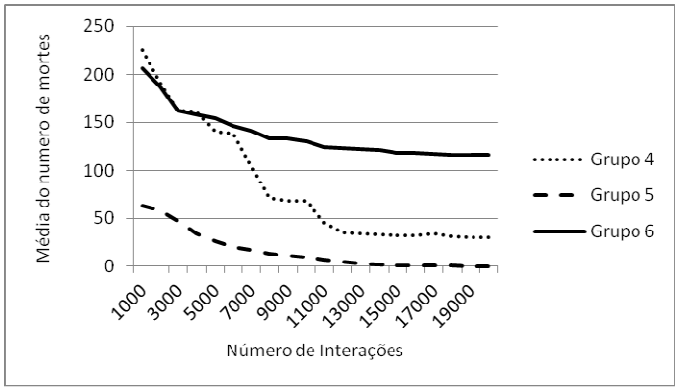

Figura 6. Médias dos grupos 4, 5 e 6

Neste grupo de testes, houve uma melhoria significativa em relação aos três primeiros grupos testados. Nesta configuração, o NPC foi obrigado a passar por regiões que antes não passava, obrigando-o a interagir mais com o ambiente. No entanto, observou-se que a partir de um determinado instante, o NPC optava por atacar ao invés de recolher os itens de forma que foi reduzido o número de mortes, assim como o número de itens recolhidos, como mostra a Figura 6.

Conclui-se que a opção de o NPC optar por atacar em vez de recolher os itens está relacionada ao tipo de retorno adotado. Caso o NPC entre em contato com algum inimigo, ele é punido com um reforço negativo. Consequentemente, se o NPC atacar um inimigo, receberá um reforço positivo. Em relação ao item a ser recolhido, o NPC apenas recebe o reforço positivo se o recolher.

Como o modelo de aprendizado adotado é direcionado para receber um reforço positivo ou negativo de modo qualitativo e não quantitativo, o valor do retorno positivo referente a recolher um item para o personagem não é suficiente para que ele possa equilibrar as ações que deve efetuar.

C. Terceiro Conjunto de Testes

Para o terceiro grupo de testes, a quantidade de funções de retorno em relação aos elementos integrantes do ambiente foi equilibrada. Diferentemente dos demais conjuntos de teste, se o NPC passar por algum item sem recolhê-lo, ele receberá um reforço negativo, -3, indicando que esta ação não foi boa. Assim, insere-se mais uma consequência da ação do NPC apresentada na Tabela 2.

O NPC tem o objetivo de alcançar uma determinada posição do tabuleiro; porém, se o NPC passar por algum item e não recolhê-lo, ele receberá uma punição. Para esta configuração, foi acrescentado um retorno negativo para o caso de o agente não recolher algum item.

Neste grupo de testes, foram efetuados 10 testes para cada configuração, cada teste executando 20.000 interações. Os grupos foram estruturados como segue:

- Grupo 7: o ajuste é feito de forma gradual após reforços positivos para 1 e negativos para 0;
- Grupo 8: o ajuste é realizado após reforços positivos para 1 e negativos para 0;
- Grupo 9: o NPC ao receber um retorno negativo ajusta a saída da rede neural para 0 apenas para esta ação. Todavia, se o retorno é positivo. o ajuste da saída é feito para 1 da ação, e as demais ações que não foram eleitas para este estado são ajustadas como se o retorno para elas fosse negativo, ou seja, 0.

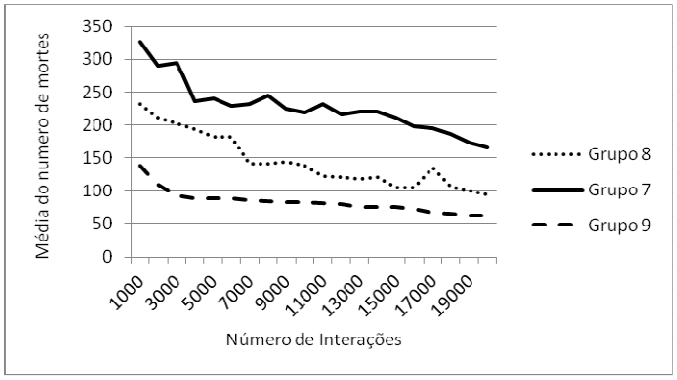

Figura 7. Médias dos grupos 7, 8 e 9

Nestes grupos de testes, ocorreu mais oscilação nas ações do personagem, e, diferente do segundo grupo de testes, tanto as ações de recolher itens como as de atacar o inimigo ficaram mais equilibradas. Observou-se, também, que o personagem demorou mais tempo para tomar uma ação ideal dependendo do estado que se encontrava.

A Figura 7 mostra o número médio de mortes para os grupos 7, 8 e 9, que teve um decaindo menor do que aquele apresentado no segundo grupo de testes, Seção IV.B. O grupo 8 obteve o melhor resultado. Nesta simulação, o NPC continuou a atacar e a recolher os itens que estavam em seu caminho.

D. Análise dos resultados das configurações adotadas

Nos testes realizados, chegou-se a diferentes resultados com as configurações adotadas nos três grupos de testes. Em relação ao número médio de mortes do NPC, o primeiro grupo de testes apresentou o pior resultado, sem mostrar nenhuma aprendizagem. O segundo grupo de testes obteve a menor média de mortes do NPC, mas depois de um determinado momento optou só por atacar. O terceiro grupo de testes apresentou um resultado mais equilibrado em relação ao número de mortes e à realização de ações distintas efetuadas pelo NPC.

V. CONCLUSÃO

A avaliação de desempenho da aprendizagem por reforço com uma rede neural para auxiliar um NPC mostrou-se satisfatória. O NPC apresentou um decréscimo no número de mortes agindo de maneira adequada nas simulações dos grupos 7 e 8. Os testes realizados também mostraram a necessidade de uma ação além dos reforços provenientes do ambiente para que o NPC possa explorar o ambiente de forma mais eficiente.

O primeiro grupo de testes mostrou-se inadequado para o modelo proposto. No segundo grupo de testes, o NPC, após um determinado tempo, optou apenas por atacar. O terceiro grupo de testes mostrou um NPC capaz de saber qual a melhor ação a tomar em um determinado estado; porém, necessitou de um quantidade maior de interações para que isto ocorresse.

O modelo proposto pode, também, ser integrado a outros métodos de adaptação, como os algoritmos genéticos; assim, pode-se adicionar novas características aos personagens. Outra questão que deve ser levada em consideração é o tempo médio em que os jogos digitais são executados, pois, de acordo com Robinson et al. [11], os adolescentes norte americanos jogam em média 9 horas por semana. Portanto, o modelo proposto neste trabalho possui espaço para ser empregado.

REFERENCIAS

[1] RUSSELL, S.; NORVIG, P. (2004) Inteligência artificial. 3. ed. Rio de Janeiro: Elsevier.

[2] BOURG, D; SEEMAN (2004) G. AI for Game Developers. O'Reilly.

[3] DALMAU, D. S. C. (2004) Core Techniques and Algorithms in Game Programming. Indianapolis: New Riders.

[4] CHARLES, D.; McGLINCHEY, S. (2004) The Past, Present and Future of Artificial Neural Networks in Digital Games. International Conference on Computer Games: Artificial Intelligence, Design and Education, Microsoft Campus, Reading, Nov 8 – 10.

[5] HOFFMAN, L. T.; SILVA, J. D. S. (2004) Um sistema de visão robótica baseada em aprendizagem por reforço. Instituto Nacional de Pesquisas Espaciais - INPE Programa de Pós-Graduação em Computação Aplicada, São José dos Campos – SP.

[6] RIBEIRO, C. H. C.; GABRIELLI, L. H.(2003) Aprendizado por reforço para times de robô. IX ENCITA, São José do Campos - SP, Outubro.

[7] GHORY, I. (2004) Reinforcement learning in board games.

[8] BIANCHI, R. A. C. (2004) Uso de Heurística para a Aceleração do Aprendizado por Reforço. Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo, São Paulo.

[9] KAELBLING, L. P.; LITTMAN, M. L.; MOORE, W. A. (1996) Reinforcement learning: A survey. Journal of Artificial Intelligence Research, p. 237-285, maio.

[10] HAYKIN, S. (2001) Redes Neurais, Princípios e prática. 2. Ed. Porto Alegre: Bookman.

[11] ROBINSON, T. et al.(2011) Violence, Sexuality, and Gender Stereotyping: A Content Analysis of Official Video Game Web Sites. Web Jornal of Mass Communication Research, 2009<http://academic.csuohio.edu/kneuendorf/c63311/Robinsonetal09.pdf >, Acesso em 27 fev.