

Modelo Computacional de Simulações Paralelas do Jogo da Vida

Elias Gabriel Amaral da Silva

dark@engcomp.ufrn.br

João Marcos

Marcel Oliveira

Depto de Informática e Matemática Aplicada, CCET, UFRN

Campus Universitário, CEP 59072-970, Natal, RN

jmarcos@dimap.ufrn.br marcel@dimap.ufrn.br

Resumo: *Este artigo descreve um modelo computacional orientado a eventos baseado no Jogo da Vida, um autômato celular. É apresentada uma aplicação do modelo, um sistema música algorítmica.*

Palavras-chave: autômato, jogo, vida, ocaml, evento, música.

Introdução

O **Jogo da Vida** é um autômato celular criado por John Horton Conway[1]. Um autômato celular simula células em um espaço e tempo discretos, por meio de regras locais simples. As células do Jogo da Vida possuem um *estado*, que varia entre **viva** e **morta**, e são organizadas em um tabuleiro infinito, bidimensional e retangular. Cada célula possui 8 **células vizinhas** ao redor e pode mudar seu estado ao longo do tempo, de acordo com suas vizinhas. A **Figura 1** ilustra uma região finita de uma simulação. As células mortas são representadas por um quadrado vazio, e as vivas por um círculo. Uma estrutura típica, chamada glider, está ilustrada nela.

O Jogo da Vida possui regras muito simples. O arranjo de todas as células em um determinado momento é chamada de **geração**. Existe uma **geração inicial**, com um arranjo arbitrário, e a partir de cada geração pode-se determinar a próxima. O futuro de cada célula depende de seu próprio estado e do número de células vizinhas vivas, de acordo com as seguintes regras:

- Cada célula com menos de dois vizinhos vivos ou mais de três morre (ou continua morta) na próxima geração.
- Cada célula com exatamente três vizinhos vivos nasce (ou continua viva) na próxima geração.
- Cada célula com exatamente dois vizinhos vivos não muda de estado na próxima geração.

A partir dessas regras é possível criar estruturas complexas, formadas por células agregadas que desempenham uma função. Por exemplo, o glider apresentado na figura 1 movimenta-se na diagonal baixo-esquerda. Isso acontece por que a cada quatro

gerações, ele faz uma cópia de si mesmo, mas deslocado uma casa na diagonal.

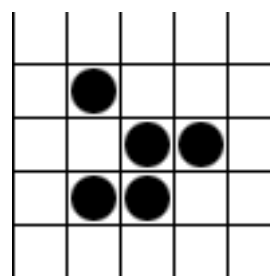


Figura 1: Glider

É possível, com estruturas adequadas de células, simular qualquer processo computacional no Jogo da Vida. Mas as aplicações que mais atraem o uso de autômatos celulares são aquelas que podem ser divididas em várias pequenas tarefas, como processamento de imagens ou simulações físicas.

Proposta

O Jogo da Vida possui uma importante limitação: toda a entrada de dados deve ser feita durante a criação de sua primeira geração; depois dela, não é possível alterar o futuro da simulação. Com isso, o Jogo da Vida não consegue interagir com o mundo externo, recebendo estímulos que pudessem mudar seu comportamento. Este trabalho se propõe a estender o Jogo da Vida através de um **ambiente computacional** baseado em **eventos**, que torna possível a utilização de **entrada e saída assíncronas**. Este ambiente é implementado como uma biblioteca escrita na linguagem de programação **OCaml**, uma linguagem de programação funcional derivada do ML.

O ambiente proposto consiste em um conjunto de simulações do Jogo da Vida executadas em paralelo, cada uma possuindo sua própria geração inicial e evoluindo de forma independente. Tais simulações serão chamadas aqui de **jogos**. Cada jogo possui um ou mais eventos associados a ele. Cada evento é **disparado** por uma determinada condição, isto é, quando certa condição é atendida, eles executam determinada ação.

Um evento pode ser disparado por 3 tipos de condições: uma **entrada externa**, um **auto-estímulo** ou um **recebimento de mensagem**. Entradas externas são dados que vêm de fora do ambiente, que podem ser processados. Condições de auto-estímulo permitem que eventos realizem ações de acordo com o arranjo de células do jogo que ele está associado. Um evento disparado por mensagens recebe mensagens de outros eventos, e executa a ação apropriada de acordo com a mensagem recebida.

Cada evento ao ser disparado, pode executar 4 tipos de ações: a **terminação** de um jogo existente, a **criação** de um novo jogo, uma **saída externa** ou o **envio de mensagem**. A terminação de um jogo existente, além de interromper a simulação, faz com que todos os eventos associados a ele desapareçam. A saída externa é o mecanismo que o ambiente usa para comunicar o resultado de suas computações ao mundo externo. Uma mensagem, quando enviada, é entregue a todos os eventos associados a determinado jogo, porém somente aqueles que estão escutando por mensagens são disparados.

A criação de um novo jogo necessita da atribuição de uma lista de eventos e a configuração inicial de células. Isso é feito pelo evento que criou o jogo. O evento que criou o jogo pode utilizar dados vindos da condição que o disparou para decidir quais os eventos e a configuração inicial do jogo criado. Ele pode utilizar, por exemplo, dados vindos de uma entrada externa ou de uma mensagem para construir o arranjo inicial de células do novo jogo. Isso permite que o sistema se modifique de acordo com entradas externas.

O ambiente possui um **jogo inicial**, com uma série de eventos associados a ele. Se o ambiente chegar a algum momento onde não existem jogos sendo executados, a computação pára.

Estudo de Caso

Como aplicação do modelo, escolheu-se a geração de **música algorítmica**, isto é, música feita inteiramente por computador. Uma forma de executar isso é, em determinados jogos, escolher regiões pequenas do jogo e tratar cada célula dessa região como a tecla de um piano ou outro instrumento musical: se estiver viva, um evento disparado por auto-estímulo toca aquela tecla, e se estiver morta ela não é tocada. Os sons são então enviados pela saída externa, onde serão interpretados por um decodificador que irá tocar de fato o som ou armazenar em disco.

O sistema de eventos atua tanto na comunicação com o mundo externo, quanto na seleção das melodias que são ou não aceitáveis: melodias muito repetitivas ou que estão com todas as células mortas, depois de um tempo são eliminadas ou combinadas com outras melodias.

Conclusão

Autômatos celulares são uma poderosa forma de expressar computações, e o Jogo da Vida permite muitas possibilidades à simulações e processamento de dados. Porém, ele não oferece uma interface com o mundo externo, o que faz com que não seja suficiente para escrever programas reais. O modelo proposto oferece maior funcionalidade ao Jogo da Vida, mantendo as suas capacidades básicas.

Referências

- [1] Martin Gardner, Mathematical Games: The fantastic combinations of John Conway's new solitaire game "life", Scientific American (Outubro de 1970). Disponível parcialmente no site http://ddi.cs.uni-potsdam.de/HyFISCH/Produzieren/lis_projekt/proj_gamelife/ConwayScientificAmerican.htm