

UTILIZANDO O SIMULADOR V-REP PARA SIMULAÇÕES COM MÚLTIPLOS AGENTES

Lucas Antunes de Almeida¹

Lucas Almeida²

Guilherme Neri Bustamante Sá³

Marcelo Resende Thielo⁴

Resumo:

Ferramentas de simulação permitem que o usuário desenvolva aplicações reais sem terem de se preocupar com aspectos relativos à modelagem explícita do ambiente, como por exemplo a força da gravidade ou o atrito entre dois objetos, além de fornecerem a possibilidade de visualização prévia de como um determinado agente pode vir a se comportar em uma situação real. O presente trabalho relata experiências relacionadas ao simulador de robôs V-REP, utilizado para verificar o comportamento de múltiplos agentes em simulações, apresentando em especial o uso da ferramenta em conjunto com a API Remota disponibilizada junto ao simulador. Os resultados apresentados agrupam simulações simples que buscam demonstrar de forma clara e simplificada algumas das características que tornam o uso da API algo relativamente simples. Como trabalhos futuros vislumbra-se um estudo mais aprofundado com foco nos sensores e atuadores disponibilizados por padrão na própria ferramenta, assim como a implementação de uma interface gráfica de controle interna ao simulador para controle direto da simulação.

Palavras-chave: V-REP, Simulador, Robôs

Modalidade de Participação: Iniciação Científica

UTILIZANDO O SIMULADOR V-REP PARA SIMULAÇÕES COM MÚLTIPLOS AGENTES

¹ Aluno de graduação. lucas.lukinha67@gmail.com. Autor principal

² Aluno de graduação. lucas.lukinha67@gmail.com. Apresentador

³ Aluno de graduação. guinbsa@gmail.com. Co-autor

⁴ Docente. marcelothiello@unipampa.edu.br. Orientador

UTILIZANDO O SIMULADOR V-REP PARA SIMULAÇÕES COM MÚLTIPLOS AGENTES

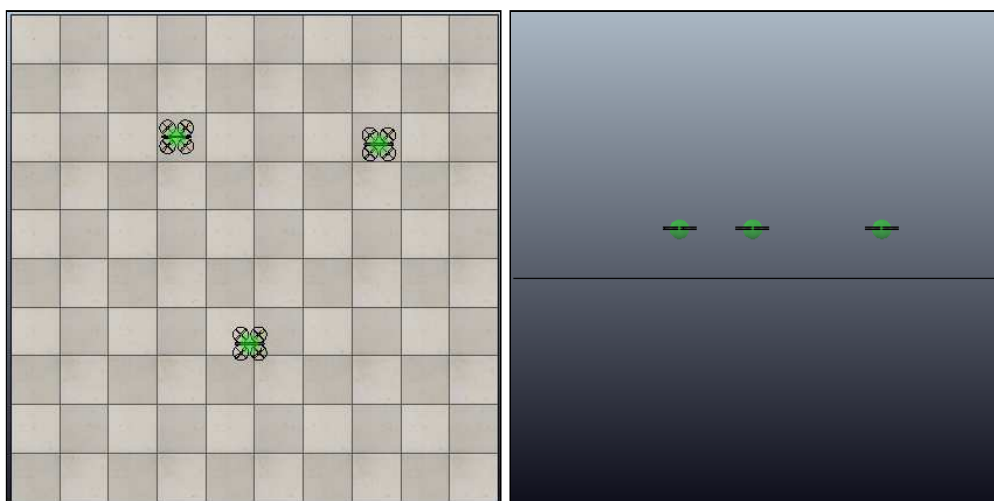
1 INTRODUÇÃO

Com o avanço da computação nos dias atuais é possível verificar a presença de sistemas robóticos precisos em nosso cotidiano. Os simuladores têm papel importante neste aspecto, pois sua facilidade de uso e realismo conseguem gerar resultados próximos aos reais e até mesmo em alguns casos, controlar remotamente sistemas inteiros. Um desses simuladores é o V-REP. [ROHMER S. P. N. SINGH, 2013]

O V-REP desempenha um ótimo papel quando utilizado em simulações relacionadas à robótica que devem conter aspectos físicos, como colisões ou gravidade. Outro aspecto importante na ferramenta é a possibilidade de alterações do código dos agentes da simulação no próprio simulador, utilizando *plugins* desenvolvidos pela própria comunidade, o que facilita muito, pois não é preciso começar um trabalho inteiramente do zero. [SILVA; OLIVEIRA, 2015]

Assim como pode ser visto na Figura 1, o simulador permite ainda a visualização do espaço simulado por ângulos distintos o que torna a tarefa de análise de comportamento de agentes algo significativamente mais simples, tornando o uso da ferramenta algo interessante quando utilizada em atividades ligadas ao aprendizado.

Figura 1- Drones no V-REP vistos de ângulos distintos.



Fonte: do autor, 2018.

Vale ressaltar que, ao se utilizar a ferramenta como uma plataforma de testes, é possível averiguar se uma ação correspondente a um agente pode ou não vir a funcionar em uma máquina real, o que por sua vez agrega um valor significativo ao uso do software em situações ligadas ao ensino ou até mesmo a aplicações de pesquisa ou comerciais. O V-REP é um software de uso gratuito para finalidades acadêmicas mas, para uso comercial existe uma versão paga.

O simulador conta com um ambiente de desenvolvimento integrado que, por sua vez, é baseado em uma arquitetura de controle distribuída onde cada objeto ou cada modelo pode ser controlado individualmente através de um *script* embutido [KERSCHBAUMER; LIMA; SIMÃO, 2014], o que torna o V-REP muito versátil e ideal para aplicações com múltiplos agentes. Por estes motivos, e por ter uma versão gratuita para estudantes, o V-REP foi escolhido como simulador utilizado para realizar simulações envolvendo múltiplos drones em diferentes rotas simultâneas,

geradas manualmente e passadas ao V-REP por um algoritmo externo na linguagem C++. Não necessariamente é obrigatório utilizar a sua Interface de Programação de Aplicações (API) para este fim, mas para quesito de estudo que servirá para trabalhos futuros esta foi a abordagem utilizada.

Neste trabalho relatamos nossa experiência levando em conta as facilidades e dificuldades enfrentados com o simulador e o que aprendemos utilizando sua API externa utilizada para fazer a comunicação entre os nosso códigos e o simulador.

2 METODOLOGIA

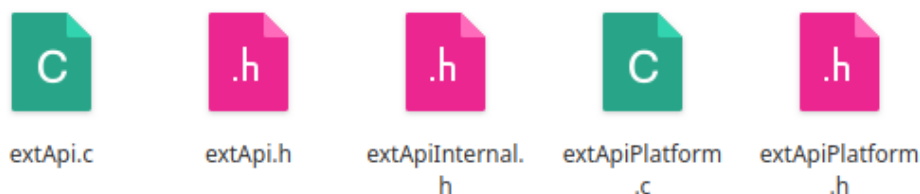
Para o estudo realizado, utilizamos um ambiente Linux executando o simulador em conjunto com uma API externa disponibilizada pela própria empresa dona do V-REP. Esta API é utilizada para que códigos feitos em C++ sejam executados em conjunto com o simulador fazendo a comunicação entre eles.

2.1 API Remota

O V-Rep oferece uma API que permite controlar uma simulação ou o próprio simulador de um aplicativo ou *hardware* externo, podendo ser chamada através de alguma aplicação C, C++, python 3, java, Matlab, Octave ou lua.

As funções presentes na API interagem com o V-REP usando uma comunicação baseada em *sockets*, e tudo acontece de maneira oculta ao usuário. A API Remota pode permitir que um ou mais aplicativos externos interajam com o V-REP, onde até mesmo o controle remoto do simulador é suportado, o que permite por exemplo iniciar ou pausar uma simulação de uma máquina distinta à que está executando simulação propriamente dita.

Figura 2- Arquivos da API Remota.



Fonte: do autor, 2018.

A utilização das funções pode ser realizada apenas depois da inclusão de arquivos disponibilizadas dentro da pasta do próprio simulador. Estes arquivos podem ser vistos na Figura 2, os quais representam a inclusão da API para C ou C++.

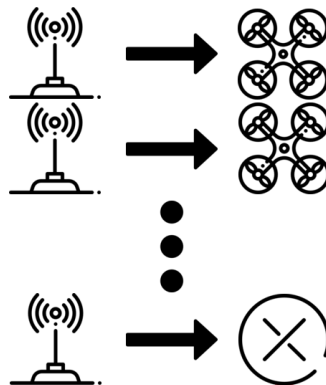
2.2 Comunicação Entre o V-REP e o Código

Para que seja possível utilizar o simulador de maneira simples, foi desenvolvida uma classe que tem atribuída a si uma função de gerenciar a comunicação e movimentação de múltiplos agentes de forma simples, podendo-se assim abstrair certos comandos utilizados pela API.

Assim como foi ilustrado na Figura 3, ao instanciar um objeto da classe de controle, a mesma trata de verificar quantos drones existem. Como os drones são numerados sequencialmente, apenas se faz necessário buscar pelos agentes da simulação (neste caso, os drones) até receber uma

resposta inválida da API, a qual sinaliza que não existem mais drones na simulação que deveriam ser adicionados na execução do algoritmo em C++.

Figura 3- Inicialização dos drones no V-REP.



Fonte: do autor, 2018.

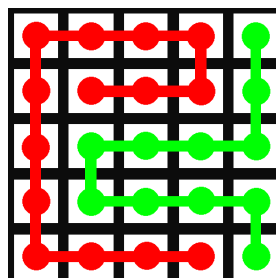
Como pode ser observado na Figura 3, a ferramenta permite que o usuário solicite informações referentes a simulação que está em vigor, geralmente as informações solicitadas tendem a serem próximas a variáveis de ambientes, como por exemplos, valores de velocidade ou de atrito, entretanto no presente trabalho as informações utilizadas foram coordenadas, posições e se o objeto em questão existia ou não na simulação.

2.2 Rotas da Simulação

As rotas foram geradas manualmente a partir de uma matriz de coordenadas tendo como base um plano cartesiano com o centro no ponto (0, 0). Como demonstrado na Figura 4, as rotas são divididas pela matriz ocupando cada posição dela. Cada rota é representada como um vetor de coordenadas reais dentro do algoritmo.

Após descobrir o número exato de drones da simulação, o algoritmo passa as coordenadas das rotas divididas entre os drones utilizando o objeto da classe de controle, através das funções da própria API.

Figura 4- Exemplo de Rota para a Simulação.



Fonte: do autor, 2018.

Nem todas as funções do V-REP estão disponíveis pela API e por este motivo foram usadas função que movem o “alvo” do drone pela simulação. Nesta função o drone simplesmente recebe um lugar destino e move-se para ele com uma velocidade suficiente para alcançá-lo.

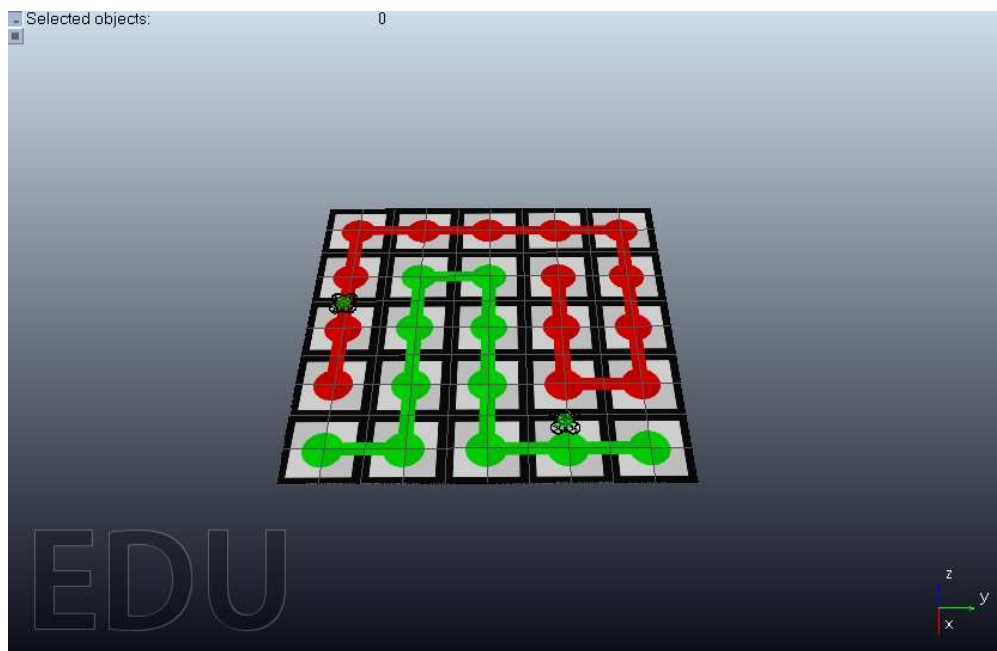
3 RESULTADOS e DISCUSSÃO

Utilizando o simulador V-REP nota-se que o mesmo torna simulações envolvendo veículos como drones, robôs móveis terrestres, braços manipuladores ou robôs de outra natureza algo simples de se realizar, já que não se faz necessária a preocupação com aspectos como a física do ambiente, colisão entre agentes ou energia utilizada, ou seja, somente é necessário preocupar-se com aspectos relativos ao planejamento e controle de ações dos agentes.

É possível ver na Figura 5 a simulação realizada para este trabalho. Nela os dois drones utilizados seguem um caminho pré-estabelecido no algoritmo em C++, seguindo coordenada a coordenada das rotas. Para facilitar a visualização uma textura foi adicionada ao chão do simulador.

Um problema encontrado neste trabalho foi a dificuldade de utilizar comandos que ajustam a velocidade do drone, quando este está indo em direção ao alvo desejado. A função implementada para mover o drone somente faz com que ele se mova para o ponto usando a velocidade suficiente para isto, porém quando o alvo está muito distante do drone a velocidade usada se torna maior que a necessária, fazendo assim com que o drone passe do ponto de destino e talvez até colida com outros drones.

Figura 5 - Apresentação dos drones em movimento no simulador V-REP..



Fonte: do autor, 2018.

4 CONSIDERAÇÕES FINAIS

Apesar de termos encontrado alguns obstáculos na utilização do simulador, este se mostra um ferramenta muito interessante a ser explorada sempre que o objetivo seja criar um algoritmo

realmente funcional em robôs. As facilidades implementadas por esta ferramenta podem inclusive evitar o desperdício de recursos em um projeto, pois o simulador consegue entregar resultados próximos aos reais, proporcionando uma maneira de fazer todos os testes possíveis antes de realmente usar componentes reais.

Algumas dificuldades se deram em relação a documentação da API utilizada que, embora consiga fazer a comunicação perfeitamente com o algoritmo em C++, ainda não tem todas as opções que se pode alcançar utilizando o V-REP pela interface gráfica normalmente, sem a API. Isto não se torna um grande problema para simulações, pois frequentemente é possível utilizar a API e os recursos no V-REP ao mesmo tempo.

A documentação da ferramenta é vasta e disponibiliza quase tudo o que o usuário precisa saber para realizar o seu projeto e mesmo que algo não esteja na documentação, um fórum é disponibilizado para tirar dúvidas, sendo que na maioria das vezes a própria Coppelia Robotics responde.

Por fim, como trabalhos futuros, um estudo mais aprofundado na ferramenta com foco nos sensores e atuadores disponibilizados deve ser feito. Como a implementação de interface gráfica para o controle do simulador também é possível, será feito um estudo sobre a biblioteca gráfica QT [QT, 2018] para implementá-la.

REFERÊNCIAS

KERSCHBAUMER, R.; LIMA, C. R. E.; SIMÃO, J. M. O simulador de robôs v-rep no ensino das técnicas de controle de robôs autônomos. XLII COBENGE: UFJF - JUIZ DE FORA/MG, 2014. Disponível em: <https://www.researchgate.net/profile/Ricardo_Kerschbaumer/publication/269093613_O_SIMULADOR_DE_ROBOS_V-REP_NO_ENSINO_DAS_TECNICAS_DE_CONTROLE_DE_ROBOS_AUTONOMOS/links/55d1ae6808ae95c3504d57ea/O-SIMULADOR-DE-ROBOS-V-REP-NO-ENSI-NO-DAS-TECNICAS-DE-CONTROLE-DE-ROBOS-AUTONOMOS.pdf>. Acesso em: 08 set. 2018.

ROHMER S. P. N. SINGH, M. F. E. V-rep: a versatile and scalable robot simulation framework. In: Proc. of The International Conference on Intelligent Robots and Systems (IROS). [S.l.: s.n.], 2013. Disponível em: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6696520>>. Acesso em: 08 set. 2018.

SILVA, R. T. da; OLIVEIRA, S. C. Uma proposta de utilização do simulador v-rep como um laboratório virtual em disciplinas do ciclo básico dos cursos de engenharia. XLIII Congresso Brasileiro de Educação em Engenharia, 2015. Disponível em: <https://www.researchgate.net/profile/Ricardo_Silva27/publication/281785768_UMA_PROPOSTA_DE_UTILIZACAO_DO_SIMULADOR_V-REP_COMO_UM_LABORATORIO_VIRTUAL_E_M_DISCIPLINAS_DO_CICLO_BASICO_DOS_CURSOS_DE_ENGENHARIA/links/55f9b77908aeafc8ac27f3ed/UMA-PROPOSTA-DE-UTILIZACAO-DO-SIMULADOR-V-REP-COMO-UM-LABORATORIO-VIRTUAL-EM-DISCIPLINAS-DO-CICLO-BASICO-DOS-CURSOS-DE-ENG ENHARIA.pdf>. Acesso em: 08 set. 2018.

The Qt Framework, Setembro de 2018, [online]. Disponível em: <www.qt.io/qt-framework/> Acesso em: 08 set. 2018.