

UFSC – CTC – INE

INE5425 - Modelagem e Simulação

RELATÓRIO

Projeto e programação de um simulador em linguagem de propósito geral

Carlos Bonetti - 12100739

Rodrigo Aguiar Costa - 12104064

Introdução

Este relatório tem o intuito de documentar, forma simples e objetiva, o projeto de Modelagem e Simulação, que envolve a simulação de um sistema de telefonia celular simplificado, como descrito nos requisitos da atividade. Para a realização do mesmo, utilizamos a linguagem JavaScript como principal ferramenta para a realização de tais funções. Algumas das classes utilizadas serão descritas a seguir.

Descrição da Solução

Para o desenvolvimento desta aplicação, utilizamos a estratégia de simulação orientada a eventos. Ou seja, a simulação dispõe de um único calendário de eventos e todas as ações que possam ocorrer na simulação são descritas por eventos. Os eventos são gerados no decorrer da simulação e postos no calendário de eventos. Estes eventos são processados a realizam ações sobre as entidades e recursos do sistema:

Entidade: Chamada

A única entidade do sistema é a “Chamada”. A chamada guarda os atributos referentes a uma chamada telefônica: tempo de início, duração e tipo.

Chamadas são geradas a partir das variáveis aleatórias e distribuições de probabilidade informadas pelo usuário.

Recurso: Célula

Cada célula contém um número de canais de comunicação disponíveis, informadas pelo usuário. As chamadas alocam um canal de comunicação da célula para serem executadas.

A cada iteração da simulação, um novo evento do calendário é processado. O evento escolhido para ser processado é o próximo evento agendado para ocorrer, de acordo com o tempo atual da simulação e o tempo de chegada do evento. Eventos são inseridos e mantidos no calendário na ordem de seu tempo agendado.

Ao ser processado, o evento em questão pode gerar outros eventos e atualizar as estatísticas.

Os seguintes eventos foram mapeados para esta aplicação:

Evento “Iniciar simulação”

O evento de início é o primeiro evento a ser inserido no calendário, logo que a aplicação é iniciada. Desta forma, o evento é agendado para ocorrer no tempo zero de simulação.

Ao ser processado, o evento de início realiza as seguintes ações:

1. Sorteia uma chamada com base nas distribuições aleatórias inseridas pelo usuário e insere no calendário um evento “Nova Chamada” atrelada a esta chamada.

Evento “Fim de simulação”

O evento de fim de simulação é inserido no momento de inicialização da aplicação, junto com a inserção do evento “Iniciar simulação”. O evento é agendado para ocorrer, porém, no tempo final da simulação, informado pelo usuário. O evento de “Fim de simulação” realiza as seguintes ações:

1. Finaliza a simulação.

Evento “Nova chamada”

O primeiro evento “Nova chamada” é inserido no calendário da simulação pelo evento “Início de simulação”. A cada processamento de um evento “Nova chamada”, uma nova chamada é gerada e um evento respectivo é inserido no calendário.

Cada evento “Nova chamada” possui uma entidade “Chamada” atrelada a ele, que contém as informações de tempo de duração, tempo de chegada, tipo da chamada, célula atual etc.

As ações realizadas por este evento, portanto, são as seguintes:

1. Sorteia uma chamada com base nas distribuições aleatórias inseridas pelo usuário e insere no calendário um evento “Nova Chamada” atrelada a esta chamada;
2. Se a célula da chamada atrelada a este evento estiver disponível (possuir pelo menos um canal de comunicação):
 - a. Incrementa o número de chamadas no sistema
 - b. Aloca um canal de comunicação da célula origem da chamada
 - c. Se esta chamada for do tipo que “se desloca” (ex: tipo C1C2, C2C1, ...)
 - i. Lança um novo evento “Migração de chamada” atrelada a esta chamada e agendada para ocorrer no tempo atual + metade da duração desta chamada.

Senão

- ii. Lança um novo evento “Fim de chamada” atrelada a esta chamada e agendada para ocorrer no tempo atual + duração desta chamada.

Senão

- d. Incrementa o contado de chamadas perdidas da célula

Evento “Migração de chamada”

O evento “Migração de chamada” representa o deslocamento realizado por chamadas do tipo “C1C2”, “C2C1”, “C1FA” e “C2FA”. Este evento é lançado ao processar um evento “Nova chamada” atrelada a alguma chamada dos tipos citados e é responsável por migrar a chamada da célula origem à célula destino.

As ações realizadas por este evento são:

1. Desaloca a célula origem da chamada associada a este evento;
2. Se a célula destino estiver disponível:
 - a. Aloca um canal de comunicação da célula destino;
 - b. Lança um novo evento “Fim de chamada” associada a esta chamada e agendada para ocorrer no tempo de fim desta chamada

Senão:

- c. Contabiliza o tempo de duração de chamada até o momento;
- d. Incrementa o contador de chamadas perdidas por falta de canal da célula destino;
- e. Retira a chamada do sistema

Evento “Fim de chamada”

Este evento é responsável por finalizar uma chamada, contabilizar suas estatísticas e retirá-la do sistema. Este evento é lançado tanto pelos eventos “Nova chamada” quando pelo “Migração de chamada”.

As ações realizadas por este evento são:

1. Desaloca um canal de comunicação da célula atual da chamada atrelada a este evento;
2. Contabiliza a chamada e seu tempo de duração;
3. Retira a chamada do sistema.

Com estes 5 eventos temos toda a lógica da simulação completa: a simulação é iniciada com os eventos “Início de simulação” e “Fim de simulação” somente e começa a processar

os eventos do calendário em ordem. Ao processá-los, novos eventos aleatórios são gerados e a simulação continua até atingir o “Fim de simulação”.

Descrição das classes / entidades

Em **simulation.js** possuímos como principais classes a classe *Simulation* e a classe *Calendar*, onde este último representa nosso calendário de eventos que tem como principais métodos a inserção de eventos no calendário, próximo evento e número de eventos que o calendário possui. *Simulation* é a classe que inicializa o calendário de eventos e o tempo de simulação do sistema, entre outras variáveis necessárias para a geração das estatísticas do sistema. Este também possui métodos de início de eventos, processamento de próximo evento, geração de nova chamada com sorteio de valores. Além de possuir valores e métodos que são usadas para a geração das estatísticas.

Em **events.js** possui como principal classe *EventSim* onde este é a classe base para os eventos da simulação, possuindo diversas subclasses que são caracterizadas como eventos:

- de início da simulação
- de fim da simulação
- de nova chamada
- de fim da chamada
- migração de chamadas

Para as distribuições de probabilidade criamos em **distribution.js** a classe principal de *Distribution* e criamos classes que a estende sendo elas as distribuições constante, uniforme, triangular, exponencial e normal.

Para as representações das células de cobertura da chamada telefônica e as próprias chamadas telefônicas foram criadas tais classes em **telephony.js**. Para as células de cobertura criamos a classe *Cell* que possui como principais métodos a verificação de disponibilidade de canais, alocação e liberação de canais, além de armazenar a quantidades de chamadas que foram perdidas devido a limitação do número de canais de comunicação e outras variáveis para a geração das estatísticas. Em *Call* criamos os métodos referentes as chamadas telefônicas em si, como se ela é uma chamada que migra, seu tempo de duração, tempo de início da chamada, entre outros.