

APRESENTAÇÃO DE RESULTADOS DA EQUIPE AKUANDUBA-UDESC NO MAPC 2018, DOS BOLSISTAS DO PROJETO DE PESQUISA COOMAS

Guilherme Rafael Deschamps¹, Tiago Luiz Schmitz²

¹ Acadêmico do Curso de Engenharia de Software (CEAVI) – bolsista PIPES/UDESC

² Orientador, Departamento de Engenharia de Software (CEAVI) – tiago.schmitz@udesc.br

Palavras-chave: Multi-agentes. MAPC. Competição.

Em 2018 durante o projeto Coomas participamos da competição MAPC (*Multi-Agent Programming Contest*) e da escrita e tradução do artigo “*A task-oriented architecture with priority queue for BDI agents applied to the Multi Agent Programming Contest scenario*” (Uma arquitetura orientada a tarefas com fila de prioridades para agentes BDI aplicada ao cenário MAPC) submetido para a editora Springer. Para o presente artigo, a pesquisa se classifica como descritiva, voltada para um contexto específico, considerando que é destacado como o principal objetivo do artigo a apresentação dos resultados adquiridos na competição MAPC no ano de 2018.

A competição

No MAPC (Concurso de Programação de Multi-Agentes) são apresentados problemas-chaves às equipes participantes, que devem identificar e desenvolver possíveis soluções apresentando casos de testes e referências adequadas. Na edição de 2018, a cada rodada dois times competiam simultaneamente com 34 agentes cada (divididos entre carros, motos, caminhões e drones) e disputavam por 3 rounds na obtenção de pontos e dinheiro, buscando manter com os veículos autônomos uma cidade constantemente com dinheiro e água. Os pontos eram adquiridos através dos poços de água que deveriam ser construídos pelos times, e o dinheiro era obtido ao serem completados trabalhos e missões, podendo ser investido em upgrades para os veículos ou em novos poços de água.

Akuanduba-UDESC

Desenvolvemos os agentes utilizando a linguagem Jason. A comunicação entre os agentes e o servidor utilizava EISMassim, e para a comunicação entre os agentes (que não eram executados todos na mesma máquina) foi utilizado o JADE. Além disso, todos os nossos agentes possuíam roles, que eram gerenciadas e distribuídas pelo artefato CArtAgO. Para o processo de decisão de nossos agentes, foi implementada uma fila de prioridades. Para facilitar a contabilização do tempo ao longo da simulação, passados alguns segundos a simulação avançava um step. A cada step que avançava, a árvore de possíveis decisões que o agente poderia optar por seguir era atualizada, e a decisão selecionada era sempre a com a maior prioridade. Apresentando um exemplo simples: no contexto dos veículos autônomos na cidade, todos os veículos possuíam bateria. Sendo assim, quando um veículo tivesse pouca bateria a atividade de prioridade máxima na árvore de decisões seria ir ao posto mais próximo para recarregar. Quando tivessem a bateria recarregada, de acordo com suas prioridades os veículos podiam sair para executar a coleta de itens básicos, ou responder a pedidos de ajuda recebidos de outros agentes para a criação de itens compostos.

Resultados

Quanto aos resultados da competição, perdemos para os 4 times contra quem competimos, porém, na combinação de tecnologias e técnicas utilizadas apresentamos desempenho satisfatório.

Devido à estratégia utilizada para a entrega de jobs (trabalhos), nosso time se manteve com dinheiro durante todas as partidas da competição, e os agentes encarregados de coletar itens mantiveram o estoque cheio por toda a simulação. Porém, perdemos todos os jogos competidos devido à fraca estratégia que utilizamos no investimento do dinheiro para a criação de poços de água.

O sistema foi desenvolvido de forma a ser extremamente simples adicionar comportamentos aos agentes. Uma classe funcionava como um planejador e gerenciador de tarefas, com vários import contendo diferentes estratégias para as atividades. O desenvolvimento do planejador de tarefas e gerenciador de tarefas foi trabalhoso, porém foi recompensador ao facilitar muito os testes para a comparação de desempenho. As lógicas para as diferentes tarefas ficavam todas nas classes importadas, permitindo em questão de segundos modificar blocos de código extremamente importantes.

Concluo afirmando que nossa experiência no Multi-Agent Programming Contest foi uma experiência recompensadora. Utilizamos a oportunidade para competir e ao mesmo tempo conhecer diferentes universidades, e apesar de termos tido uma colocação ruim no campeonato nosso esforço não foi em vão. A maioria das estratégias utilizadas se mostraram eficientes, nossa comunicação funcionou sem problemas, mesmo ao distribuir nossos agentes em 3 máquinas, e ainda nossas lógicas de coleta de itens e entrega de trabalhos se mostrou funcional e eficaz.

Referências

Developing Mult Agent Systems with JADE. Disponível em: <<https://www.mit.bme.hu/system/files/oktatas/targyak/8671/Developing-Multi-Agent-Systems-with-JADE.pdf>>. Acesso em: 31 jul. 2019.

The 2018 Contest. Disponível em: <<https://multiagentcontest.org/2018/>>. Acesso em: 31 jul. 2019.

About CARTAgO. Disponível em: <<http://cartago.sourceforge.net/>>. Acesso em: 31 jul. 2019