

# Explorando Concorrência em Simulação de Humanos Virtuais\*

Otávio Corrêa Cordeiro,  
Soraia R. Musse, Gerson Geraldo H. Cavalheiro

Programa Interdisciplinar de Pós-Graduação em Computação Aplicada  
Universidade do Vale do Rio dos Sinos  
Av. Unisinos, 950 - Bairro Cristo Rei - CEP 93.022-000 São Leopoldo - RS - Brasil  
otaviocc@turing.unisinos.br, {soraiaarm, gersonc}@unisinos.br

## Resumo

A grande concentração de indivíduos em lugares públicos é característica crescente em grandes centros urbanos, tornando comum aglomerados em diferentes situações, como em estações de transporte público, bancos ou centros comerciais. Este fato, aliado ao constante movimento das pessoas nestas situações torna o estudo de simulação de multidões desejado, seja para assegurar conforto ou garantir segurança à multidão.

Torna-se desafiador reproduzir este tipo de simulação em tempo real, uma vez que o número de *agentes* – representando indivíduos no sistema simulado – é grande, assim parâmetros do sistema real simulado. Desta forma uma implementação paralela, baseada em regras e leis física é apresentada, permitindo a simulação realística de milhares de agentes.

Na literatura, inúmeros modelos vêm sendo propostos com o objetivo de simular o comportamento de multidões com diferentes propósitos práticos, porém, na grande maioria, a abordagem é feita sobre um sistema de partículas, onde a interação é tratada com formalismo físico. No modelo proposto por Helbing [HEL 00], forças física e sócio-psicológicas são aplicadas para melhor representar o comportamento de multidões de humanos, em particular, situações de pânico. Seu modelo é composto por um sistema de partículas, onde cada partícula  $i$ , de massa  $m_i$  possui uma velocidade pré-estabelecida  $v_i^0$ , um vetor direção  $\hat{e}_i^0$  que define o objetivo do agente, normalmente direcionado ao local de evacuação, e sua velocidade instantânea  $\vec{v}_i$ , que tende a se adaptar na direção do objetivo em um intervalo de tempo  $\tau_i$ .

A simulação evolui através do tempo de forma *discreta*, sendo  $\vec{F}_i^H$  (Eq. 1) computado em cada passo de simulação (*timestep*), envolvendo o cálculo da força de repulsão entre agentes distintos, paredes e obstáculos. A proximidade entre dois agentes – exemplo claro quando próximos do objetivo – favorece a erros de integração, forçando o uso de *timesteps* relativamente pequenos, tornando a simulação custosa computacionalmente.

$$m_i \frac{d}{dt} \vec{v}_i = \vec{F}_i^H = m_i \frac{v_i^0 \hat{e}_i^0 - \vec{v}_i(t)}{\tau_i} + \sum_{j \neq i} \vec{f}_{ij} + \sum_w \vec{f}_{iw} \quad (1)$$

Quinn [QUI 03] propôs uma implementação para o modelo de Helbing explorando

---

\*Este trabalho foi desenvolvido em colaboração parcial com a HP Brasil R&D.

o esquema mestre/escravo para controlar a evolução da simulação. Nesta implementação é explorado o paralelismo de dados da aplicação, com os mestres manipulando uma lista de *zonas* – segmentos espaciais à serem simulados –, enquanto enviam estas para os escravos nos nodos de processamento.

Este trabalho propõe a implementação concorrente, considerando características específicas da multidão em situação de pânico. A partir da divisão do espaço físico em células de tamanho fixo é possível reduzir a área de atuação das forças de cada agente, tornando a cobertura da força de pelo menos uma célula, podendo expandir em até três células. Além de reduzir o custo computacional, esta abordagem possibilita a execução sobre todas as células da simulação de forma concorrente.

A unidade de trabalho – tarefa – é definida, então, em termos de células e, embora possuam a mesma dimensão, apresentam diferentes cargas computacionais (carga de natureza irregular), dado que no interior de cada célula possam existir quantidades distintas de agentes e/ou obstáculos. Cada *timestep*  $t$  possui um estado  $S_t$  com informação das células e evolui esse conjunto para um estado  $S_{t+\Delta t}$ .

Para explorar o problema ao máximo, a paralelização é vista em dois níveis distintos: *i*) um escalonamento de tarefas intra-nodo, fazendo uso de *multithreading* – que possui um conjunto  $S_t$  de células armazenados em uma lista, sendo consumidas por um conjunto de *threads* que estão dedicadas a execução dos cálculos das forças; e *ii*) um escalonamento entre-nodos, que leva em consideração a carga entre os nodos. Uma distribuição estática de carga é feita em um estágio inicial, dividindo o conjunto de  $c$  células em  $p$  domínios, onde  $p$  é o número de máquinas no agregado de computadores. Cada nodo recebe  $c/p$  células, introduzindo a necessidade do uso de troca de mensagens entre os nodos para o cálculo de forças entre células vizinhas em domínios distintos. Cada domínio é capaz de evoluir no *timestep* sua solução interna, ou seja, de células que independem dos resultados de células de outros domínios, e, em uma segunda etapa, o cálculo das bordas enquanto seus vizinhos resolvem o passo anterior, de forma compassada.

O balanceamento de cargas é feito na execução distribuída, onde parte do *overhead* gerado pela comunicação entre os nodos é sorvido pelo cálculo das forças em dois estágios distintos [VAL 90]. Durante a execução, os agentes migram de um nodo para outro e a carga torna-se irregular. Neste ponto, cada domínio é reorganizado mantendo coerente o trabalho *per nodo* da simulação. Esta estratégia mostra-se particularmente interessante em situações de pânico, onde agentes tendem a agrupar-se em um único foco.

Desta forma o trabalho apresenta o modelo de implementação em arquiteturas paralelas onde é dado destaque à importância de identificação da unidade de trabalho para obter uma estratégia de escalonamento eficiente, objetivando a simulação realística de uma multidão em situações de pânico.

## Referências

- [HEL 00] HELBING, D.; FARKAS, I.; VICSEK, T. **Simulationg dynamical features of escape panic**. Nature, v.407, p.487-490, 2000.
- [QUI 03] QUINN, M. J.; METOYER R. A.; ZAWORSKI, K. H. **Parallel implementation of the social forces model**, Int. Conf. in Pedestrian and Evacuation Dynamics, 2003.
- [VAL 90] VALIANT, L. G. **A bridging model for parallel computation**, Commun. ACM, v.33, n.8, p.103111, 1990.