



[página inicial](#) | [resumos expandidos](#) | [índice onomástico](#)

# *Sistema de comunicação OPC para uma coluna de destilação piloto*

Ronald Coutinho da Silva\*  
Adelson Siqueira Carvalho\*\*  
Dênis Barbosa do Nascimento\*\*\*

A proposta deste trabalho é apresentar duas diferentes formas de comunicação entre uma coluna de destilação piloto e o software Matlab®, permitindo, desta forma, que as variáveis do sistema possam ser visualizadas e modificadas pelo software e, conseqüentemente, permitiria a utilização de sistemas de controle avançado. Dois protocolos de troca dinâmica de dados são utilizados bem como os comandos e ferramentas para realizar a comunicação. Os resultados demonstram que, através da utilização da metodologia, podem-se desenvolver sistemas de controle adequado às necessidades dos sistemas produtivos industriais, tais como colunas e plantas de destilação.

**Palavras-chave:** Colunas de destilação. DDE. OPC. *Foundation Fieldbus*.

## ***1 Introdução***

Com o passar dos anos, os processos industriais foram se tornando cada vez mais complexos. Conseqüentemente os sistemas de automação evoluíram. Essa evolução tornou ainda mais complexa a determinação dos equipamentos e do tipo de controle que será utilizado. Com isso houve a necessidade de aprimoramento das técnicas de controle.

O desvio entre as técnicas de controle convencionais, há muito utilizadas nos processos industriais, e a real necessidade de tais sistemas não lineares motivam a utilização de sistemas de controle avançado utilizando inteligência artificial.

Todavia tais sistemas de controle avançados demandam qualificação da mão-de-obra, além do fato de não serem bem aceitos em ambiente industrial. Outro fator comprometedor destas tecnologias é o desinteresse por parte dos fabricantes de instrumentos no encapsulamento dos programas baseados em sistemas inteligentes nos produtos.

O desenvolvimento de sistemas de controle inteligente, e posterior aceitação de seus resultados na indústria, será possível à medida que minimizarmos a distância entre modelos de simulação e os processos industriais. Tal procedimento requer mais uma vez alta qualificação técnica e esforço matemático.

Para superar tais inconvenientes, pode-se adotar uma metodologia baseada na integração de *softwares* de projeto e simulação de sistemas de controle avançado com sistemas de instrumentação característicos de processos industriais.

---

\* Aluno do Curso Superior de Tecnologia em Automação Industrial do CEFET Campos. Bolsista de IC do projeto OCDEE vinculado ao NPO. E-mail: coutinhods@yahoo.com.br.

\*\* Mestre em Engenharia de Produção/UENF. Professor do Curso de automação do CEFET Campos. Orientador do projeto de IC - OCDEE). Pesquisador do Núcleo de Pesquisa Operacional (NPO). E-mail: adelsonsc@yahoo.com.br; acarval@cefetcampos.br.

\*\*\* Aluno do Curso Superior de Tecnologia em Automação Industrial do CEFET Campos. Bolsista de IC do projeto OCDEE vinculado ao NPO. E-mail: denisbn01@yahoo.com.br.

A proposta deste trabalho é apresentar duas diferentes formas de comunicação entre uma coluna de destilação piloto e o *software* Matlab®, permitindo, desta forma, que as variáveis do sistema possam ser visualizadas e modificadas pelo *software* e conseqüentemente a utilização de sistemas de controle avançado.

O artigo está organizado da seguinte forma: na seção 2, são apresentados a coluna de destilação didática e seu sistema de aquisição de dados; na seção 3, o estado da arte da pesquisa. Nas seções 4 e 5, são apresentados o ferramental teórico e a metodologia proposta para aplicação ao processo. Na seção 6, os resultados obtidos; e na seção 7, a conclusão do trabalho de pesquisa.

## ***2 Descrição do processo***

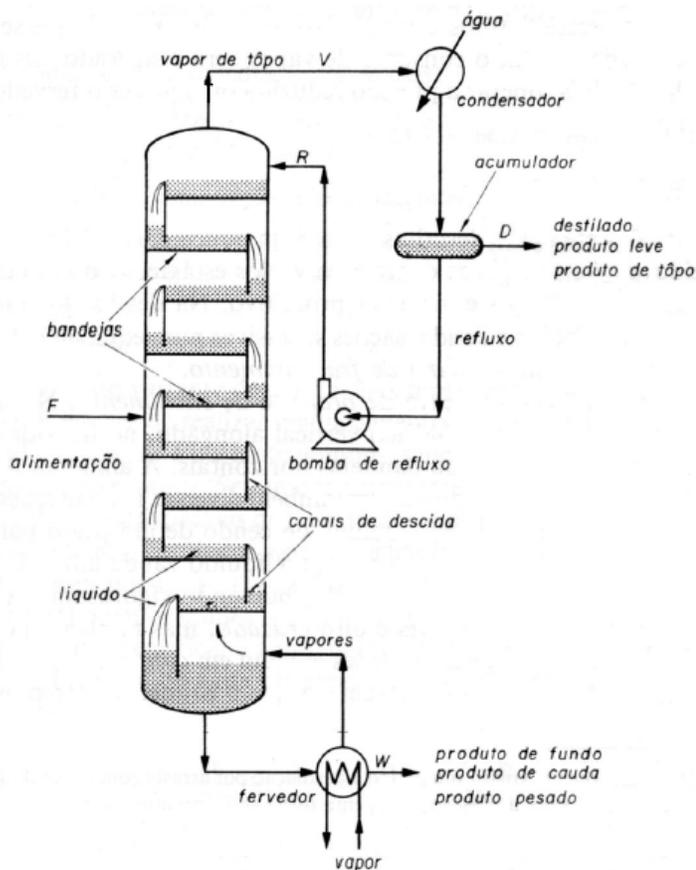
O processo de destilação introduzido na Europa pelos Árabes na Idade Média é utilizado para separar líquidos homogêneos. Este processo basea-se na diferença de volatilidade entre os líquidos, ou seja, na capacidade de um líquido passar para a fase gasosa. Portanto, deve haver uma diferença de volatilidade entre os líquidos envolvidos no processo de destilação.

Uma coluna de destilação compreende uma torre, a qual é formada por pratos ou bandejas perfurados que permitem a troca de calor entre a fase líquida e a gasosa. A coluna de destilação também conta com um sistema de aquecimento que irá fornecer calor ao sistema para que a mistura possa entrar em ebulição.

Internamente a coluna descreve dois fluxos: um iniciado do ponto de alimentação da coluna, na fase líquida, orientada para baixo, onde o teor alcoólico da mistura vai diminuindo com o conseqüente aumento do ponto de ebulição, até que este alcance a base da coluna com uma pequena concentração alcoólica; e outro fluxo partindo do refeedor (base da coluna) orientado para cima, no qual o teor alcoólico da mistura aumenta, até que este alcance a temperatura de 78,2°C, na fase gasosa, onde estará praticamente isenta de água.

Entretanto, quando este vapor é resfriado e condensado, o líquido gerado será mais rico no componente mais volátil do que o líquido que lhe deu origem. Para que o processo atinja alto grau de separação, faz-se necessário o uso de uma coluna de destilação.

Na Figura 1, é apresentada a coluna de destilação básica, conforme Gomide (1988).



**Figura 1: Coluna de destilação**  
 Fonte: GOMIDE, 1988.

A coluna de destilação é um dos equipamentos de separação mais empregados na indústria química e petroquímica. Também é o equipamento que impede o aumento da produção. Uma das formas de solucionar esse problema passa pelo aperfeiçoamento do sistema de controle.

De um modo geral, as destilações, na prática industrial, envolvem misturas multicomponentes, porém a compreensão dos princípios da destilação de misturas binárias é particularmente importante, pois ela constitui a base de operações mais complexas, como afirma Gomide (1988).

Do ponto de vista operacional dentro de uma refinaria, a coluna de destilação não atua como um processo isolado, ou seja, faz parte de um complexo conjunto de processos no qual suas correntes de saída são correntes de entrada de outras unidades.

Do ponto de vista de controle, a coluna de destilação pode ser classificada como um processo com restrições, acoplado, não estacionário e não linear. Além disso possui uma dinâmica muito lenta e a presença de tempo morto na ordem de 3 a 6 segundos por prato. Em virtude deste fato, a coluna tem um transiente muito longo quando uma alteração em alguma de suas variáveis de entrada perturba o sistema. Este fato é extremamente desagradável, pois além de consumir uma grande quantidade de energia, durante este período transiente a coluna gera derivados fora de especificação que irão afetar a carga de outros processos que estão acoplados a ela, conforme avalia Panissa (2003).

Diferente dos sistemas industriais, a coluna de destilação em questão sobrepõe em uma única torre, duas colunas: uma coluna A de destilação e uma coluna B de retificação (SILVEIRA; HAERLES, 2004). Não apresenta subsistemas como refeedor da base da coluna e refluxo de topo. A fonte térmica é uma resistência elétrica de 1000W, alojada no interior da base da coluna. Todo o destilado obtido se condensa e é coletado em uma proveta, portanto a temperatura de topo é controlada através da vazão na entrada, uma vez que a resistência elétrica não varia (CRESPO, 2000).

Com o objetivo de reproduzir em laboratório o ambiente encontrado na indústria, a coluna de destilação possui um sistema de aquisição de dados de porte industrial. Os instrumentos de medição das variáveis do processo são interligados em uma rede Foundation Fieldbus (H1) do fabricante Smar®. Através de uma DFI, a rede de instrumentos se integra a uma rede ponto-a-ponto com o computador (HSE), onde estão disponibilizadas as informações oriundas do processo e estas podem ser monitoradas e modificadas através do *software* de configuração dos instrumentos (Syscon®).

Através de uma integração entre os aplicativos Syscon®, OPC link® e InTouch 6.0®, um script foi criado para registrar os valores numéricos das variáveis, ao longo do tempo, em arquivo texto. Este arquivo é manipulado posteriormente para que seja montada a matriz de dados para formar os conjuntos de estimação e validação dos modelos paramétricos a fim de gerar o modelo do sistema (CARVALHO; GUILLERMO, 2007).

Na Figura 2, é apresentada a imagem da coluna de destilação piloto.



**Figura 2:** Coluna de destilação piloto do laboratório de pesquisa e desenvolvimento em automação do CEFET Campos

### ***3 Estado da arte***

Processos industriais como o de destilação são sistemas dinâmicos, multivariáveis e de comportamento não-linear. Seu controle é possível lançando-se mão de técnicas avançadas de controle que exigem comunicação e integração entre sistemas computacionais. Este problema de comunicação abre caminho para aplicações diversas que estão disponíveis na forma de textos científicos.

Péres *et al.* (2003) descreveram a aplicação de uma ferramenta do Matlab® (xPC Targed Toolbox), em práticas de laboratório, como complemento na docência em aulas de controle. O objetivo destas aulas práticas era que o aluno aplicasse os conhecimentos adquiridos nas aulas teóricas em problemas típicos que aparecem na identificação de processos e na implementação de controladores digitais para sistemas reais. O sistema xPC targed permite executar em tempo real uma aplicação do Simulink® em um PC de escritório. Isto nos permite, de uma forma simples e rápida, aplicar modelos de processos arbitrariamente complexos no Simulink®, os quais poderiam ser identificados e controlados pelos alunos.

Uberti *et al.* (2003) apresentaram uma forma de controle adaptativo via rede Fieldbus, utilizando múltiplos modelos e lógica difusa como ferramenta capaz de suavizar a ação do controlador na passagem de uma faixa de operação para outra. A técnica apresentada pode ser aplicada a processos da indústria de petróleo e gás, caracterizada pela predominância de processos não lineares. Consideraram que atualmente existe uma tendência para uma crescente utilização de controle e monitoração de processos via rede industrial do tipo *Fieldbus*. Esses autores mostraram uma aplicação da técnica, efetuando o controle de nível em uma planta didática, baseada em uma rede *Foundation Fieldbus*. Utilizaram o OPC para comunicação entre a planta e o Matlab®, e por meio desta conseguiram implementar uma formulação matemática utilizando os valores dos parâmetros dos três controladores encontrados.

Gonçalves (2004) pesquisou sobre a comunicação do Matlab® com o sistema supervisor via DDE, mas optou pelo uso de scripts devido a sua simplicidade e pouco uso de memória. Utilizou este tipo de comunicação, na simulação dinâmica de modelos de sistemas elétricos de potência em diversos cenários e na verificação experimental do comportamento dos grupos geradores em diversas condições de operação, incluindo a análise de desempenho estático, ensaios de curto circuito e as variações de despacho ativo e reativo de potência. Obteve com isso uma melhor identificação e quantificação do impacto dos produtores independentes e autoprodutores na análise de sistemas elétricos de potência.

Duarte *et al.* (2006) apresentaram a utilização do *software* Matlab® 7.0 no ensino da tecnologia OPC aplicada a controle de processos, mais precisamente no controle de uma malha de vazão. Destacaram a importância da aprendizagem das tecnologias utilizadas na indústria para a formação dos futuros engenheiros, aproximando teoria e prática.

Pastorello (2008) propôs a criação de um ambiente virtual para testes por ser economicamente mais viável e ao mesmo tempo mais seguro. Utilizou uma coluna de destilação de mistura binária, onde

se necessita controlar as composições de topo e de fundo. Trata-se de uma coluna simulada, modelada em Matlab®. Neste ambiente, utilizou-se um controlador preditivo comercial (*DeltaV Predict*) que faz parte do sistema de controle Delta V do fabricante Emerson Process®. A integração do modelo da coluna de destilação e do DeltaV Predict® foi feita através da ferramenta OPC, normalmente utilizada para integração de sistemas.

## **4 Ferramental teórico**

### **4.1 DDE**

O protocolo *Dynamic Data Exchange* (DDE) é composto por um conjunto de mensagens e conceitos. O DDE envia mensagens entre aplicações que partilham dados e usa memória partilhada para efetuar a troca de dados entre aplicações. As aplicações podem usar o DDE tanto para fazer uma troca de dados isolada, como para efetuar trocas contínuas de dados em que as aplicações enviam atualizações à medida que os novos dados vão surgindo.

O DDE apareceu pela primeira vez no *Windows 3.0*. Surgiu como a primeira forma de comunicação entre processos e ainda hoje é suportado e usado por algumas aplicações. Apesar de ser um processo limitado, não muito fácil de ser corretamente implementado e vagaroso, o DDE foi o primeiro passo no desenvolvimento de diversas tecnologias de integração e comunicação.

### **4.2 OLE**

A tecnologia *Object Linking and Embedding* (OLE) foi desenvolvida pela Microsoft, em meados de 1990, para suprir a necessidade de se integrarem diferentes aplicações dentro da plataforma Windows, de forma a solucionar os problemas de desempenho e confiabilidade do até então utilizado padrão DDE Silva (2005). Nessa época, introduziram-se dois conceitos:

- *Linking* – Cria vínculos ou referências aos objetos, armazenando no documento principal apenas os dados realmente necessários para exibir, imprimir, etc.;
- *Embedding* – Incorpora os dados dos objetos ao documento principal. Neste contexto, surgiram os conceitos de objeto vinculado e do objeto incorporado: *Objeto Vinculado* – São informações (objetos) criadas em um arquivo (arquivo origem) e inseridas em outro arquivo (arquivo destino). Embora o objeto vinculado não se torne parte do arquivo de destino, existe um vínculo, uma conexão entre os dois arquivos de forma que o objeto vinculado no arquivo de destino seja automaticamente atualizado quando o arquivo de origem é atualizado. *Objeto incorporado* – São informações inseridas em um arquivo de destino. Ao ser incorporado, o objeto se torna parte do arquivo. Ao clicar duas vezes no objeto incorporado, ele é aberto no programa de origem em que foi criado. Qualquer alteração feita no objeto incorporado se refletirá no arquivo de destino.

Outro conceito importante na tecnologia OLE é o conceito de Cliente Servidor:

- ✓ cliente – Aplicação que solicita os dados;
- ✓ servidor – Aplicação que disponibiliza os dados.

No OLE 2.0, houve muitos aperfeiçoamentos dentre os quais o mais importante é a automação OLE, pois permite que uma aplicação seja controlada por outra aplicação.

A tecnologia OLE é montada sobre a tecnologia *Component Object Model* (COM) que define um modo padronizado para um módulo-cliente e um módulo-servidor se comunicarem por meio de uma interface específica. Módulo indica um aplicativo ou uma biblioteca uma *Dynamic Link Libraries* (DLL). Os dois módulos podem ser executados no mesmo computador ou em máquinas diferentes conectadas através de uma rede. Muitas interfaces são possíveis, dependendo do papel do cliente e do servidor, e podem-se incluir novas interfaces para propósitos específicos.

### 4.3 OPC

Os controles de processos industriais têm se tornado cada vez mais complexos devido à exigência de qualidade dos produtos, rapidez na entrega e concorrência de mercado. Tal complexidade produz grandes quantidades de dados a serem gerenciados pelos três níveis de controle de processos: dispositivos de campo, sistemas de controle e *softwares* para gerenciamento e negócios. A integração desses níveis, utilizando efetivamente a informação disponível em cada um deles, é de fundamental importância para a indústria.

Na busca de uma solução para esse problema, foi desenvolvida a tecnologia OPC, que é uma tecnologia para conectar aplicações Windows e equipamentos de controle de processos. O OPC é um protocolo de comunicação aberto que permite um método consistente de acesso aos dados de inúmeros equipamentos dos mais diversos fabricantes. O método é o mesmo, independente da origem dos dados, o que vem oferecer ao usuário final uma maior liberdade na escolha dos equipamentos independentemente da disponibilidade de *drivers* de comunicação proprietários. O OPC é construído usando tecnologia Microsoft OLE/COM, mas a especificação OPC foi desenvolvida por uma fundação aberta, a OPC Foundation, para atender as necessidades gerais da indústria e não as necessidades específicas de alguns fabricantes de hardware e *software* (OPC FOUNDATION, 1988). A especificação ainda prevê a evolução das funcionalidades ao longo do tempo e por isso, os componentes OPC podem se manter no topo das necessidades emergentes da indústria.

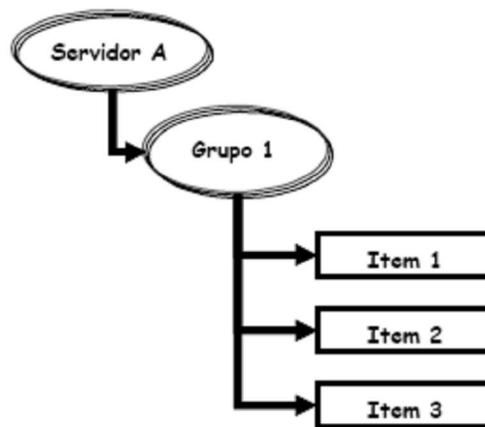
Os componentes OPC se classificam em duas categorias: Clientes OPC e Servidores OPC. Um Cliente OPC é tipicamente um usuário dos dados tais como uma Interface de Operação ou um Sistema supervisório (SCADA). Um Servidor OPC é uma fonte de dados que coleta ou gera dados a partir de um processo, disponibilizando-os aos Clientes OPC. O Cliente OPC interage com o Servidor OPC usando uma interface bem definida. Qualquer Cliente OPC pode se comunicar com qualquer Servidor

OPC, independentemente do tipo de dispositivo e do fabricante. Essa comunicação é válida somente para OPC-DA (Data Access), uma vez que existem diferentes tecnologias OPC.

Basicamente, o padrão OPC estabelece as regras para que sejam desenvolvidos sistemas com interfaces padrões para comunicação dos dispositivos de campo (controladores, sensores, etc.) com sistemas de monitoração, supervisão e gerenciamento (SCADA, MÊS, ERP, etc.). (Retirado de DUARTE, 2006).

#### 4.4 OPCTOOL®

A arquitetura OPC pressupõe três objetos básicos: servidor, grupo e item, apresentados na Figura 3. Do ponto de vista do cliente, um servidor é essencialmente uma estrutura de armazenagem para grupos que, por sua vez, têm como função básica o armazenamento de itens. Esses itens, elementos mais simples na especificação, representam conexões a pontos de entrada ou saída. Assim, o item OPC não é um valor, mas apenas um meio de acesso a um valor. Desta forma, uma única variável de entrada ou saída pode ser representada por itens diferentes, com propriedades distintas e compartilhada por mais de um cliente. É tarefa dos grupos reunir o conjunto de itens que interessam a um determinado cliente, assumindo o papel principal na interação do cliente-servidor.



**Figura 3: Arquitetura OPC**  
Fonte: DUARTE, 2006.

Os grupos também são responsáveis por satisfazer pedidos de leitura e escrita, bem como enviar atualizações para seus clientes, periodicamente ou por exceção. Essas transações de atualização podem ser ativadas ou desativadas no grupo ou nos itens individuais.

Os grupos presentes em um servidor OPC são normalmente definidos pelos clientes, e somente o cliente criador do grupo pode acessá-lo; tal tipo de grupo é dito privado. Em alguns casos, porém, pode ser interessante que o servidor ofereça grupos passíveis de serem compartilhados por vários clientes. Quando essa capacidade é desejada, implementa-se a funcionalidade opcional dos grupos públicos.

Do ponto de vista do cliente, a função básica do servidor é prover uma infra-estrutura de suporte aos grupos. Além disso, cabe também a ele gerenciar aspectos relacionados à conexão com uma fonte de dados, tais como parâmetros de comunicação ou taxa máxima de amostragem. Outra responsabilidade do servidor é implementar uma estrutura de endereçamento capaz de associar itens com variáveis reais.

O item é uma estrutura à qual estão associadas três propriedades:

- *value*: último valor armazenado pelo servidor no cache de memória do item e que é atualizado sempre que o servidor faz uma leitura no dispositivo;
- *quality*: informação de estado que define a qualidade do dado que pode ser: *Good* – dado válido; *Bad* – perda do link de comunicação com o dispositivo de campo; e *Uncertain* – no caso de existir o link e o dispositivo de campo estiver fora de comunicação;
- *time Stamp*: tempo de amostragem.

O *OPC Toolbox* é uma coleção das funções que estendem a potencialidade do Matlab®. Usando funções do toolbox, você cria um objeto cliente para acesso aos dados (objeto cliente do *opcda*) que represente a conexão entre o Matlab® e um usuário OPC. Usando propriedades do objeto cliente, você pode controlar vários aspectos da comunicação, tais como períodos de tempo, status da conexão, e armazenamento dos eventos associados a este cliente.

Antes que você interaja com os usuários OPC usando o *OPC Toolbox*, deve compreender o relacionamento entre usuários e clientes OPC. Exemplos desse relacionamento são: como os usuários OPC organizam suas variáveis e como os clientes podem interagir com aqueles dados do usuário.

Usando o *OPC Toolbox*, você pode registrar os dados (uma lista dos dados que mudaram, e seus valores novos) de um usuário no disco rígido ou numa memória, para mais tarde processar. A tarefa de registro dos dados é controlada pelo objeto do *datagroup*. Para trabalhar com os dados que você adquire, deve trazê-los para o espaço do Matlab®. O toolbox fornece diversas maneiras para trazer um ou mais registros dos dados para o espaço de trabalho onde você pode analisar ou visualizar esses dados.

O toolbox definiu determinadas ocorrências do *OPC Toolbox*, tais como o começo de uma tarefa de aquisição; e também ocorrências iniciadas de um usuário, como, por exemplo, a notificação de que o estado de uma variável mudou, como eventos. Você pode associar a execução de uma função particular a um evento particular.

## **5 Metodologia**

### **5.1 DDE Matlab® / INTOUCH®**

A integração entre os sistemas Matlab® (simulação e controle) e InTouch® (supervisão do processo) será realizada em um primeiro momento através de comandos DDE do Matlab. Neste tipo de

aplicação específica, o Matlab® funcionará como *software* cliente da comunicação; e o InTouch®, como servidor.

Esta configuração faz-se necessária, pois o InTouch® como *software* de supervisão do processo recebe do mesmo os sinais das variáveis controladas do processo e de forma análoga pode enviar sinais para as variáveis manipuladas do sistema atuando nos elementos finais de controle.

Para o estabelecimento inicial da conexão utilizando o protocolo DDE, faz-se necessário utilizar o comando `ddeinit` no Matlab®, cuja sintaxe é:

`a=ddeinit ('nome da aplicação', 'nome da sessão')`

Onde **a** é a variável que armazena a resposta à requisição por conexão (afirmativa ou negativa); nome da aplicação é o nome da aplicação servidora e nome da sessão se refere à sessão da aplicação servidora que será acessada.

Este comando retorna, caso a conexão seja estabelecida, a um valor diferente de “0”, que certifica que um canal virtual foi estabelecido entre as duas aplicações.

Um segundo comando será executado para depois de estabelecida a conexão. Trata-se do `ddereq`. Este comando solicita ao servidor o valor contido em uma dada variável da sua sessão e retorna para a variável que armazena dentro do Matlab®.

`b=ddereq ('canal', 'nome da variável')`.

Onde **b** é a variável que armazena a resposta à requisição por valor da variável controlada; **canal** é o nome da variável que recebeu o retorno do pedido de conexão e **nome da variável** se refere à variável controlada que está sendo medida e disponibilizada no processo.

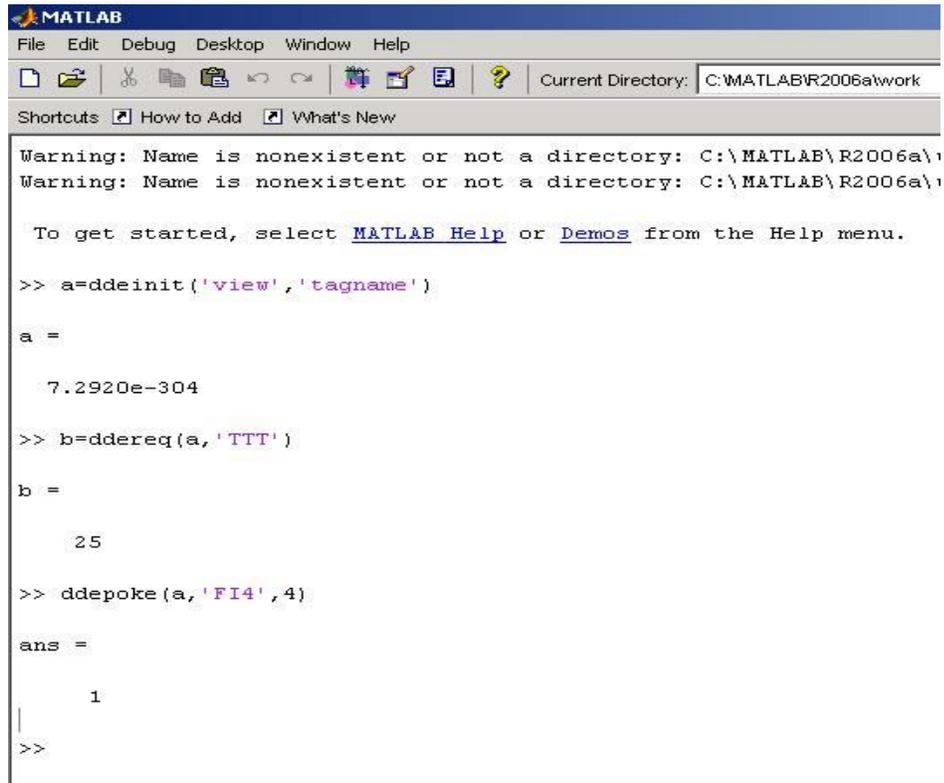
Tendo lido o valor de uma variável controlada através do comando `ddereq`, faz-se necessário tomar uma ação de controle que gera um valor numérico a ser alterado na aplicação servidora. Desta forma, é possível a utilização de um comando `ddepoke` que tem a função de realizar esta alteração, uma vez estabelecida a comunicação através do `ddeinit`.

`ddepoke ('canal', 'nome da variável', 'valor')`.

Onde o **canal** e o **nome da variável** já foram esclarecidos anteriormente e o **valor** é o que será escrito na variável manipulada na sessão da aplicação servidora.

Com estes três comandos básicos, é possível realizar o ciclo básico de um sistema de controle no qual a função dos sensores é realizada pelo comando `ddereq` e a função dos atuadores realizada pelo comando `ddepoke`.

Na Figura 4, é apresentada a tela com a seqüência de comandos executada no Matlab®, e respectivos resultados, para abertura de uma conexão, leitura do valor de uma variável e escrita de um valor em uma variável, adequados aos *softwares* envolvidos neste trabalho.



```
MATLAB
File Edit Debug Desktop Window Help
Current Directory: C:\MATLAB\R2006a\work
Shortcuts How to Add What's New
Warning: Name is nonexistent or not a directory: C:\MATLAB\R2006a\
Warning: Name is nonexistent or not a directory: C:\MATLAB\R2006a\
To get started, select MATLAB Help or Demos from the Help menu.
>> a=ddeinit('view','tagname')
a =
    7.2920e-304
>> b=ddereq(a,'TTT')
b =
    25
>> ddepoke(a,'FI4',4)
ans =
    1
>>
```

Figura 4: Tela do Matlab® com os comandos para conexão com o InTouch® via DDE

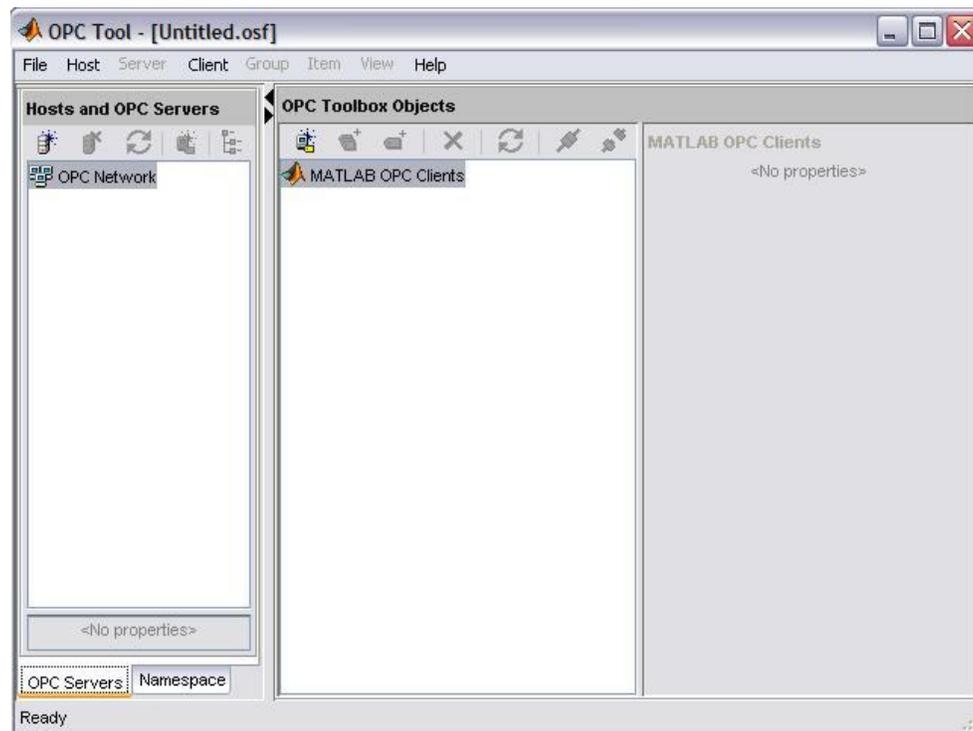
## 5.2 OPC Matlab®/ SYSCON®

A mesma operação de conexão virtual realizada entre InTouch® e Matlab® é possível através do protocolo de comunicação OPC. A diferença é que as aplicações envolvidas neste processo serão os *softwares* Matlab®, operando como cliente na comunicação e o Syscon® funcionando como servidor. Este *software* é responsável pela configuração de redes *Foundation Fieldbus* e como tal se conecta aos dispositivos de campo através de uma rede de alta velocidade denominada HSE, que trabalha de acordo com o modelo TCP/IP, a DFI (dispositivo que comunica os instrumentos com o PC) e possui uma interface de rede padrão e um endereço IP. Basta que o IP da máquina na qual o Syscon® esteja instalado seja da mesma classe e rede que o da DFI para que se comuniquem. Os parâmetros e valores dos instrumentos podem então ser alterados on-line pelo Syscon® através de um servidor OPC que se conecta a DFI para aquisição dos dados da rede *fieldbus*.

Aproveitando-se deste servidor OPC instalado e sendo executado no computador que supervisiona o processo, o Matlab® pode estabelecer uma conexão virtual direto com estes servidores, dispensando, assim, a presença do *software* de supervisão InTouch®.

Isto é possível graças à incorporação do *toolbox* OPC em versões mais recentes do Matlab® e sua interface gráfica denominada *opctool*®, que abre uma sessão de conexão via OPC ao detectar

qualquer servidor OPC instalado e executado no computador onde o Matlab® está instalado. Na figura 5, é apresentada a tela da ferramenta *opctool*® do Matlab®.

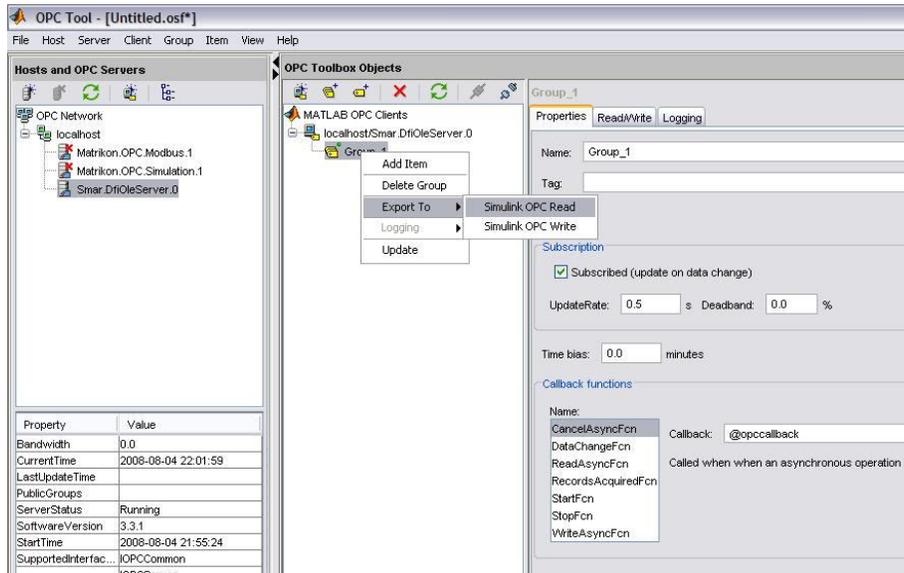


**Figura 5: Tela inicial da ferramenta de comunicação OPCTOOL®**

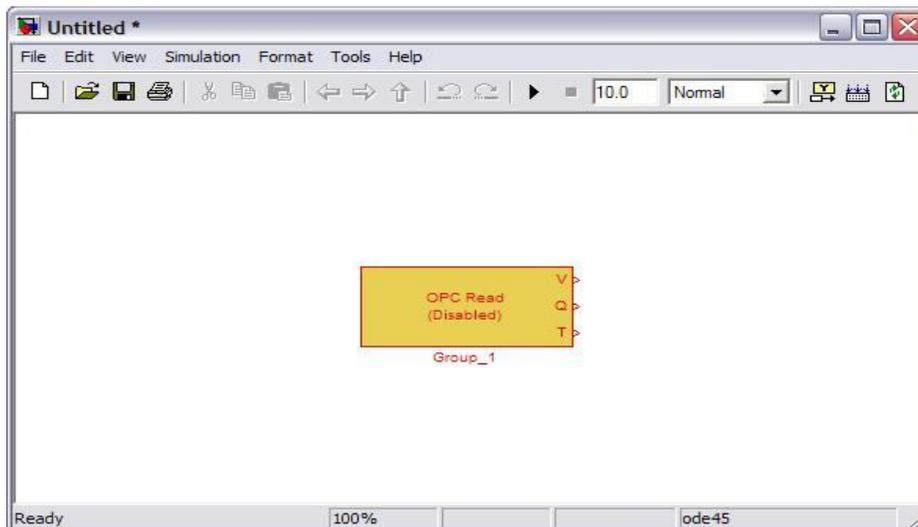
Esta ferramenta trabalha com conceitos como:

- *hosts* – computadores com servidores OPC instalados;
- *OPC servers* – servidores de comunicação, via protocolo OPC, utilizado em equipamentos industriais;
- *matlab OPC Clients* – agentes clientes OPC que disponibilizarão os dados no ambiente do Matlab®;
- *group* – conjunto de itens OPC;
- *item* – variável ou parâmetro que será lido ou escrito pelo Matlab®.

Na tela apresentada na Figura 6, está selecionada a opção de geração de um arquivo do SIMULINK® a partir da opção do menu na tela da ferramenta *opctool*. Tanto a seleção da opção *write* como na *read*, um novo arquivo .mdl é gerado, conforme Figura 7.



**Figura 6:** Tela de exportação para modelos do SIMULINK®

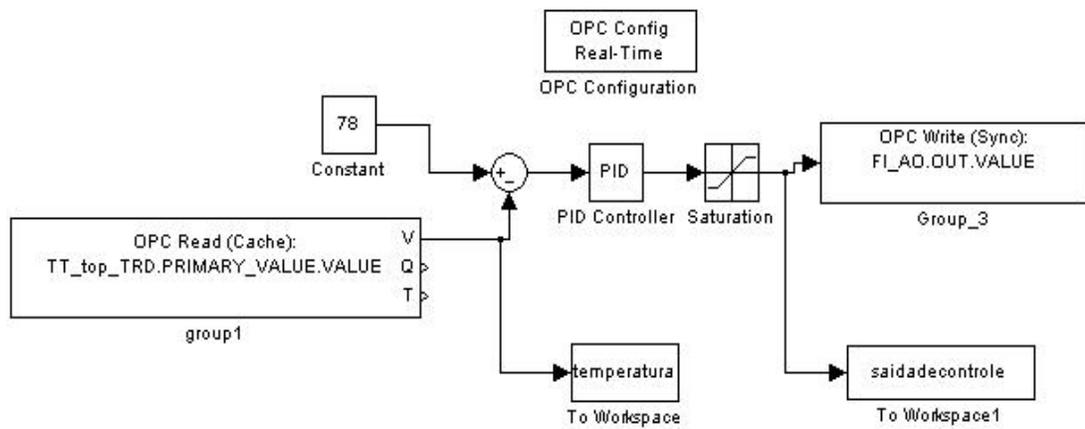


**Figura 7:** Modelo .mdl gerado a partir do menu *Export to > Simulink OPC Read*

## 6 Resultados

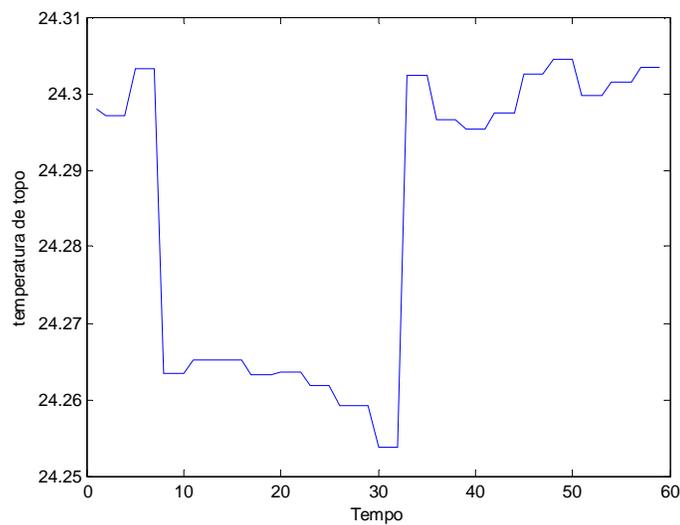
Os resultados deste trabalho circundam a implementação do sistema de comunicação proposto na coluna de destilação piloto. As características inerentes aos comandos necessários, bem como a integração da ferramenta *opctool* com o simulink para a geração do modelo de simulação já foram elucidadas na seção 5. Cabe, então, a apresentação do modelo de simulação utilizado e dos gráficos referentes aos sinais enviados e recebidos do processo pelo sistema de comunicação.

Na Figura 8, é apresentado o modelo de simulação comunicando com o sistema real via protocolo OPC.

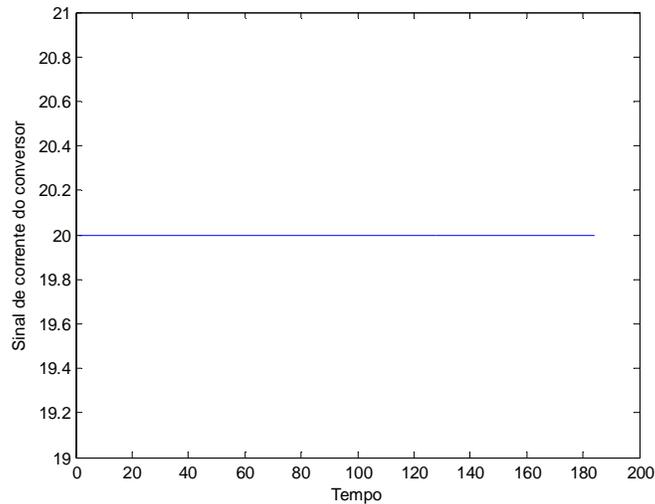


**Figura 8: Modelo de simulação comunicando com o sistema real**

Nas figuras 9 e 10 são apresentados os sinais registrados pelo sistema de comunicação OPC referente a temperatura de topo e ao sinal de controle enviado para o elemento final de controle.



**Figura 9: Sinal captado pelo sistema de comunicação OPC para a temperatura de topo**



**Figura 10: Sinal captado pelo sistema de comunicação OPC para a saída de controle**

## **7 Conclusão**

A partir dos resultados descritos na seção 6, é possível verificar que a integração entre ferramentas computacionais de diferentes fabricantes, se integradas da forma correta, pode propiciar o surgimento de uma plataforma de comunicação fundamental para o desenvolvimento de sistemas de controle avançados.

As vantagens do Matlab®, no projeto e simulação de sistemas de controle, é fundamental para sua utilização como *software* que realiza na prática o controle de uma planta de processo, seja esta em escala laboratorial ou industrial.

A integração da coluna de destilação ao Matlab® via OPC apresenta-se como uma boa alternativa para controle de processos industriais mediado por computador. A facilidade na criação de sistemas de controle por parte do Matlab® proporciona uma interface mais amigável para o engenheiro de controle, uma vez que se utilizam conceitos conhecidos de disciplinas teóricas de sua formação como controle clássico e controle avançado, baseados em diagramas de blocos e fluxo de sinal.

Como sugestão para trabalhos futuros cabe evidenciar a utilização da metodologia aqui referida para aplicação em um sistema real com instrumentação industrial como a coluna de destilação, e posterior sintonia do sistema de controle uma vez que o exemplo apresentado na seção 6 é meramente ilustrativo e tal aplicação foge ao escopo deste trabalho.

## **Referências**

CARVALHO, A. S.; GUILLERMO, L. H. F. Modelagem de colunas de destilação através de modelos auto-regressivos, SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA, Resende, 2007.

CRESPO, L. S. *Montagem identificação e modelamento de uma torre de destilação piloto*. Dissertação (Mestrado em Automação) – Universidade Federal do Espírito Santo, Vitória, 2000.

DUARTE, C. R. M.; FIGUEIREDO, L. C.; CORRÊA, M. V. Utilização do Matlab® no ensino da tecnologia OPC aplicada a controle de processos. *Anais do XVI Congresso Brasileiro de Automática*, Salvador, 2006.

GOMIDE, R. *Operações unitárias*. V. IV. São Paulo: Edição do Autor, 1988.

GONÇALVES, L. F. *Contribuição para o estudo teórico e experimental de sistemas de geração distribuída*. Dissertação (Mestrado em Engenharia Elétrica) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2004.

PANISSA, G. L. Uma nova abordagem para o controle de colunas de destilação, *Relatório Final de Bolsa* – Universidade Federal de Santa Catarina, Florianópolis, 2003.

PASTORELLO, R. *Integração via OPC de controlador preditivo comercial com simulador dinâmico de processos gerando ambiente virtual de testes*. Dissertação (Mestrado em Engenharia de Processos e Bioquímicos) – Escola de Engenharia Mauá do Centro Universitário do Instituto Mauá de Tecnologia, São Caetano do Sul, 2008.

PÉREZ, E. *et al.* Uso del Toolbox xPC Target en prácticas de control e identificación. *Seminario anual de automática, electrónica industrial e instrumentación*, Vigo (Spain), 2003.

SILVA, D. S. *et al.* Projeto de uma planta para avaliação de medições de instrumentos de vazão e BS&W. 3º CONGRESSO BRASILEIRO DE P&D EM PETRÓLEO E GÁS, Salvador, 2005.

SILVEIRA, R. D.; HAERLES, C. O. R. *Implementação de sistema de comutação de controladores com tecnologia Fieldbus Foundation em coluna de destilação didática*. Monografia (Curso de Tecnologia em Automação Industrial) – Centro Federal de Educação Tecnológica de Campos (CEFET Campos), Campos dos Goytacazes, RJ, 2004.

UBERTI, R. C.; SANTOS, R. S.; PLUCENIO, A. Controle adaptativo utilizando múltiplos modelos em uma rede Fieldbus. 2º CONGRESSO BRASILEIRO DE P&D EM PETRÓLEO & GÁS, Rio de Janeiro, 2003.



[página inicial](#) | [resumos expandidos](#) | [índice onomástico](#) | [ir para o topo](#)