

# Modelagem Inicial do Guindaste usando CoppeliaSim

João Victor de Castro Alves  
Juliana Assis Alves  
Pierre Victor da Silva Sousa  
Roberto Pires Oliveira  
Vitor Corrêa Silva

Departamento de Engenharia Elétrica, Universidade Federal de Minas Gerais, Brasil  
Curso de Engenharia de Sistemas

## 1. Introdução

Nesse trabalho iremos modelar a versão básica do guindaste com leves modificações no software CoppeliaSim. Isso tem como objetivo não só comprovar o entendimento do grupo sobre o software e a tarefa mas também consolidar os conhecimentos adquiridos tendo em vista que o projeto modelado deverá ser implementado em sua forma física.

## 2. Definição do Problema

O primeiro passo para desenvolver o modelo é estabelecer quais são as diretivas que iram guiar nosso design. Tendo em vista que nenhuma restrição de dimensão ou design foi diretamente feita pelo professor, iremos usar somente as características do problema que o modelo deve solucionar.

A definição formal do nosso problema pode ser escrita como :

”O guindaste deve ser capaz de pegar uma moeda de 50 centavos a uma altura de 7mm e 150mm de distância do guindaste levar essa moeda até uma base, também a 150mm de distância e com uma altura de até 220mm.”

## 3. Modelagem das peças

Utilizando somente essas definições é possível já extrair algumas das medidas físicas que nosso guindaste deve ter. Permitindo assim a criação de um modelo suficientemente satisfatório. Sendo assim é possível escrever essas definições da seguinte forma :

- A altura da torre deve exceder 220mm.

- O comprimento da lança deve ser, no mínimo, 150mm.
- O atuador deve ser capaz de acoplar e manter acoplado por tempo indeterminado um objeto metálico de 8g.

### 3.1. Base

A base é responsável por garantir a estabilidade de toda a estrutura do guindaste. Temos que garantir que a base tenha tamanho e peso suficiente para garantir que o guindaste não caia, estando ele carregado ou não. Portanto a peça que será usada terá as seguintes características :

- Tipo : Cuboid
- Dimensões : 70mm x 70mm x 20mm
- Massa : 200g

### 3.2. Torre

A Torre precisa atender a restrição de altura tendo uma nível razoável de ”folga”, e ainda sendo factível que esta torre consiga carregar a Lança estando ou não em movimento. Portanto a peça que será usada terá as seguintes características :

- Tipo : Cuboid
- Dimensões : 50mm x 50mm x 300mm
- Massa : 450g

### 3.3. Lança

A lança é responsável por mover atuador até a posição alvo e deve ser capaz de aguentar tanto o peso da carga quanto da estrutura do atuador

- Tipo : Cuboid
- Dimensões :  $250mm \times 30mm \times 15mm$
- Massa : 200g

### 3.4. Contrapeso

O contrapeso tem a função de evitar o desequilíbrio da Lança e garantir que o guindaste não tombe mesmo com o peso da carga.

- Tipo : Cuboid
- Dimensões :  $30mm \times 30mm \times 30mm$
- Massa : 120g

### 3.5. Atuador

O atuador usado será feito de forma customizada no software de forma que sua geometria é de escolha livre. A geometria escolhida será a de um pequeno cilindro, buscando mimetizar a peça utilizada na versão física a ser construída. Portanto :

- Tipo : Cylinder
- Dimensões :  $30mm \times 30mm \times 60mm$
- Massa : 300g

### 3.6. Base do Atuador

Como o atuador será movido tbm horizontalmente ele precisará de uma base própria que terá as seguintes características:

- Tipo : Cuboid
- Dimensões :  $30mm \times 30mm \times 30mm$
- Massa : 30g

## 4. Sensoriamento

Para ser possível ter o set completo de funcionalidades visando controle estável do guindaste será adicionado sensoriamento especificamente dois tipos de sensores :

### 4.1. Sensor de Distância

Será usado um sensor de distância, especificamente do tipo raio, para calcular a distância entre o atuador e tudo que está logo abaixo. As características de configuração podem ser vistas abaixo :

- Tipo : Ray
- OffSet =  $5mm$
- Distância Máxima =  $250mm$

### 4.2. Sensor de Visão

O sensor de visão usado será o de perspectiva , visando garantir certo grau de noção de profundidade. Para isso será usada uma distância mínima de  $3mm$  e uma distância máxima de  $300mm$  e uma resolução de  $1080 \times 720$ . As configurações podem ser vistas a seguir:

- Render Mode : OpenGL
- Near Clipping :  $3mm$
- Far Clipping :  $300mm$
- Persp. angle :  $60^\circ$

## 5. Juntas

Juntas são objetos que tem o a função de simular movimento de uma estrutura em relação a outra. Esse movimento pode ter múltiplos parâmetros definidos porém para a essa iteração da simulação usaremos somente dois. O módulo máximo de velocidade e o torque máximo.

### 5.1. Suporte

A junta de suporte tem como objetivo garantir a união da torre a base gerando assim estabilidade para o guindaste. Sua velocidade estará definida como zero e seu torque terá um valor simbólico de 100 N, buscando dar robustez ao modelo:

- $\left| \vec{V}_{max} \right| = 0$
- $T_{max} = 200$

### 5.2. Plano XY

- $\left| \vec{V}_{max} \right| = 20^\circ / S$
- $T_{max} = 200$

### 5.3. Eixo Z

- $\left| \vec{V}_{max} \right| = 6 \cdot 10^{-2} m/S$
- $T_{max} = 100$

### 5.4. Alinhamento Horizontal

Assim como no vídeo, o guindaste em questão também terá três graus de liberdade portanto é necessário uma junta prismática para gerar esse movimento. Ela terá as mesmas limitações da junta do Eixo Z.

- $\left| \vec{V}_{max} \right| = 6 \cdot 10^{-2} m/S$
- $T_{max} = 100$

## 6. API

Todo o controle do modelo será feito via API usando a linguagem Python. Para isso as seguintes classes e métodos foram desenvolvidos.

### 6.1. Template

### 6.2. Juntas

#### 6.2.1. Junta de Revolução.

#### 6.2.2. Junta Prismática.

### 6.3. Sensor de Distância

### 6.4. Sensor de Visão

### 6.5. Atuador

## 7. Conclusão