

# Introdução ao PSO Aplicado a Controle de Processos

M.Sc. André Felipe Oliveira de Azevedo Dantas

# Tópicos Abordados

- Origens;
- Ideia;
- O algoritmo Básico;
- Principais variantes;
- Pesquisas na Área;
- Aplicação a controle;
- Exemplo de aplicação.

# Origens do PSO



Como os pássaros, peixes, abelhas possuem um comportamento coletivo tão coordenado?



# Origens do PSO

Reynolds propôs um modelo de comportamento nos quais os agentes seguem 3 regras:

- Separação:
  - Cada agente tenta se distanciar de seus vizinhos se estiverem muito perto.
- Alinhamento :
  - Cada agente se move em direção a média dos vizinhos;
- Coesão:
  - Cada agente tenta posicionar-se na média dos vizinhos;

# Origens do PSO

Kennedy and Eberhart incluíram um ponto central num modelo semelhante ao de Reynolds tal que:

- Cada agente fosse atraído em direção ao ponto;
- Cada agente “lembrasse” o quão próximo está do ponto;
- Cada agente compartilhasse informações com seus vizinhos sobre a localização mais próxima do ponto.

# A ideia do PSO

E se a noção de distância desse ponto fosse variante de acordo com uma função desconhecida? Os agentes conseguiriam atingir ao mínimo da função?

# O algoritmo Básico

- Criar duas populações de agentes distribuídos uniformemente ao longo de um plano  $X$  (chamados partículas) e de agentes distribuídos uniformemente ao longo de um plano  $P_{bi}$  (Melhores Locais).
- Avaliar a posição de cada partícula de acordo com a função escolhida que se deseja minimizar ou maximizar.

# O algoritmo Básico

- Se a posição avaliada é melhor que a atual “melhor posição”, substitua pela partícula.
- Determinar a melhor partícula.
- Atualize as velocidades de acordo com a equação:

$$\mathbf{v}_i^{t+1} = \mathbf{v}_i^t + \varphi_1 \mathbf{U}_1^t(\mathbf{pb}_i^t - \mathbf{x}_i^t) + \varphi_2 \mathbf{U}_2^t(\mathbf{gb}^t - \mathbf{x}_i^t).$$

- Mova as partículas para novas posições:

- $$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_i^{t+1}.$$



# O algoritmo Básico

- Se o critério de avaliação não foi atingido volte para o ponto 2. Caso tenha sido satisfeito finalize a otimização.

# Principais variantes do PSO

- Quase todas as modificações são realizadas de alguma maneira na lei de adaptação da velocidade das partículas.

$$\mathbf{v}_i^{t+1} = \underbrace{\mathbf{v}_i^t}_{\text{inertia}} + \underbrace{\varphi_1 \mathbf{U}_1^t (\mathbf{pb}_i^t - \mathbf{x}_i^t)}_{\text{personal influence}} + \underbrace{\varphi_2 \mathbf{U}_2^t (\mathbf{gb}^t - \mathbf{x}_i^t)}_{\text{social influence}}$$

# Principais variantes do PSO

- Algumas variantes do PSO são adaptadas para incorporar:
  - Diferentes topologias dinâmicas de vizinhos;
  - Aumento da diversidade entre as partículas;
  - Componentes de outras técnicas;
  - Adaptações para sistemas discretos;

# Pesquisas na Área de PSO

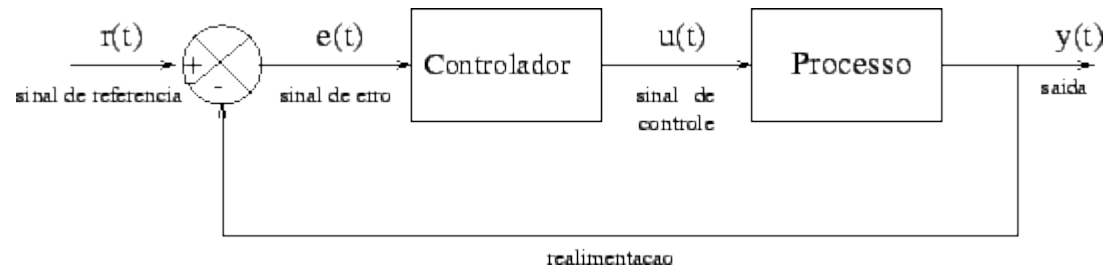
- As principais linhas de pesquisas, na área de PSO, em desenvolvimento são:
  - Utilização do PSO em vários tipos de sistemas (Dinâmicos, estocásticos, combinatórios);
  - Seleção de parâmetros (quantas partículas deveriam ser utilizadas, qual topologia);
  - Comparação entre diferentes tipos de PSOs já existentes;
  - Novas variantes do algoritmo (híbridos, modificações);
  - Aspectos teóricos (Convergência, comportamento das partículas, como melhorar a busca)

# Aplicação a Controle

- A Utilização do PSO em controle de processos visa “melhorar” um determinado processo otimizando funções que refletem o comportamento desejado para o referido sistema. É possível utilizá-lo para:
  - Identificar um sistema dinâmico (otimizando modelos não lineares, sujeitos a ruídos de várias naturezas);
  - Otimizar parâmetros de controladores (encontrando o melhor parâmetro para o desempenho desejado);
  - Otimizar as funções dos controladores (sintetizando o sinal de controle utilizando o algoritmo).

# Exemplo de aplicação

- Sintonia de Controladores PID:
  - Dado o seguinte sistema:



- O controlador PID sintetiza a seguinte lei de controle:

$$u(t) = K(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt})$$

# Exemplo de aplicação

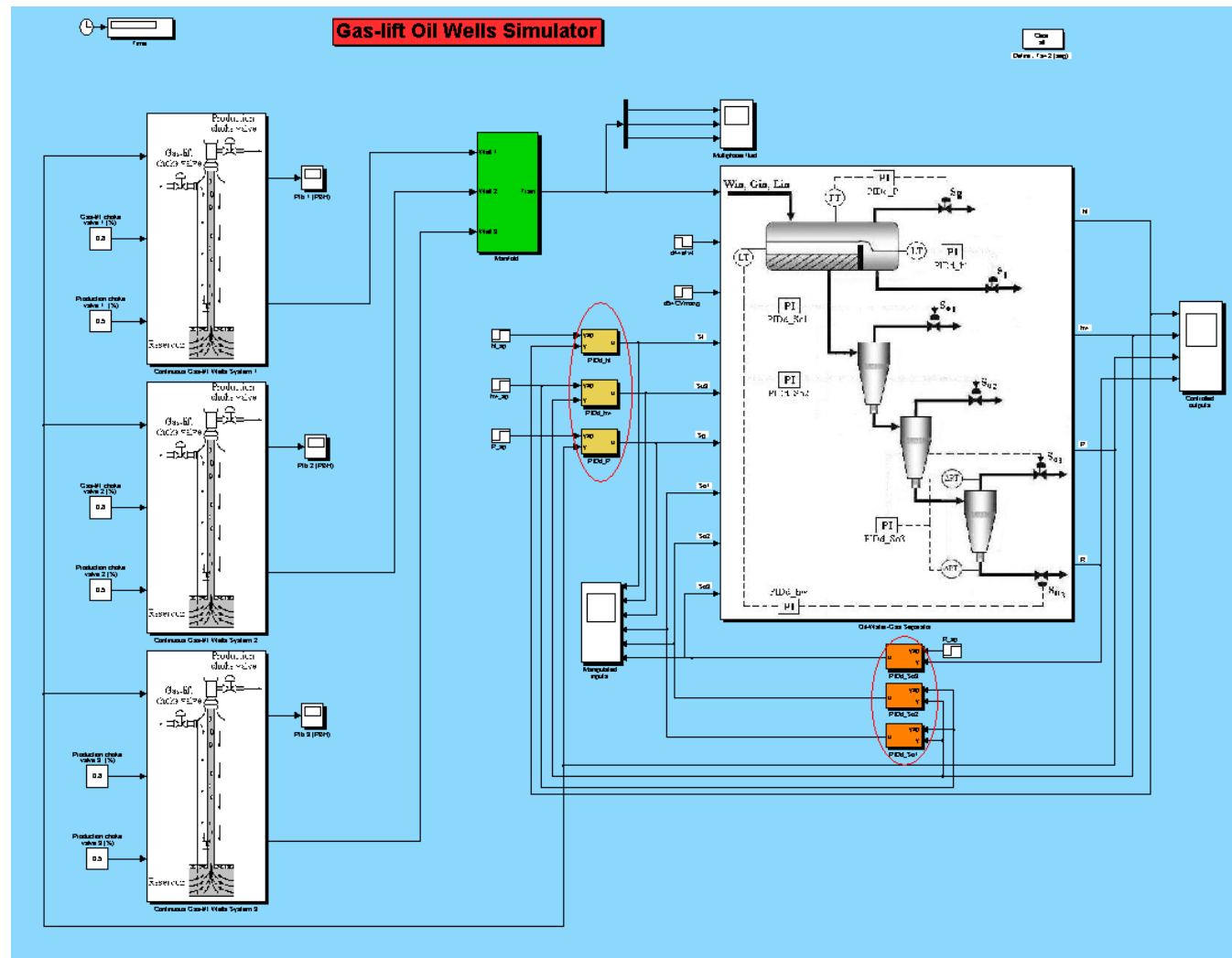
- Ou seja, em cada controlador existem 3 parâmetros a ser encontrados. Para isso existem métodos consagrados na literatura que são facilmente implementados. Ex:
  - Ziegler e Nichols;
  - Método do Relé (adaptativo).

# Exemplo de aplicação

- Porém, para uma malha de controle com mais de um controlador como encontrar os melhores parâmetros para que o controle do sistema inteiro seja o melhor possível?

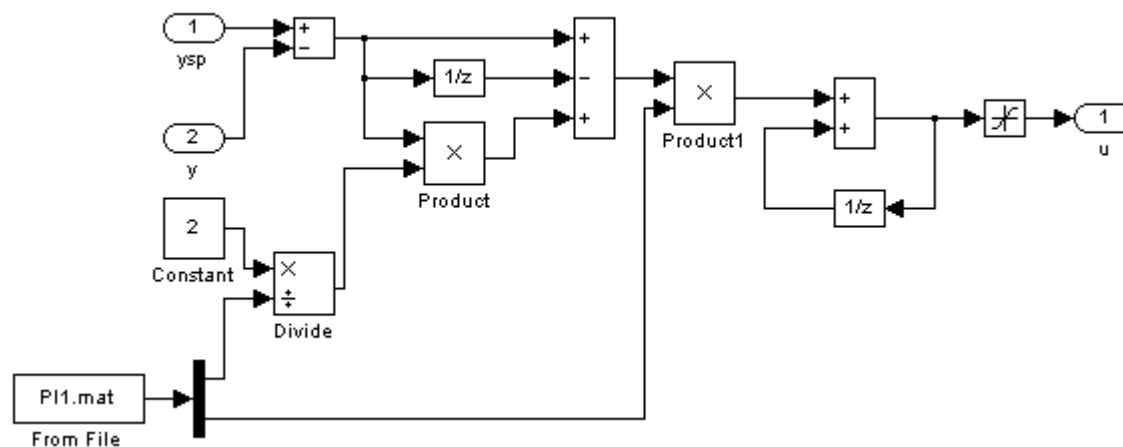


# Exemplo de aplicação



# Exemplo de aplicação

Controlador PI, ou seja,  $T_d = 0$ .



# Exemplo de aplicação

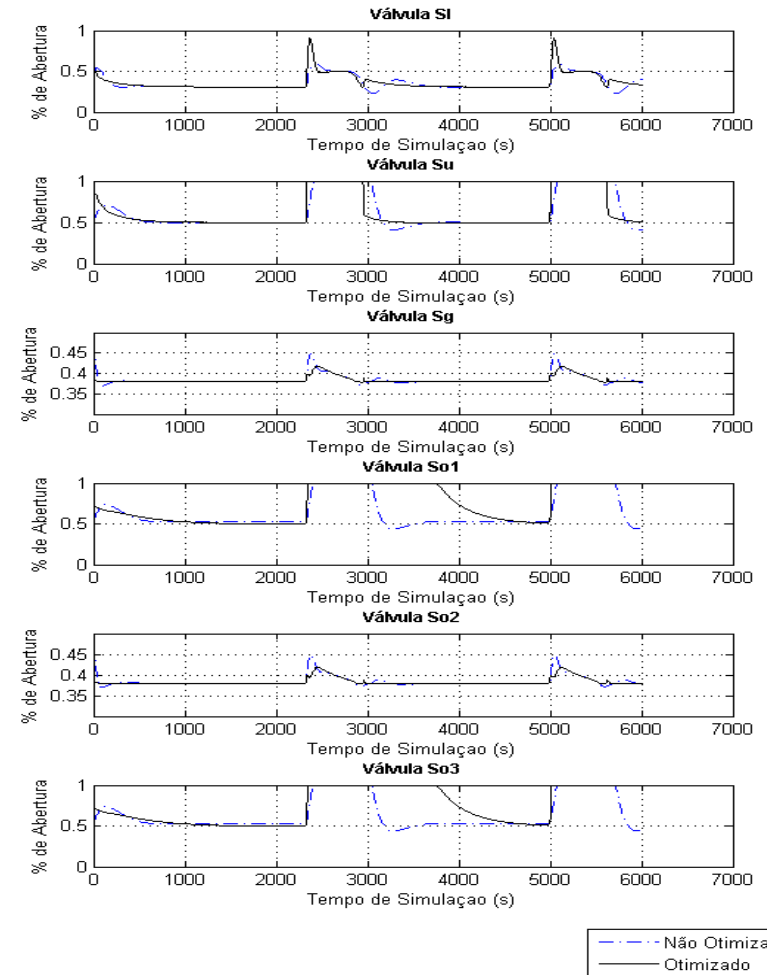
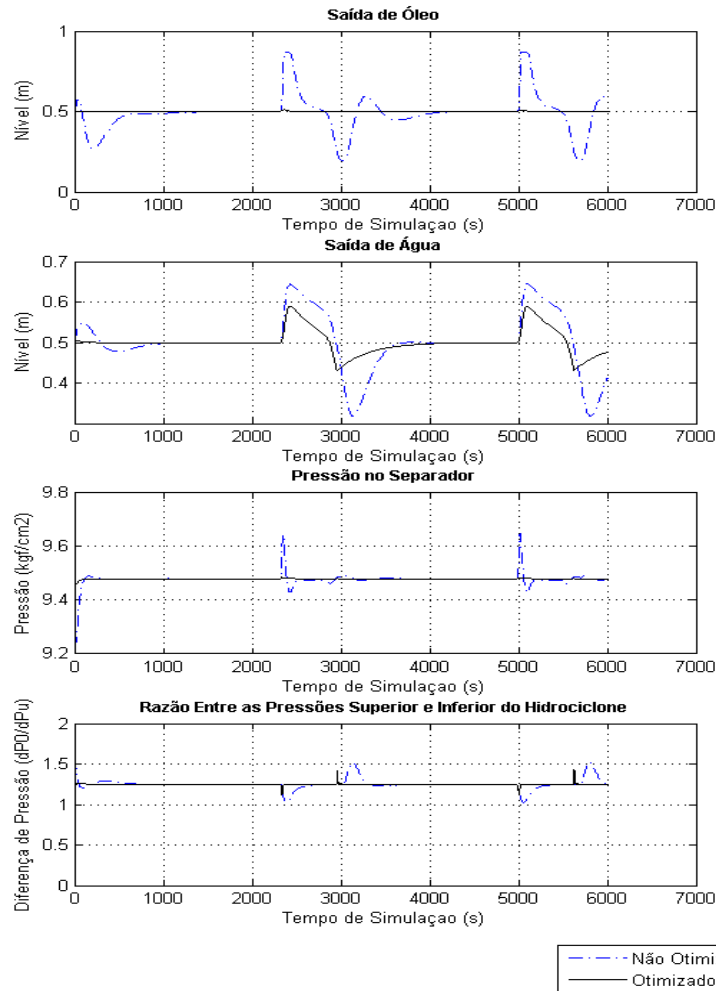
Algoritmo Base para coleta de dados do simulador.

```
function [OI] = SintoniaPred(X)

    SIMIN1 =[0; X(1:2)'];
    for i=1:2;
        SIMIN1 =[ SIMIN1 [0;X(1:2)']];
    end
    save('PI1','SIMIN1');
    sim('gasliftwells33_03pocos_03hidros.mdl')
    RefO = 0.5;
    OI = Ss.signals(1,1).values - RefO;

end
```

# Exemplo de aplicação



# Conclusões

- Sistema otimizado com sucesso;
- Esforço de controle dentro das limitações e não muito alterado;
- Aplicações em linguagens mais rápidas pode ser um diferencial devido ao custo computacional;

# Perguntas?