

Aprendizado de Técnicas de Geometria Computacional e Aplicações à Robótica

Luciano Bracht Barros - bracht.barros@ufrgs.br
Waldir Leite Roque (orientador) - roque@mat.ufrgs.br

Objetivo

O presente trabalho tem como objetivo o desenvolvimento de métodos e algoritmos para a identificação da configuração do espaço livre de colisões, utilizando a Soma de Minkowski.

Introdução

A pesquisa que está sendo desenvolvida consiste em desenvolver um algoritmo para “somar” figuras geométricas, usando para isso a Soma de Minkowski. Essa soma pode ser vista, didaticamente, como a translação de uma figura ao longo das bordas de uma outra figura. Essa ferramenta é útil para somar a figura de um robô às figuras que o rodeiam, obtendo, assim, o espaço livre de colisão.

Soma de Minkowski

Definição: Sejam A o conjunto de todas as coordenadas dos pontos da Figura A e B o conjunto de todas as coordenadas da Figura B. Então:

$$A \oplus B = \{a + b : a \in A, b \in B\}$$

Ou seja, todos os pontos da figura resultante da soma $A \oplus B$ podem ser expressos como a soma de algum elemento de A com algum elemento de B.

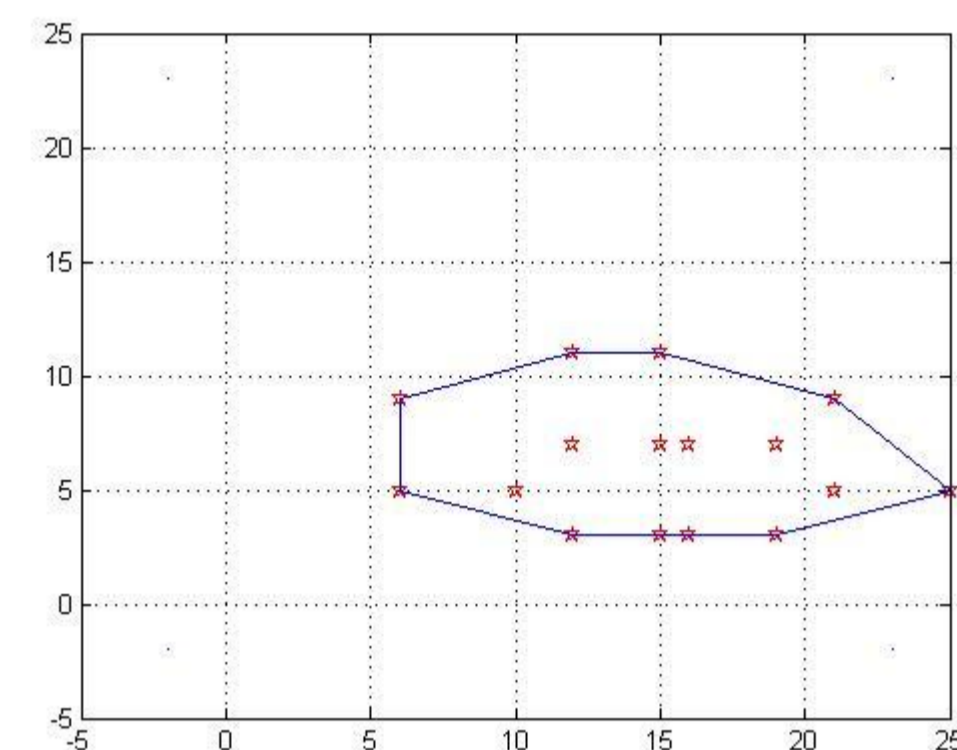
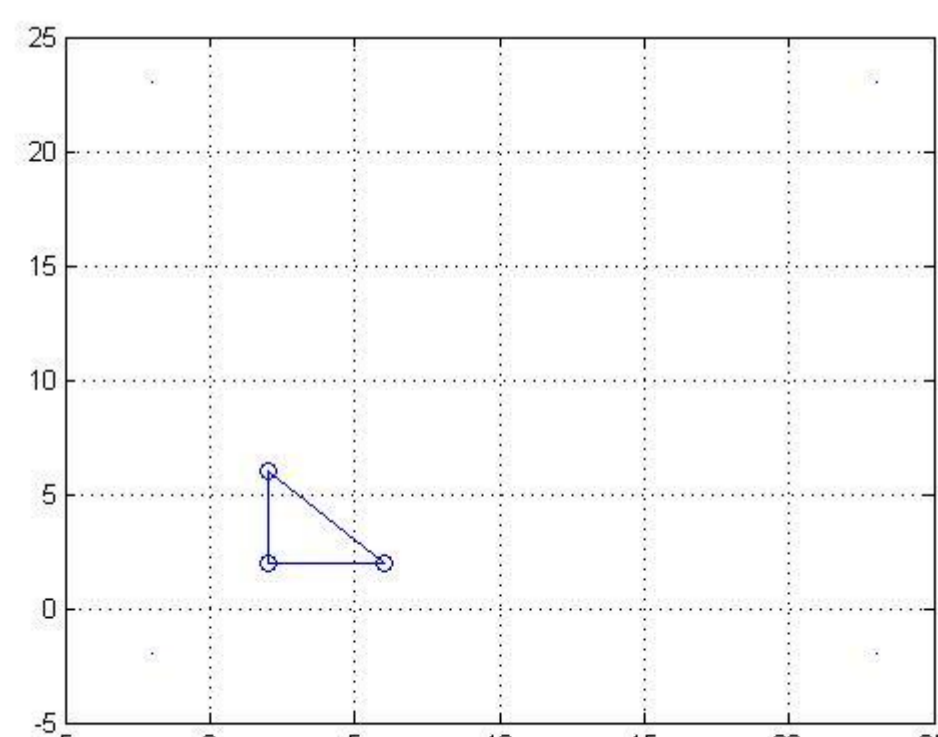
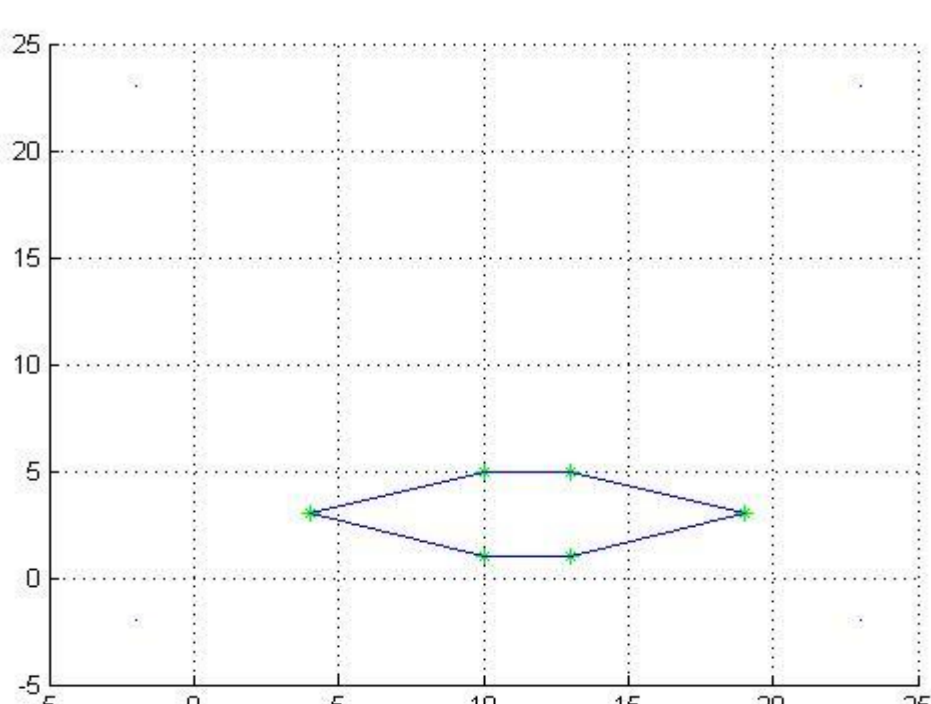
Resultados obtidos pelo algoritmo

Sejam duas Figuras convexas A e B tais que os vértices das figuras são:

$$A = [4 \ 10 \ 10 \ 13 \ 13 \ 19 ; 3 \ 1 \ 5 \ 1 \ 5 \ 3]$$

$$B = [2 \ 2 \ 6 ; 2 \ 6 \ 2]$$

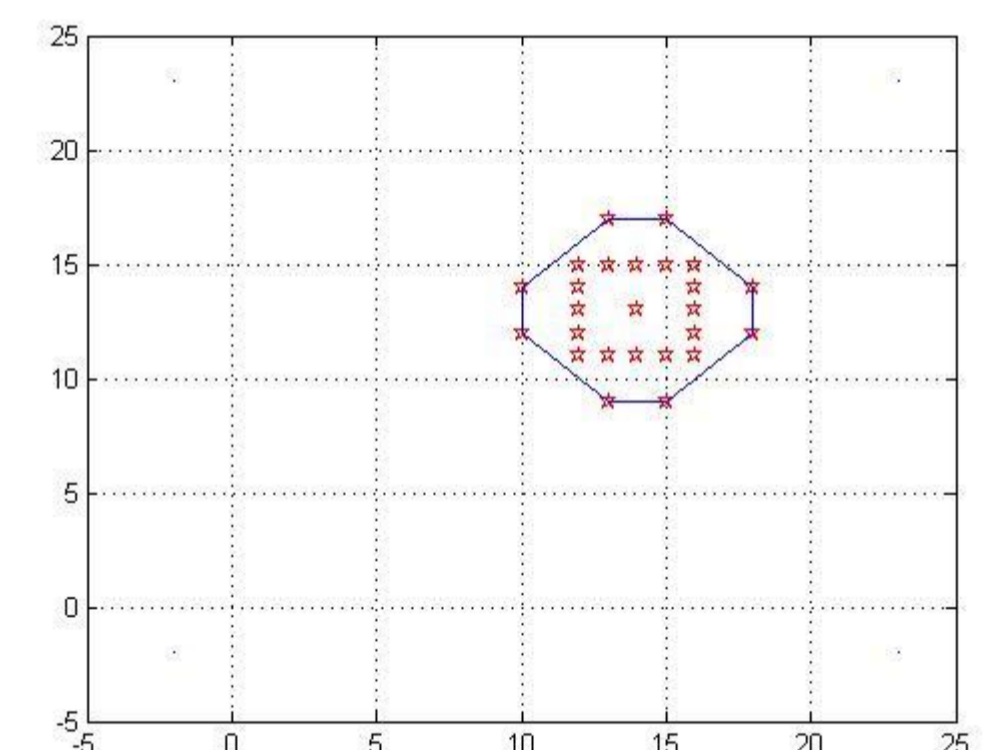
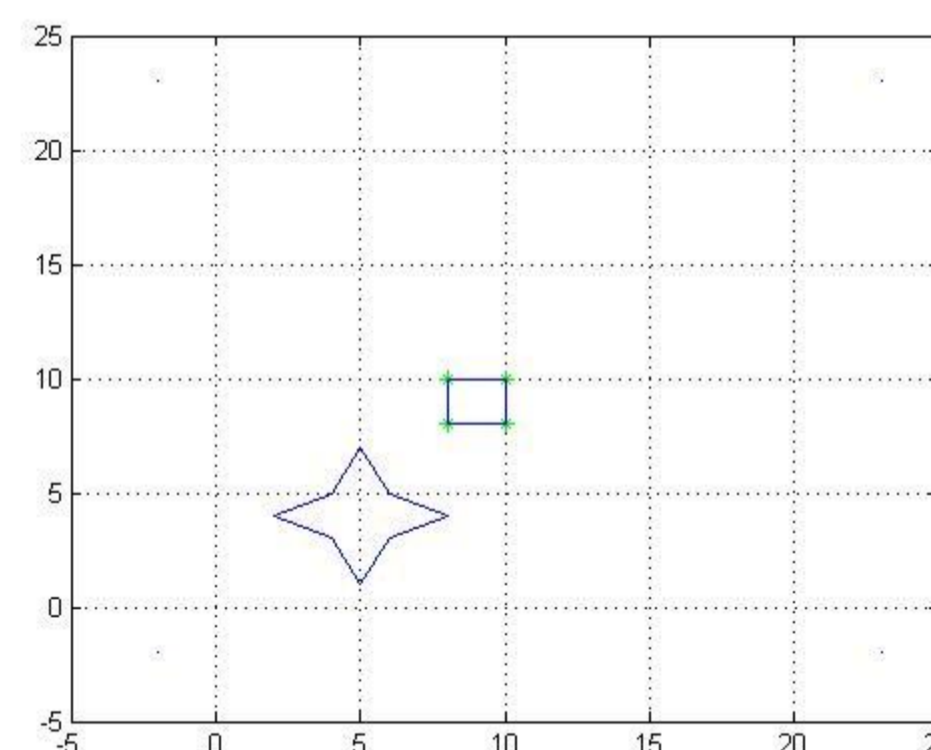
A representação das três Figuras (A, B e $A \oplus B$) obtida pelo programa desenvolvido é, respectivamente:



No caso de figuras não-convexas, a soma de Minkowski é obtida com aproximações, considerando-se no final o fecho convexo a fim de evitar colisões. Veja um exemplo:

$$A = [2 \ 4 \ 5 \ 6 \ 8 \ 6 \ 5 \ 4 \ 2 ; 4 \ 5 \ 7 \ 5 \ 4 \ 3 \ 1 \ 3 \ 4] \text{ Não Convexo}$$

$$B = [8 \ 10 \ 10 \ 8 ; 10 \ 10 \ 8 \ 8] \text{ Convexo}$$



Trabalhos Futuros

O algoritmo ainda precisa de aprimoramentos e expansões. É necessário ampliar a soma para 3 dimensões (o que já está em desenvolvimento), desenvolver um algoritmo capaz de trabalhar com não-convexidade mais precisamente e, não menos importante, diminuir o tempo de processamento para possibilitar a obtenção da configuração livre de um espaço dinâmico, ou seja, com objetos em movimento.

Referências

1. Berg, M.; Cheong, O.; Kreveld, M.; Overmars, M. *Computational Geometry – Algorithms and Applications Third Edition*, Springer, 1998.
2. Matsumoto, E. Y.; *Matlab 6.5 - Fundamentos de Programação*, 2ª Edição, 2003.