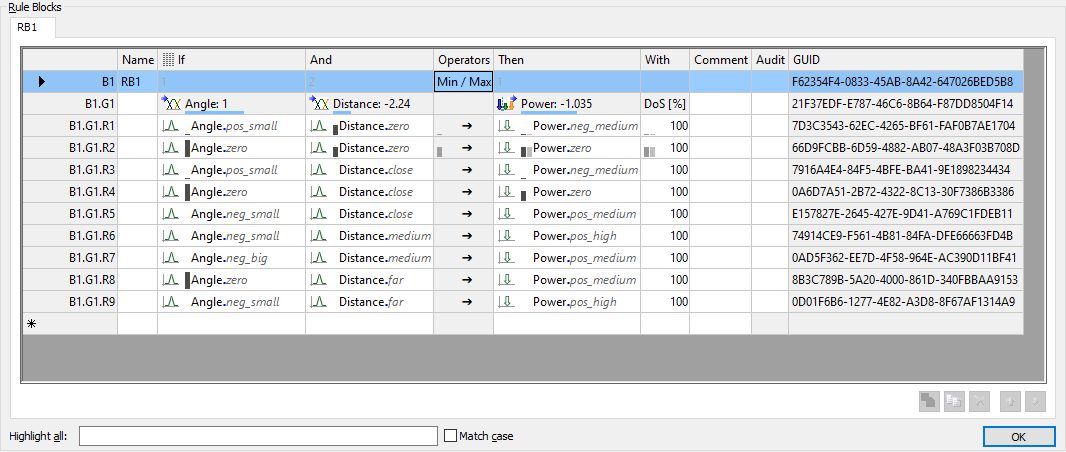
**Relatório de Inteligência Computacional**

Crane example – Guindaste

**Regras do programa**

O software permite que mudemos a base de regras com alguns cliques. Inicialmente, são descritas 9 regras de acordo com a figura abaixo.

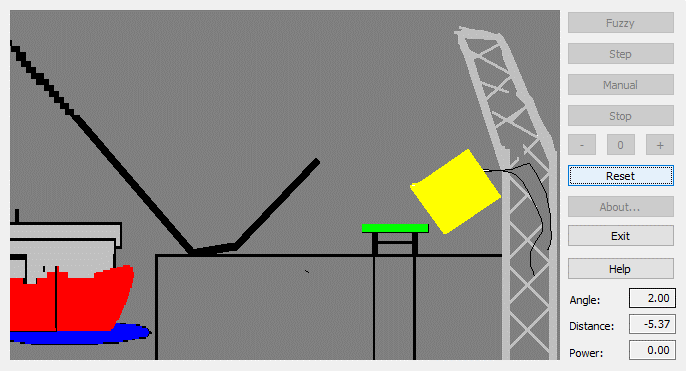


São usadas duas variáveis de entrada (Angle e Distance) e uma de saída (Power). O angle é relacionado ao ângulo que a caixa tem com a cabeça do guindaste e distance é a distancia da caixa para a plataforma. Power é a velocidade de deslocamento do guindaste.

Em relação as regras, é possível observar que qualquer mudança pode ser prejudicial para o bom comportamento do programa. Por exemplo, caso mudemos a regra 2 para:



O resultado é o seguinte:

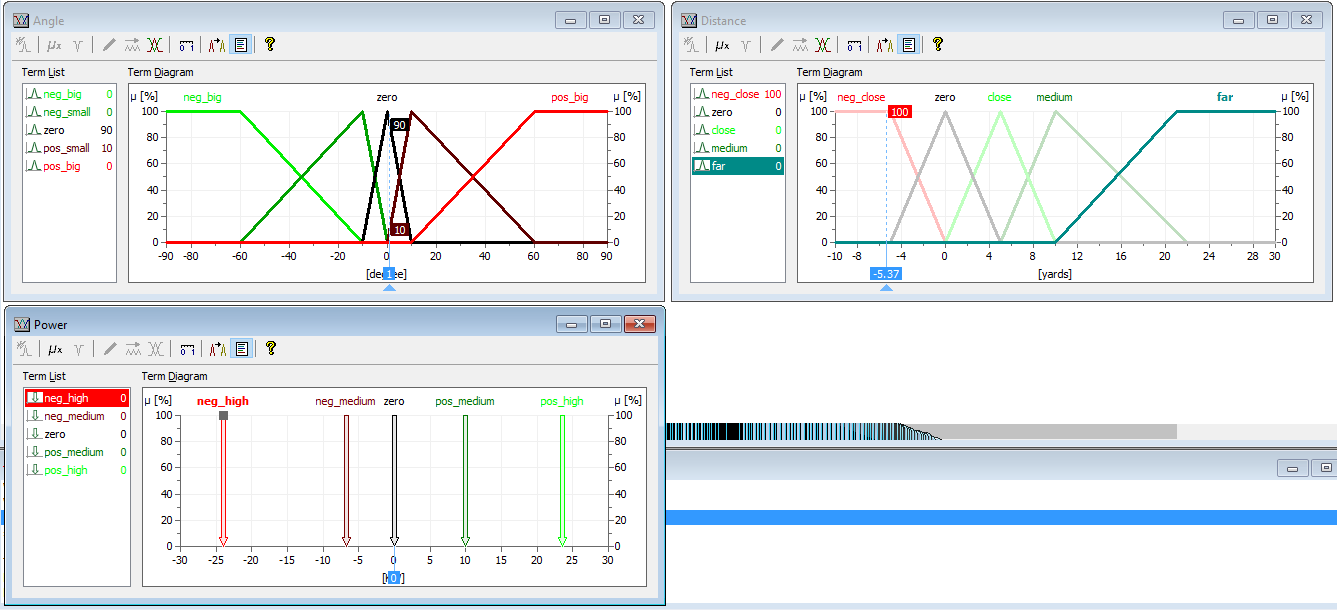


Interpretando, conseguimos ver que a regra diz para o programa aumentar a velocidade caso tenha chegado no destino.

**Grupos de Pertinência**

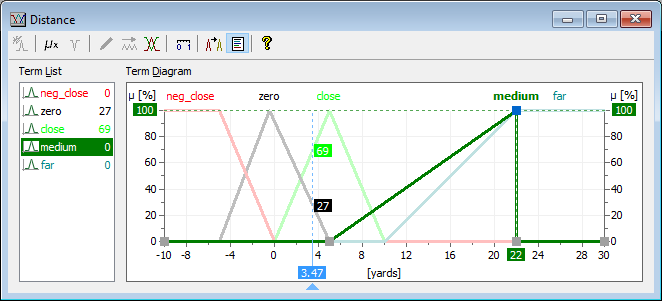
Os grupos de pertinência são essenciais para definir a situação do guindaste. Durante o funcionamento do programa, a pertinência do guindaste em cada grupo mudará até chegar no objetivo que é estacionar na plataforma sem bater.

Os grupos de cada variável são os seguintes:

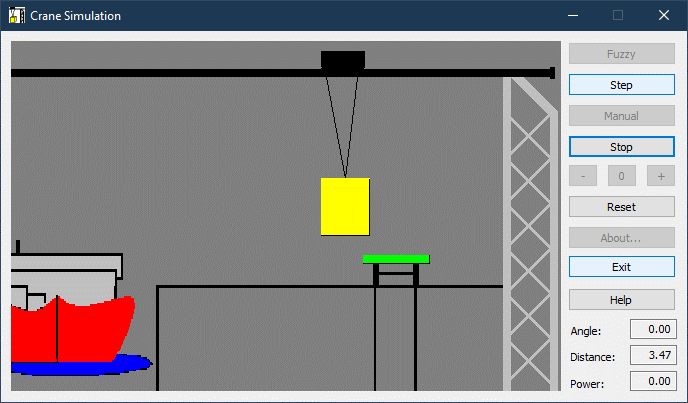


Como visto anteriormente, ao mudar a função de pertinência dos grupos, o resultado final também poderá ser alterado erroneamente.

Ao mudar a definição dos grupos de distancia para:

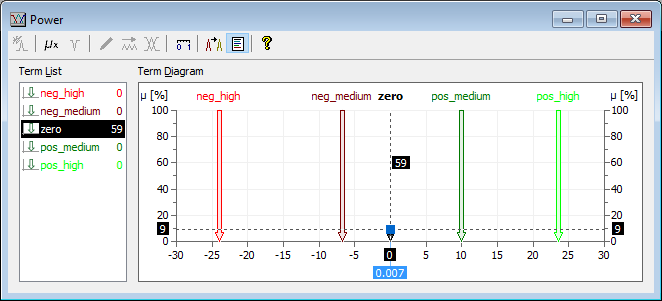


O resultado é o seguinte:

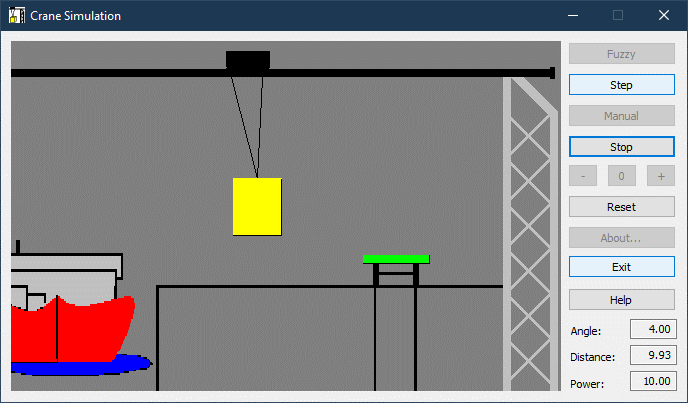


Interpretando novamente, vimos que isso se dá porque o modelo se confunde ao distinguir a distância entre ‘Perto’ e ‘Meio perto’ e nunca chega ao destino.

Outra análise possível se dá modificando a variável de saída para:



O resultado é o seguinte:



O guindaste fica andando de um lado para o outro sem parar na plataforma devido a baixa pertinência ao grupo de velocidade zero.