



# 2º CONGRESSO BRASILEIRO DE P&D EM PETRÓLEO & GÁS

## DESENVOLVIMENTO E ANÁLISE DE INTERFACES DE COMUNICAÇÃO ENTRE UM SIMULADOR COMERCIAL E SOFTWARES DE PROGRAMAÇÃO

Jessé Felipe Müller<sup>2</sup>, Letícia Gomes Moura<sup>1</sup>, Raúl Alves Santos<sup>2</sup> e Julio E. Normey-Rico<sup>2</sup>

<sup>1</sup>Departamento de Engenharia Química e Engenharia de Alimentos, Universidade Federal de Santa Catarina 88040-900  
CTC-UFSC Florianópolis-SC Brasil, leticia@enq.ufsc.br

<sup>2</sup> Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina  
88040-900 CTC-UFSC Florianópolis-SC Brasil, {jesse, ralves, julio}@das.ufsc.br

**Resumo** – Este trabalho apresenta uma análise de um simulador de processos e as suas interfaces de comunicação como ferramentas para o estudo de processos e projeto de controladores para a indústria do petróleo e gás natural. Utilizando o simulador INDISS, desenvolveram-se duas aplicações. Na primeira implementou-se um processo de controle de temperatura e nível em reservatórios com o objetivo de analisar o desempenho do simulador e a possibilidade de aplicação de algoritmos de controle avançado. O segundo processo implementado, uma coluna de destilação depropanizadora, foi utilizado para avaliar a capacidade do simulador para trabalhar com modelos de grande porte e testar a sua fidelidade. Para permitir a programação de algoritmos de controle avançado de forma simples e eficiente, foi desenvolvida uma interface que permite o acesso externo ao simulador a partir do software MATLAB. Os resultados de utilização desta interface são ilustrados com o uso de um controlador aplicado ao processo de controle de temperatura de topo na coluna de destilação.

Palavras-Chave: simulador; controle de processos; MATLAB; comunicação

**Abstract** – This work presents the analysis of a process simulator and its communication interfaces as tools used for the analysis and design of oil and gas industry process controllers. Two applications were implemented using the INDISS simulator. In the first one, a temperature and level control process in reservoirs is used to analyze the performance of the simulator and the possibility of application of advanced control algorithms. The second implementation consists of a depropaniser distillation column, which has been used to evaluate the capacity and fidelity of the simulator when working with large scale models. To allow the programming of advanced control algorithms in a simple and efficient way, an interface was developed allowing the external access to the simulator from the software MATLAB. Some simulation results with a control structure illustrate the use of the proposed interface.

Keywords: simulator, process control, MATLAB, communication

## 1. Introdução

A técnica de simulação já é bastante difundida na indústria do petróleo e gás, visto que proporciona um ambiente favorável ao estudo aprofundado de processos, sem que seja necessário o contato direto com o processo real. Desta forma, operadores inexperientes podem ser treinados em simuladores até que atinjam um grau de capacitação e conhecimento exigidos para a operação de processo. Isto é, eles devem ser capazes de conduzir o processo e detectar eventuais anomalias, tomando as devidas providências para garantir seu bom funcionamento (de Prada et al., 2002). Assim, evitam-se eventuais problemas graves tanto de segurança, como de qualidade do produto a que se destina o processo, que possam ocorrer devido à inexperiência no manuseio da planta, como também da falta de conhecimento do processo. Do mesmo modo, através do uso de simuladores, torna-se viável o estudo e o aprimoramento de estratégias avançadas de controle, que se tornaram uma necessidade urgente nas indústrias, dado que os processos produtivos tiveram que ser adaptados às novas características do mercado internacional (Camacho e Bordons, 1999).

O desenvolvimento de simuladores é realizado tradicionalmente com linguagens de propósito geral, como C ou FORTRAN, ou com linguagens de simulação, como SIMULA, ACSL, SIMSCRIPT, etc., que facilitam o desenvolvimento de simulações de processos.

Nos últimos anos têm surgido outros tipos de linguagens de simulação mais avançadas, com orientação a objetos e componentes reutilizáveis. Um exemplo disto é a linguagem EL utilizada pelo software EcosimPro (EcosimPro, 2003). Está é uma linguagem de simulação orientada a objetos que permite desenvolver bibliotecas de componentes para a reutilização posterior no desenvolvimento de simuladores. O objetivo principal destas novas linguagens é diminuir o tempo de desenvolvimento e conseqüentemente os custos, ao usar componentes que já foram provados, e ser capazes de reutilizar facilmente componentes anteriormente desenvolvidos.

Com esta idéia surgiu o projeto CAPE-OPEN (Computer Aided Process Engineering – Open Simulation Environment), procurando uma modelagem rápida, precisa e efetiva para a síntese, projeto, monitoração e otimização de processos químicos e relacionados (CAPE-OPEN, 2003). Este projeto propõe um padrão para as interfaces dos componentes utilizados em simulações, que deve ser respeitado por aqueles que os desenvolvem, a fim de permitir a integração e troca destes entre diferentes fabricantes (Braunshweig et al., 2002). Um destes fabricantes é a empresa francesa RSI (RSI, 2003), que desenvolveu o software de simulação INDISS (Industrial and Integrated Simulation Software), capaz de representar o sistema real como um todo no desenvolvimento de processos simulados.

Existem basicamente dois tipos de simuladores industriais utilizados atualmente. O primeiro tipo dispõe de uma vasta biblioteca de componentes padronizados. Na concepção de um novo processo, basta utilizar os componentes disponibilizados pelo simulador e somente fazer suas interligações e o ajuste dos parâmetros, de acordo com o processo a ser simulado. Estes normalmente não permitem a criação de novos componentes e são muito utilizados para o treinamento de novos operadores de processo.

Já o segundo tipo de simuladores, utilizados para aplicações mais restritas, permitem não só a edição de componentes existentes, como também a criação de novos, tornando muito mais flexível sua utilização em diferentes tipos de processos. Devido a esta flexibilidade, estes são muito utilizados na pesquisa, pois permitem que novos processos ou novas técnicas de controle sejam simulados e estudados, possibilitando um estudo de viabilidade destes.

Após a criação de novos processos em simuladores, há a necessidade de se fazer a validação do modelo. Esta pode ser realizada através da comparação dos resultados de simulação com o processo real ou com outro modelo de simulação já implementado e validado em outro simulador. A opinião de operadores experientes também desempenha vital importância na validação.

Este trabalho apresenta uma análise do simulador INDISS e de uma interface com o MATLAB, implementada para ampliar as possibilidades de uso do simulador comercial. O trabalho é resultado de um projeto de pesquisa que tem por objetivo analisar softwares de simulação como ferramentas para o estudo de processos da indústria do petróleo e gás, e posteriormente para projeto e ajuste de controladores nestes processos.

O artigo está organizado da seguinte forma: Na próxima seção é apresentado o simulador INDISS. Na seção 3, analisa-se o sistema de comunicação e interfaces desenvolvidas para a conexão com outros softwares. A seção 4 apresenta resultados de aplicação do simulador e o sistema de comunicação para um processo com controle de nível. O artigo é finalizado com os resultados obtidos e as conclusões.

## 2. O Simulador INDISS

O INDISS pertence ao grupo de simuladores baseados em bibliotecas de componentes. Funciona tanto no modo estático quanto no dinâmico. Tem como característica um código fechado, impossibilitando a edição e criação de novos componentes. Pode ser utilizado como uma ferramenta para: projeto e desenvolvimento de processos, otimização de sistemas de controle e procedimentos de operação, treinamento de operadores ou ainda para ensino em cursos técnicos e de engenharia.

São estudados neste trabalho dois casos de utilização do simulador. O primeiro trata-se de um processo com controle de nível e temperatura em um reservatório através de controladores PID (Proporcional, Integral, Derivativo). O segundo caso estudado é o controle de uma coluna de destilação do tipo depropanizadora.

### 2.1. Controle PI para Controle de Nível

O processo considerado consiste de um controle de temperatura e nível de fluidos em um tanque atmosférico. O controle de temperatura é feito através da variação de potência de um aquecedor elétrico na alimentação do reservatório. O controle de nível por sua vez é realizado atuando na abertura da válvula de controle associada à tubulação na saída do reservatório.

A concepção de um modelo de processo no INDISS se faz utilizando as bibliotecas de componentes disponíveis e ajustando os parâmetros de acordo com o processo a ser modelado. Como exemplo, é apresentado o desenho esquemático deste processo (Process Diagram) conforme Figura 1 a). Os componentes necessários para a montagem do modelo foram escolhidos nas bibliotecas e dispostos no ambiente de desenvolvimento do simulador.

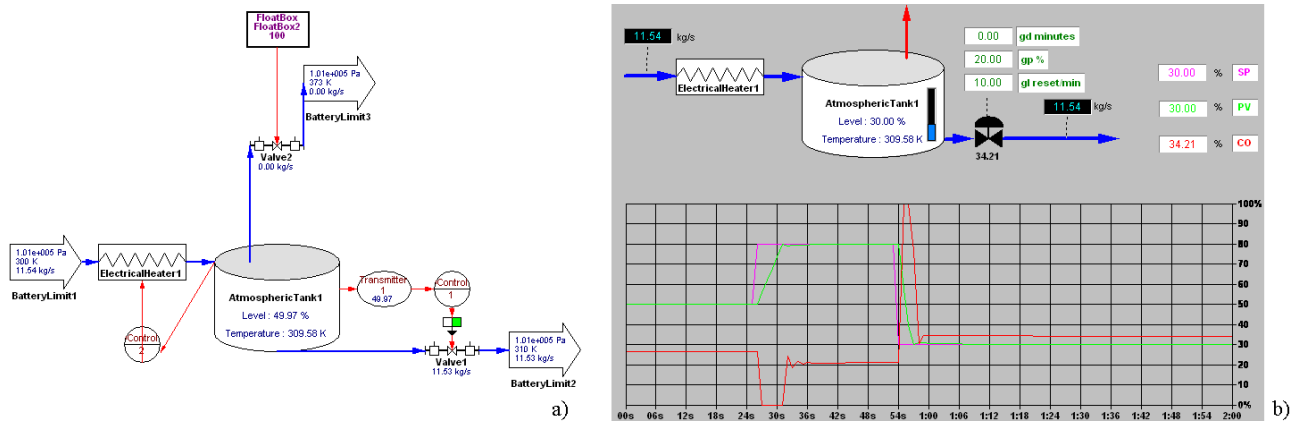


Figura 1. a) Diagrama de Processo do tanque; b) Tela de operação do processo

Este software também permite a criação de uma tela de operação para o processo modelado. Como exemplo, é apresentada na Figura 1 b) a tela de operação (Mimic Display) do sistema de controle de nível e temperatura do reservatório, que possibilita tanto a visualização da evolução das variáveis de interesse no tempo como também a definição dos parâmetros do controlador PI associado a válvula de atuação.

### 2.2. Coluna de Destilação

A disponibilidade de um simulador viabiliza a avaliação de um sistema complexo, como o processo de destilação industrial, que envolve a coluna de destilação como unidade principal, e muitos outros equipamentos periféricos. É possível fazer modificações do sistema com facilidade, sem riscos e a baixo custo. Em uma planta real, alguns estudos seriam inviáveis, outros impossíveis. O diagrama da Figura 2 traz a representação da coluna:

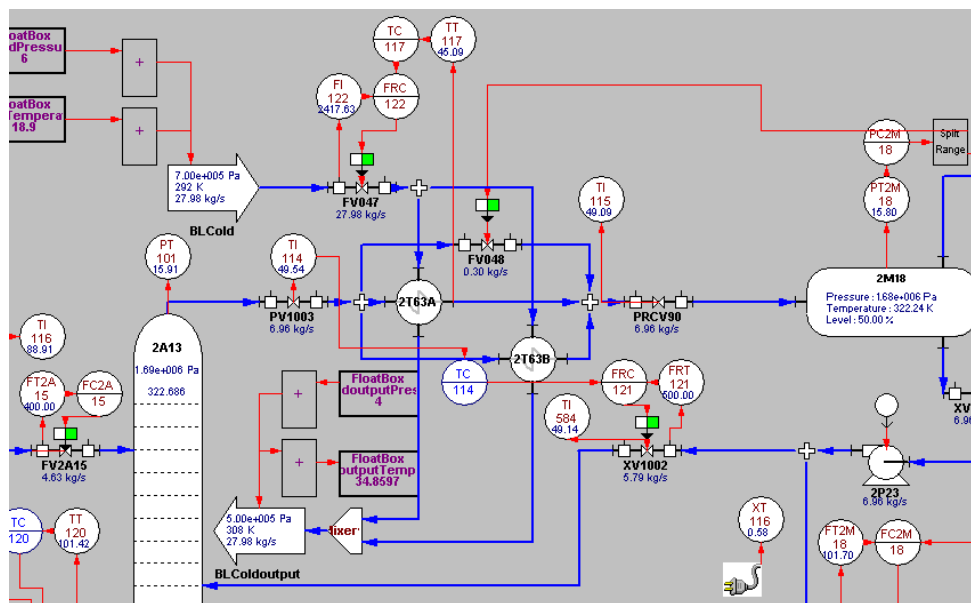


Figura 2. Diagrama de Processo da coluna de destilação

Utilizando-se deste simulador, fez-se um estudo de um modelo de simulação de uma coluna de destilação do tipo depropanizadora, com 12 metros de altura e 30 pratos, cuja alimentação variou entre misturas de diversas composições de propano, n-butano, isobutano, pentano e hexano. Foi avaliado o comportamento do processo para modificações de ponto de operação e ainda perturbações nas variáveis de entrada. O objetivo final neste caso é a validação de uma Rede Neural como modelo não linear na identificação da dinâmica do processo representado pela coluna de destilação. Estão sendo feitos vários testes para analisar o comportamento da coluna, e a partir destas informações, deseja-se propor modificações do seu sistema de controle com o intuito de melhorar seu comportamento dinâmico.

### 3. Comunicação

Apesar das vantagens do simulador INDISS e sua fidelidade na representação de modelos reais, seu inconveniente principal é ter um código fechado, não tornando possível a criação de novos componentes diferentes daqueles que vem junto com o pacote do simulador. Isto impossibilita, por exemplo, o uso de controladores avançados que não vem junto com a biblioteca do INDISS.

Dentro deste contexto, foi concebida uma interface entre o simulador INDISS e o software MATLAB (The MathWorks, 2003), que tornou possível tanto a aquisição de dados do INDISS pelo MATLAB quanto à definição de variáveis do INDISS através do MATLAB.

Pelo fato de o simulador INDISS ter sido desenvolvido em Microsoft Windows, ele tem uma série de facilidades para sua comunicação com outras aplicações por meio da sua interface OLE/COM (Object Linking and Embedding / Component Object Model) (Microsoft MSDN, 2003) e com componentes desenvolvidos de acordo com o padrão CAPE-OPEN. Também permite o uso de componentes incluídos em DLLs (Dinamic-Link Library) desenvolvidos externamente, incluindo um SDK (Software Development Kit) para o desenvolvimento deles.

O acesso ao simulador através da sua interface OLE é direto a partir de aplicações que podem acessar a um servidor de automatização OLE, como Excel ou Visual Basic, como pode ser observado na Figura 3.

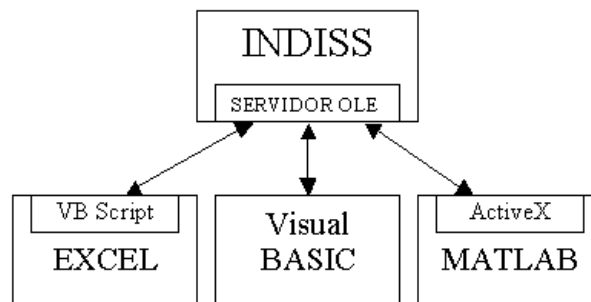


Figura 3. Comunicações entre INDISS e outras aplicações

O procedimento para o acesso consiste na criação de um objeto do tipo da aplicação INDISS que uma vez criado, pode-se acessar às funções definidas na interface OLE:

```
Public INDISSOLE As Object
Public Const ApplicationName As String = "Builder.Application"
Set INDISSOLE = CreateObject(ApplicationName)
INDISSOLE.RunSimulation
```

As funções mais importantes para o acesso ao simulador INDISS através do seu servidor OLE são apresentadas na Tabela 1.

Tabela 1. Funções de acesso ao servidor OLE do INDISS

Função	Ação
RunSimulator	Inicia a simulação que esteve carregada no INDISS
StopSimulator	Para a simulação
EvaluateExpression	Lê um dado da simulação
SetVariableValue	Escreve um dado na simulação
GetSimulationTime	Captura o tempo da simulação

Existem outras séries de funções que permitem, por exemplo, alterar a velocidade da simulação, obter a lista de variáveis disponíveis e carregar uma simulação.

Um dos objetivos do trabalho é o acesso ao simulador, através do seu servidor de automatização OLE, desde MATLAB (MathWorks, 2003). Em princípio não é possível levar a cabo de um modo direto a comunicação com INDISS. Para isto, desenvolveu-se um controle ActiveX em Visual Basic, o qual encapsula todo o acesso às funções do INDISS, de forma que atua como ponte de comunicação entre ambos.

Para acessar a partir do MATLAB as funções do controle ActiveX criado, existe uma série de funções de acesso geral a controles ActiveX. A tabela 2 mostra um exemplo de como são as chamadas às funções a partir de um programa MATLAB.

Tabela 2. Chamadas às funções desde MATLAB

Ação	Sentença em MATLAB
Criação do ActiveX	INDISS = actxserver('CONTROL.INDISS');
Início da simulação	invoke(INDISS,'RunSimulation')
Captura do tempo	invoke(INDISS,'GetSimulationTime');
Leitura de um dado da simulação	invoke(INDISS,'EvaluateExpression',strPV);
Escritura de um dado na simulação	invoke(INDISS,'SetVariableValue',strMV, U);
Espera	invoke(INDISS,'delay',T*1000);
Parada da simulação	invoke(INDISS,'StopSimulation')

A primeira ação a ser feita é a criação do controle ActiveX. Quando este já estiver criado, pode-se chamar as funções nele contidas. A partir das chamadas às funções, pode-se gerar programas em MATLAB para capturas de dados, controle, etc.

#### 4. Aplicação da Interface para o Controle de Nível

Utilizando a interface desenvolvida entre o simulador INDISS e o software de programação MATLAB, foi programado em MATLAB um modelo de controlador PID com dois graus de liberdade (PID 2DOF). Este controlador, diferentemente dos controladores PIDs tradicionais, é adequado quando se deseja controlar sistemas com especificação de resposta tanto para o seguimento de referência como para a rejeição de perturbações. A Figura 4 apresenta o diagrama de blocos deste controlador:

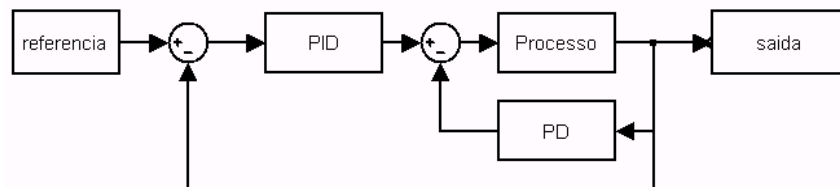


Figura 4. Diagrama de blocos do controlador (PID 2DOF)

Este controlador foi utilizado para o controle de nível do tanque citado anteriormente, de forma tal que as variáveis utilizadas pelo controlador (neste caso o nível atual e o nível desejado), são lidas do INDISS para o MATLAB, é realizado o cálculo da lei de controle no MATLAB e finalmente a ação de controle é aplicada no processo simulado no INDISS (neste caso, no valor de abertura da válvula).

#### 5. Resultados e Conclusões

A interface de comunicação concebida trouxe uma série de vantagens na utilização do INDISS como ferramenta para o estudo de processos, tais como: aquisição de dados do simulador a fim de fazer identificação de modelos através de ensaios em malha aberta; manter arquivos de histórico de dados de simulações; projeto de controladores avançados no MATLAB controlando processos no INDISS.

O software de programação MATLAB apresenta uma série de vantagens para projeto de controladores visto que contém várias funções e ferramentas especialmente concebidas para este fim. Sua linguagem de programação é bastante simples e intuitiva. Muitos algoritmos de controle avançado tem sido programados utilizando-se este software. Estes podem ser facilmente testados e avaliados em processos simulados no INDISS por intermédio da interface desenvolvida.

Como primeiro exemplo, é apresentada a seguir a resposta temporal do processo de um trocador de calor na saída do topo da coluna de destilação apresentada na seção 3, atestando o funcionamento da interface de comunicação. Neste processo, o controlador PI de temperatura TC117, conforme Figura 2, foi substituído pelo controlador PID 2DOF citado anteriormente.

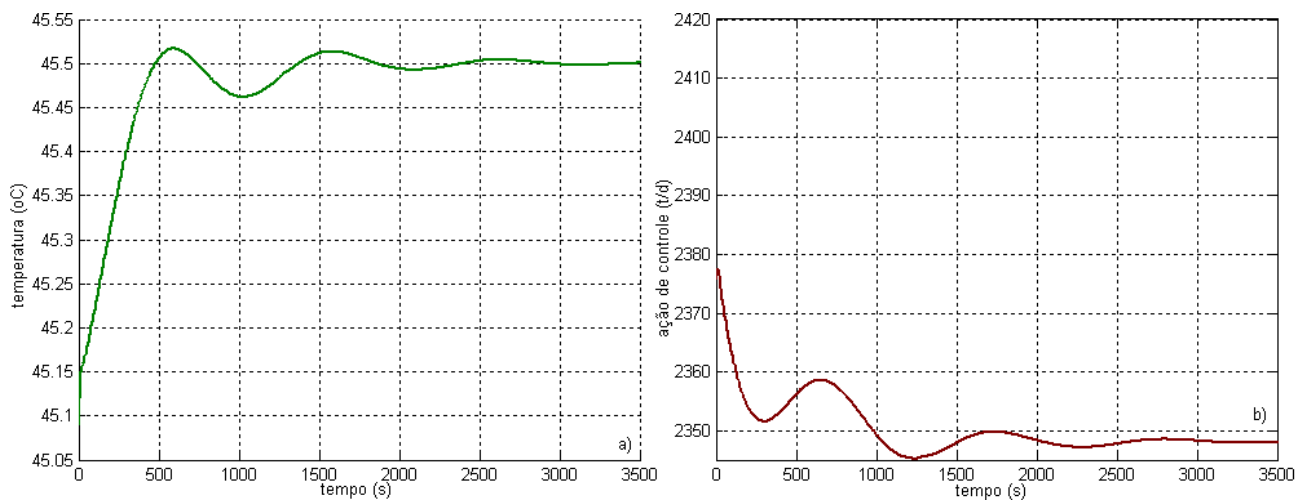


Figura 5. a) variável de processo – temperatura; b) ação de controle

Outro exemplo de aplicação da interface ocorreu no estudo da coluna de destilação (descrita em 2.2). A interface foi utilizada para obtenção e armazenamento dos dados do processo e concomitante alteração de condições de operação. O objetivo final, neste caso, foi o treinamento de uma Rede Neural para posterior controle avançado da coluna. O desenvolvimento de um sistema de comunicação entre os dois softwares permitiu avaliar o processo e suas modificações. O MATLAB iniciava o processo simulado no INDISS; 500 variáveis eram amostradas a cada 2 segundos de tempo de máquina e periodicamente modificações eram impostas ao processo. Foi observado que a comunicação entre o MATLAB e o INDISS não prejudicou o desempenho da simulação, tanto em termos de velocidade de cálculo e armazenamento quanto à fidelidade dos dados.

A inclusão no INDISS de um servidor OLE permite uma fácil comunicação com outros programas, diretamente ou através de alguma interface intermediária, como o caso do controle ActiveX no MATLAB. A principal dificuldade na comunicação com o simulador é a manipulação do tempo. A interface OLE não tem funções específicas para o avanço controlado da simulação, o que dificulta a sincronização do simulador com os programas que o acessam.

Em ambos os casos estudados, a interface entre o MATLAB e o INDISS funcionou satisfatoriamente mostrando que pode ser utilizada com sucesso em trabalhos de pesquisa e desenvolvimento, combinando as excelentes características de simulação de processos do INDISS e a versatilidade de programação do MATLAB.

## 6. Agradecimentos

Os autores agradecem o apoio da Agência Nacional do Petróleo - ANP através do programa PRH-ACIPG 34, e do CNPq-CTPetro projeto 460214/01-2 e a empresa RSI pela concessão do software INDISS.

## 7. Referências

- BRAUNSCHWEIG B., PAEN D., ROUX P., VACHER P., INTITUT FRANÇAIS DU PÉTROLE, RSI. The use of CAPE-OPEN interfaces for interoperability of Unit Operations and Thermodynamic Packages in Process Modelling. ERTC Computing, Paris, France. 2002.
- CAMACHO, E. and BORDONS, C., Model Predictive Control. Springer Verlag, 1999.
- CAPE-OPEN, [www.colan.org](http://www.colan.org), acessado em abril de 2003.
- DE PRADA, C., ACEBES, F., ALVES, R., MERINO, A., PELAYO, S., GARCIA, A., RUEDA, A., GUTIERREZ, G. & GARCIA, M. Un Simulador de Alcance Total para la Formación de los Operarios de Sala de Control de Factorias Azucareras. II Taller Iberoamericano de Informática Industrial, Octubre 2002.
- EcosimPro by EA Internaciona, Dynamic Modeling & Simulation Tool [www.ecosimpro.com](http://www.ecosimpro.com), acessado em abril de 2003.
- Manuais do simulador INDISS: Presentation, Graphic Editor, Operating the simulator, Process Diagram Builder, Quick Refereces for Instructor Mode, PID Conroler User Manual. 2002.
- Microsoft MSDN, [msdn.microsoft.com](http://msdn.microsoft.com), acessado em abril de 2003.
- RSI, [www.rsi-france.com](http://www.rsi-france.com), acessado em abril de 2003.
- The MathWorks, [www.mathworks.com](http://www.mathworks.com), acessado em abril de 2003.