



Sistemas de Tempo Real

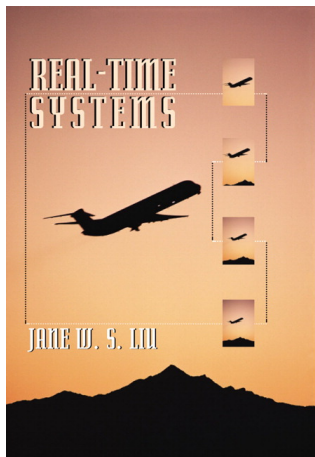
Sérgio Campos — `scampos@dcc.ufmg.br`

Sistemas de Tempo Real

- Motivação — aplicações típicas:
 - Controle e sistemas de controle,
 - Processamento de sinais,
 - Sistemas embutidos,
 - Robótica,
 - Multimedia.
- Antecedentes:
 - Sistemas reativos,
 - Processos: comunicação, sincronização,
 - Escalonamento,
 - Sistemas distribuídos.

Sistemas de Tempo Real

- Modelagem:
 - Processadores,
 - Recursos,
 - Comunicação.
- Múltiplas dimensões:
 - Hard X soft real time,
 - Periódicos X aperiódicos,
 - Preemptividade.



- Jane Liu, *Real-Time Systems*, Prentice Hall, 2000

Motivação

- O que são ?
- O que não são ?
- Exemplos
- Controle e sistemas de controle,
- Processamento de sinais,
- Sistemas embutidos,
- Robótica,
- Multimedia.

O que são ?

Sistemas aonde um resultado atrasado não tem valor, ou tem valor negativo.

Tempo adiciona um componente ortogonal ao projeto:

- Tudo que era feito antes continua sendo feito, mas agora levando em conta o tempo.
- Não basta estar correto,
- Mas tem que estar correto!

Sistemas de Tempo Real

Extremamente importantes; Estão presentes em:

- Telefonemas: controle de centrais telefônicas;
- Carros: abs, i.e., suspensão, sinais;
- Aviões: controle de tráfego, voo, direção;
- Momentos ruins: monitoração de batidas cardíacas, pressão de sangue, cirurgias.
- Momentos alegres: jogos, vídeo.

Normalmente escondidos: quando funcionam bem não os notamos.

O que não são ?

Controle de estoque, folha de pagamento, reserva de passagens.

- Nestes casos um dado atrasado ainda serve.

Device drivers

- Software básico é usado em RT, e tem que ser RT, mas por si só não fazem um sistema RT.

Pedaços de sistemas

- Não adianta fazer somente parte do sistema RT. Por exemplo, X Windows nunca pode fazer parte. Porque ?

Sistemas rápidos

- Rapidez é sempre bom, mas não garante nada. Por exemplo, um deadlock a 4GHz é tão ruim quanto um a 30MHz!

Importância e Dificuldade

Setembro de 1994:

- Durante a aproximação ao aeroporto de Pittsburgh, EUA, um 737 voando a 6000 pés com dia claro e sem turbulências caiu.
- Sem aviso o avião começou a virar para a esquerda. Os pilotos não conseguiram recuperar o controle.
- O avião caiu a 301 milhas por hora, matando 132 pessoas instantaneamente!

O acidente:

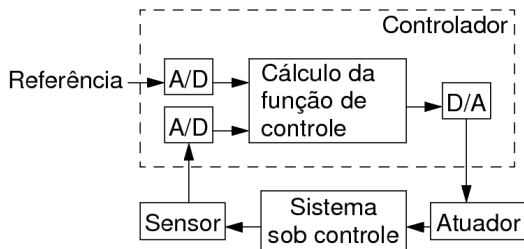
- Aumento de velocidade — turbulência ?
- 3 seg: nariz 3 graus à esquerda
- Em seguida: abruptamente 6 graus à esquerda — provavelmente estabilizador vertical.
- Acelerou asa direita, freou a esquerda, avião começou a virar para esquerda.
- 8 seg: “roll” de 60 graus, avião em queda.

Importância e Dificuldade

Uma explicação (não existe uma oficial):

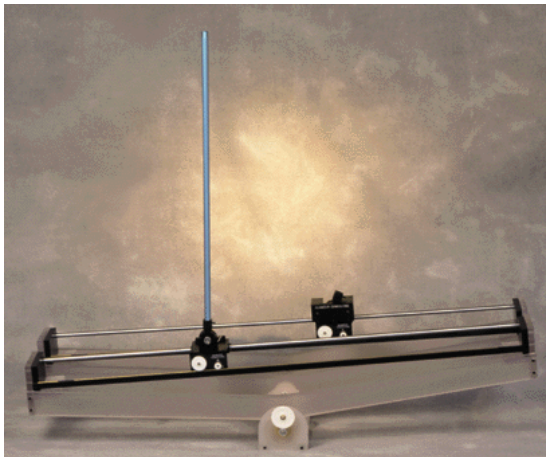
- O estabilizador vertical virou de uma vez causando o desvio do nariz do avião, de forma independente (possivelmente um motor se desligou).
- O avião “escorregou” para a esquerda.
- O piloto automático achou que o piloto queria virar para a esquerda e tentou ajudar.
- Obviamente a ajuda não ajudou e em 10 segundos o avião estava de cabeça para baixo e em queda livre.
- Os pilotos desligaram os motores pois acharam que o piloto automático estava com problemas
 - Técnica usada para dar tempo de recuperar o controle do avião
 - Não havia tempo para se recuperar, pois o avião estava muito próximo do chão.

Exemplo: Controle

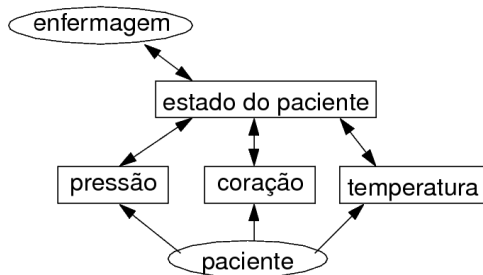


- Usado antes de computadores existirem.
- Versão digital usa amostragem:
 - Dados são lidos dos sensores e atuados no sistema periodicamente;
 - Precisão da amostragem depende do período de leitura;
 - Tempo de resposta depende do período do atuador.
- Função de controle determina corretude:
 - problemas de estabilidade.

Pêndulo Invertido



Hierarquia de Controladores



- Frequentemente são necessários diversos controladores:
- Interação entre controladores torna projeto extremamente complexo
 - Diversas oportunidades para erros: temporais ou funcionais
 - Controle conflitantes: e.g. cada controlador associado a um robô, eles podem se chocar.
 - Ou pior, ao invés de robôs, aviões.

Controle

Controle é um problema muito difícil:

- Funções de controle nem sempre são fáceis de achar;
- Problemas de periodicidade:
 - Períodos menores : maior precisão e custo;
 - Períodos maiores : pior estabilidade.
- Múltiplos controladores:
 - Frequentemente resolver conflitos implica em algoritmos de otimização;
 - Mas algoritmos de otimização não podem ser usados. Porque ?
 - Exemplo: Decidir a rota de cada avião para minimizar consumo de combustível é NP-completo!

Sistemas Embutidos

From Computer Desktop Encyclopedia
© 2008 The Computer Language Co., Inc.



High-Speed Control

- ◆ Antilock Braking
- ◆ Central Electronics
- ◆ Electronic Throttle
- ◆ Engine Control
- ◆ Steering Wheel
- ◆ Transmission Control

Low-Speed Control

- ◆ Audio
- ◆ Climate Control
- ◆ Driver's Door
- ◆ Driver Information
- ◆ Passenger Door
- ◆ Phone
- ◆ Power Seat
- ◆ Rear Electronics
- ◆ Sun Roof
- ◆ Supplemental Restraint System
- ◆ Upper Electronics

Sistemas Embutidos



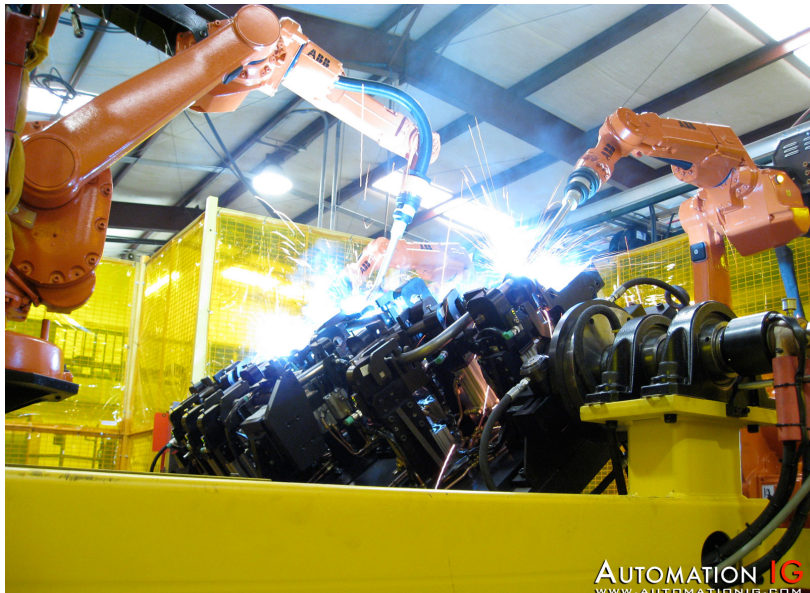


Photo Courtesy of Inuktun

Usado em usinas nucleares para determinar se a tubulação se deformou.

- Ele anda por dentro das áreas radioativas;
- Sensores dizem se os tubos estão deformados;
- Sensores dizem sua posição — evita bater em paredes;
- Recebe comandos via rádio;
- Comandos podem ser para ligar e desligar sensores e para se mover;
- Cada item acima é implementado como um processo independente e periódico.

Robótica



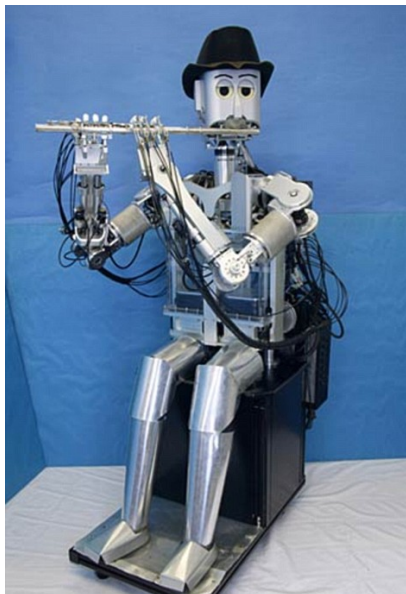
Robótica



Robótica



Robótica



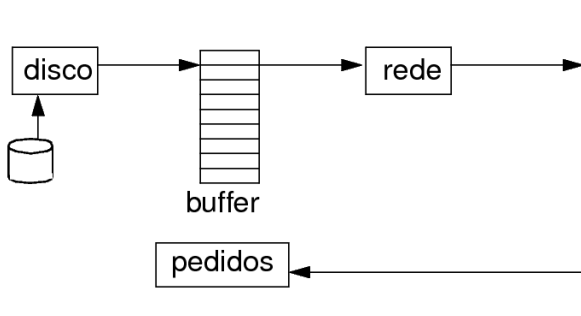
Multimedia

Transmissão de áudio e vídeo:

- Tempo real ?
 - Se áudio ou vídeo atrasarem a transmissão não funciona;
 - Mas ninguém morre.
- Critérios de otimização diferentes:
 - Qualidade de serviço — QoS;
 - Parâmetro que determina com que qualidade vai-se atender clientes
 - QoS 0: custo baixo, qualidade ruim;
 - QoS 1: custo alto, boa qualidade.
 - QoS depende da aplicação:
 - Áudio: QoS 1
 - Vídeo (entretenimento): QoS 0.8
 - Vídeo-conferência: QoS 0.5

Vídeo sob Demanda

Servidor:



Multimedia: QoS > 1

