# 5
# Estudo de Caso

No capítulo anterior ilustramos de forma geral como utilizar o mecanismo de replicação para replicar um sistema de gerenciamento de *workflow*. Nesse capítulo apresentaremos o estudo de caso desse mecanismo para replicação de um sistema específico. Iremos inicialmente apresentar características e componentes principais do sistema em questão, bem como modificações feitas para que esse sistema pudesse ser replicado pelo mecanismo e se tornar tolerante a falhas. O sistema em questão é o sistema MPA, desenvolvido pelo Tecgraf/PUC-Rio em parceria com o CENPES/Petrobras.

## 5.1
## O Sistema MPA

O sistema MPA foi desenvolvido visando disponibilizar um ambiente de modelagem e execução de processos produtivos para automação industrial. Nesse sistema, os processos são modelados em fluxogramas, com ações e condições. Essas ações podem descrever diferentes atividades, desde a manipulação de variáveis locais a execução de manobras em equipamentos de uma planta industrial. A interação do sistema com as plantas é feita por meio de pontes de comunicação que acessam a base de dados onde estão representados valores de sensores e atuadores. Sendo assim, o sistema MPA atua no mesmo nível dos operadores da planta. O sistema MPA é composto por uma aplicação cliente, o cliente MPA, para modelagem, controle e monitoração da execução dos processos e uma aplicação servidora, o servidor MPA, composta pelo motor de execução, pontes de comunicação e interfaces remotas de gerenciamento e monitoração da execução. Os componentes do sistema MPA estão representados na figura 5.1

### 5.1.1
### Modelagem de Processos

No sistema MPA processos são modelados em fluxogramas. Esses fluxogramas são construídos usando elementos gráficos que descrevem a forma e a ordem em que as atividades devem ser realizadas durante a execução dos

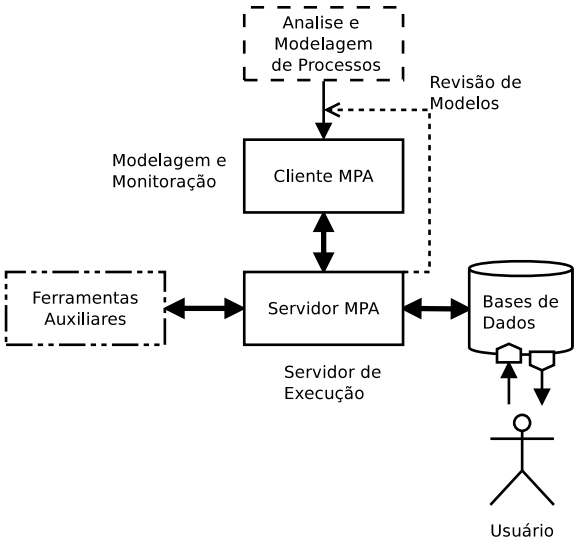


Figura 5.1: Componentes do Sistema MPA

processos. Esses elementos implementam alguns dos padrões de construções identificados para *workflows*.

Como mencionado no início desse capítulo, os fluxogramas do MPA permitem a modelagem dos processos em ações e condições. As ações no fluxograma são representadas por elementos comando e podem ser usados para atribuições a variáveis, chamadas de funções auxiliares, chamadas de fluxos, ou chamadas de funções sobre equipamentos. Esses elementos são ligados por linhas direcionadas indicando o próximo elemento que deve ser executado. A ligação entre ações representa o padrão de sequência de ações. Entretanto, o padrão de condição de ativação não é representado nos elementos gráficos do MPA. Nesses fluxogramas as condições são representadas por elementos escolha que funcionam como uma implementação do padrão de escolha múltipla que possui apenas duas opções mutuamente exclusivas de saída. Esses elementos podem ter duas saídas, uma caso o resultado da avaliação descrita seja verdadeiro e outro caso o resultado seja falso. A ação associada a uma escolha não pode ser uma atribuição, mas pode ser qualquer expressão ou uma ação que tenha um resultado lógico. É possível obter o comportamento de uma condição a partir de um elemento escolha utilizando um laço de espera na saída falsa, impedindo a realização da próxima ação até que o resultado da escolha seja verdadeiro.

Outros padrões básicos possíveis para criação de fluxogramas no MPA são implementados usando os elementos executor e sincronizador. O elemento executor permite que a saída de um elemento seja ligada a vários elementos e indica que esses elementos devem ser executados em paralelo, implementando o padrão execução em paralelo. Analogamente, o elemento sincronizador permite

que elementos finais de sequências de ações que executam em paralelo sejam ligados a um elemento único e indica que a execução desse elemento só deve ser iniciada quando todas as sequências forem concluídas, implementando o padrão de sincronização. Entretanto, esses elementos possuem uma restrição, o uso deles deve ser balanceado, ou seja, um elemento sincronizador deve ser conectado por elementos de sequências criadas por apenas um elemento executor.

Quanto aos padrões avançados, os fluxogramas do MPA podem usar ainda os padrões de construção: término implícito, cancelamento e múltiplas instâncias. Para explicar o término implícito e o cancelamento é necessário apresentar os dois tipos de fluxogramas existentes no MPA: aplicação e função.

Fluxogramas aplicação possuem apenas uma execução, ou seja, só podem ser instanciados uma vez. Essa característica permite que fluxogramas desse tipo sejam comandados e monitorados a partir de uma única instância de execução. Por apresentarem essa característica, fluxogramas desse tipo são usados para descrever processos que devem ser executados continuamente ou interrompidos como um todo. Por outro lado, os fluxogramas função podem ser instanciados inúmeras vezes, mas sempre, direta ou indiretamente, a partir da instância de um fluxo aplicação por meio de elementos de comando ou escolha. Esse tipo de fluxo permite a definição de parâmetros e a obtenção de resultados, implicando na espera pela conclusão do fluxograma executado. Os dois tipos de fluxogramas apresentam o término implícito. Fluxos aplicação só são considerados concluídos se todas as suas co-rotinas (iniciadas a partir de elementos de execução em paralelo) forem concluídas. Por outro lado, fluxogramas função são considerados concluídos uma vez que a sequência principal de ações for concluída.

Outro padrão que pode ser usado nos fluxogramas do sistema MPA a partir da definição de fluxogramas função é o padrão de cancelamento. Esse padrão é disponibilizado pelo elemento comando que, além de atribuições e chamadas de funções e métodos, permite comandar o início e o término de fluxos aplicação. Uma chamada de término irá implicar na interrupção de todas as ações que estiverem sendo executadas. Alternativamente, é possível cancelar a execução de fluxos aplicação por meio da interface programática. Isso é feito principalmente por uma aplicação cliente, mas pode ser feito por uma aplicação que conheça a localização da aplicação servidora (descrita na seção 5.1.2) e a sua interface de controle.

O padrão de múltiplas instâncias é disponibilizado pelo elemento iterador. Esse elemento tem duas ligações de saída, uma obrigatória para indicar a sequência de elementos que deve ser executada para todos os dados de entrada,

e uma opcional, a sequência que deve ser executada ao terminar a iteração sobre todos os dados de entrada. Caso não tenha a ligação de término, o fim da iteração levará ao término implícito da sequência de ações. Para um elemento iterador, os dados de entrada são instâncias de uma classe de equipamento. Essa característica decorre da utilização do elemento, repetindo o mesmo procedimento para várias instâncias de equipamento da mesma classe. Sendo assim, é possível utilizar o padrão de múltiplas instâncias se considerarmos que o fluxo de dados é representado pelos atributos das instâncias.

Além dos padrões suportados diretamente, é possível identificar adaptações que podem ser feitas para obter outros padrões de construções com os elementos de fluxogramas do MPA. Elementos do fluxograma podem ser ligados arbitrariamente, desde que não sejam de sequências de ações de um executor em paralelo ou da sequência de um iterador ligadas com elementos externos a eles. Esses elementos exigem que a sua sequência de ações estabeleça parcialmente um ciclo estruturado. Essa característica permite construções como convergência de sequências e mesclagem múltipla. Respeitando a restrição de iteradores e de execuções em paralelo, é possível definir ciclos arbitrários. É possível também obter o padrão de execução baseada em marcos utilizando escolhas e comandos de início e interrupção de fluxograma. Como os fluxogramas aplicação só podem apresentar uma instância de execução, um comando de início de um fluxograma é ignorado caso o fluxograma já esteja em execução. Sendo assim, para obter o comportamento de um marco, basta encadear escolhas cujo resultado falso é ligado a interrupção do fluxograma e o resultado verdadeiro é ligado a próxima escolha, sendo que última é ligada ao comando de início do fluxograma.

Na tabela 5.1.1 listamos os padrões de construções para *workflows* [3] que o MPA dá suporte de forma direta ou parcialmente com adaptações.

### 5.1.2 Execução de fluxogramas

Os processos, uma vez modelados, são executados no servidor MPA. Nele, o papel do motor do sistema é feito por um componente executor que disponibiliza uma interface de controle e monitoração de processos. O estado inicial do servidor é composto pelos modelos de processos, e é definido por arquivo de inicialização ou transmitido via rede a partir de um cliente MPA. Além de carregar o estado inicial, o servidor MPA é responsável por instanciar as pontes de comunicação que irão realizar o acesso às bases de dados com valores indicados nos elementos dos fluxogramas. Uma vez que todos os componentes necessários estão em operação, o início da execução dos

| Padrão | Suporte |
|---|---|
| Ordenação | Sim |
| Condição | Parcial |
| Convergência | Sim |
| Paralelo | Sim |
| Sincronização | Sim |
| Escolha múltipla | Parcial |
| Múltiplas instâncias | Sim |
| Sincronização múltipla | Não |
| Mesclagem múltipla | Sim |
| Ciclos arbitrários | Parcial |
| Ciclos estruturados | Parcial |
| Término implícito | Sim |
| Escolha Adiada | Não |
| Execução Intercalada | Não |
| Marcos | Parcial |
| Cancelamento | Sim |

Tabela 5.1: Suporte a padrões de construções de *workflows* no MPA



fluxogramas pode ser comandado por um usuário em uma aplicação cliente.

### 5.1.3 Monitoração

A monitoração da execução dos fluxogramas é feita na aplicação cliente e pode ser feita de forma textual ou visual. A monitoração textual da execução permite que o usuário indique o nível de mensagens da execução dos fluxogramas que deseja acompanhar. Existem três níveis de mensagem: erro, execução e acompanhamento. A monitoração textual de mensagens de erro consiste em acompanhar apenas erros de execução do fluxograma, como erros no acesso a base de dados ou chamadas sobre equipamentos que não tenham sido modelados corretamente. De forma complementar, a monitoração textual de mensagens de execução consiste em acompanhar as informações de início e término da execução dos elementos do fluxo. Esse nível de monitoração permite o acompanhamento completo de todas as ações realizadas pelo executor e também informa, quando existirem, resultados de chamadas. Por último, o monitoramento textual de mensagens de acompanhamento consiste em receber mensagens de texto descritas nos elementos do fluxograma. Mensagens desse tipo podem ser definidas antes ou após a execução dos elementos do fluxo, permitindo, inclusive, a obtenção de valores de variáveis do fluxograma.

A monitoração visual dos fluxogramas é disponibilizada no próprio ambiente de modelagem e, uma vez ativada, marca os elementos do fluxograma que foram executados e exibe uma tabela com o valor das variáveis do fluxograma.

As informações para a monitoração visual da execução dos fluxogramas são geradas pelo executor e enviadas pelo servidor em momentos em que não existe uma ação pronta para a execução. Esses momentos ocorrem quando apenas comandos de espera estão ativos no servidor. Por esse motivo, somente os elementos executados entre a última espera e a espera atual são marcados como executando. Dessa forma, o último passo de execução de cada fluxograma fica em destaque para o usuário. Para fluxogramas que definam execução em paralelo de elementos é possível acompanhar cada uma das sequências de elementos em paralelo, bem como o valor observado para as variáveis do fluxograma em cada uma.

### 5.1.4 Interação com Bases de Dados e Outras aplicações

O sistema MPA é desenvolvido usando a linguagem de programação Lua [19]. Lua é uma linguagem de *script* que combina uma sintaxe procedural simples com descrição de dados baseada em vetores. Por usar apenas o padrão ANSI C, Lua pode ser associado a aplicações escritas em C ou C++, além de permitir a integração com bibliotecas auxiliares por meio de sua API C. Isso permite que o servidor do MPA carregue bibliotecas externas para interagir com ferramentas auxiliares. Dessa forma, é possível definir funções auxiliares que podem ser usadas pelos elementos dos fluxogramas para realização de tarefas específicas como cálculos ou o acesso a sistemas legados.

A linguagem não possui suporte a execução simultânea de múltiplas *threads*, mas oferece suporte a co-rotinas [20]. Com co-rotinas, a troca de contexto de execução é feita explicitamente no código. Essa característica impede o aproveitamento de máquinas com múltiplos processadores, mas, em contrapartida, simplifica o desenvolvimento e compreensão do código desenvolvido pois não há necessidade de sincronização e de definição de áreas críticas.

A comunicação entre as aplicações do sistema utiliza o padrão CORBA [21]. Utilizando esse padrão, uma aplicação externa pode acessar, controlar e monitorar os fluxogramas do servidor MPA a partir da sua interface e localização.

Outro tipo de aplicação que os fluxogramas podem acessar são aplicações que utilizam o padrão COM [22]. Isso porque o servidor MPA vem embutido com o LuaCOM [23], uma biblioteca desenvolvida para comunicação com aplicações que utilizem o padrão COM. Exemplos de aplicações desse tipo são aplicações que utilizem OLE/ActiveX ou ainda sistema de automação que utilizem o padrão OPC (*OLE for Process Control*) [24].

### 5.1.5 Comparação com Sistemas de Gerenciamento de Workflow

Observando as características do sistema MPA, percebe-se que ele pode ser caracterizado como um sistema de gerenciamento de *workflow*. O sistema possui um ambiente para modelagem dos processos, uma motor para execução dos processos, mecanismos de monitoração e gerenciamento da execução dos processos, além da capacidade de interagir com bases de dados e aplicações externas.

Os modelos de processos que podem ser descritos no MPA são diferentes dos modelos padrão de sistemas de gerenciamento de *workflow*. Isso porque fluxogramas do MPA não apresentam o padrão de condição para ativação de forma explícita e pelo fato de não haver um fluxo de dados entre os elementos do fluxograma. No lugar do fluxo de dados o MPA possui o conceito de um ambiente de variáveis do fluxograma. Essas variáveis são compartilhadas entre os elementos do fluxograma e podem ser de dois tipos: variáveis locais e variáveis globais. Variáveis locais são compartilhadas entre elementos de uma sequência direta de execução, sem execuções em paralelo. Variáveis globais são compartilhadas por todos os elementos de um fluxograma, no caso, esse conjunto de variáveis é funcionalmente compatível com o conceito de fluxo de dados, pois todas as ações poderão obter e alterar o valor dessas variáveis. Os elementos do fluxograma são avaliados pelo motor do sistema apenas quando o fluxo de execução os alcança, ou seja, não são necessários ciclos de atualização [2], em que o motor itera sobre todas as ações do modelo do processo para determinar quais devem ser ativadas. Sendo assim, como não há o conceito explícito de ativação, o padrão de condição usado para inibir transições entre ações não está presente. No lugar disso, ações são ligadas diretamente e, quando há a necessidade de aguardar alguma condição, são usados elementos de escolha e esperas explícitas.

Uma diferença conceitual, é o foco da utilização do MPA para o controle de processos produtivos. Por esse motivo, não existe no MPA o conceito explícito de recursos, documentos necessários ou de lista de tarefas para um empregado. Entretanto, é possível criar ações que utilizem esses conceitos a partir de uma aplicação externa ou do gerenciamento de uma interface externa a partir de métodos e funções. Além disso, o volume de dados é pequeno, cada fluxograma acessa apenas os atributos necessários para sua lógica e não demandam, por exemplo, informações completas ou listas de documentos para sua execução.

Em relação ao tempo de resposta, o MPA funciona em intervalos de tempo na ordem de segundos. Isso acontece pois as informações que ele utiliza

são obtidas a partir de bases de dados atualizadas em ciclos de varredura. Como a varredura de toda a base de dados é demorada, esse intervalo de tempo não é crítico. Por outro lado, sistemas de gerenciamento de *workflow* tem tempo de resposta mais rápido pois se baseiam em ciclos de atualização. Entretanto, as esperas no MPA são definidas explicitamente nos fluxogramas e liberam o executor para realização de outras ações enquanto o fluxograma aguarda a atualização dos valores relevantes. Isso ocorre porque o motor do MPA herda o conceito de multi-tarefa cooperativo [20] da linguagem Lua. Então, caso seja necessário, é possível definir tempos de espera menores para garantir a execução imediata de ações de processos, levando em conta que o motor ficará ocupado a maior parte do tempo, podendo interferir em funções com menor prioridade, como a monitoração da execução.

## 5.2 Modificações Necessárias

Como discutido no capítulo anterior, a utilização do mecanismo de tolerância a falhas implica em modificações no sistema de gerenciamento de *workflow*. Essas modificações devem ser feitas no motor do sistema, nesse estudo de caso, o servidor MPA. No caso, as modificações feitas foram: permitir obter e aplicar um estado do executor, inibir e iniciar a execução do estado do executor, associar um identificador único de estado, encaminhar a conexão de um cliente para um outro servidor do MPA, criar e aplicar apenas atualizações no estado do executor, enviar mensagens de atualização sempre que uma ação for concluída.

O estado do executor do MPA é composto por informações dos fluxogramas e informações do gerenciamento da execução. As informações do fluxogramas consistem em variáveis globais, variáveis locais e os elementos do fluxograma. Já o gerenciamento da execução precisa conhecer qual elemento está sendo executado, quais co-rotinas estão associadas a quais fluxogramas, se existem chamadas recursivas (elementos iteradores, chamadas de fluxogramas função ou sincronização). Essas informações elevam o grau de complexidade do estado que precisa ser informado e ajustado. A implementação do executor do MPA não disponibilizava explicitamente todas essas informações, em especial a sequência de chamadas recursivas.

Para deixar o encadeamento de chamadas recursivas explícito, a implementação do executor do MPA foi alterada para representar a execução do fluxograma como um pilha em que os elementos do fluxograma chamados recursivamente eram empilhados. Nessa representação, uma nova pilha passa a ser associada a cada co-rotina criada para execução dos fluxograma, seja pelo

início de um novo fluxograma aplicação ou pelo início de uma nova execução em paralelo. Nessa pilha, o contexto da chamada de cada elemento fica armazenado em uma lista encadeada. Dessa forma, chamadas que realizam alguma troca de contexto, como chamadas de fluxogramas função ou iterações sobre instâncias de uma classe são empilhadas e, uma vez que forem concluídas, o contexto anterior é recuperado incluindo os resultados, se houverem, da execução do elemento do fluxograma. Essa modificação no executor permitiu a obtenção do estado completo da execução dos fluxogramas, com toda a sequência de chamadas e contextos associados.

Aproveitando essa modificação, foi possível implementar o conceito de mensagens de atualização para ajuste do estado do executor sem a necessidade de enviar o estado completo. Isso porque apenas a pilha da co-rotina que executa um elemento do fluxograma é alterada após a conclusão da execução desse elemento. Sendo assim, para enviar todas as alteração feitas por esse elemento, basta enviar o conteúdo atual da pilha. Nesse caso, é necessário associar as pilhas às co-rotinas que já foram criadas em estados anteriores. Para tal, todas as co-rotinas criadas para a execução em paralelo de ações ou para novos fluxos são associadas a um identificador único.

Além dos casos em que ações atualizam o estado, outros eventos que alteram o estado do executor são o início e a conclusão de um fluxo aplicação. Ao iniciar um fluxo aplicação é criada uma mensagem com o identificador da co-rotina e a sua pilha. Quando o executor recebe uma mensagem de atualização com uma pilha e um identificador de co-rotina desconhecido, uma nova co-rotina é criada e associada a essa nova pilha. Para identificar a conclusão de uma co-rotina, é usada uma mensagem de atualização com o identificador da co-rotina concluída sem uma nova pilha. Ao receber essa mensagem com essa construção, a co-rotina identificada é removida.

Todas as atualizações implicam no incremento do identificador do estado. Esse identificador é mantido pelo próprio executor e enviado com o estado completo ou com uma mensagem de atualização.

A modelagem dos processos é feita em um ambiente de desenvolvimento e não é necessário que haja tolerância a falhas nessa fase da utilização do sistema. Entretanto, o sistema precisa ser tolerante a falhas quando a sua execução real é iniciada. Por esse motivo, apenas os modelos de processo e estado de execução fazem parte do estado relevante a ser replicado, sendo o estado inicial carregado na inicialização das réplicas do servidor.

Caso o executor tenha apenas recebido o estado completo e atualizações, nenhuma de suas co-rotinas estará registrada para a execução. Sendo assim, o executor estará em espera e, para que fique ativo, foi definido um comando que

identifica e registra as co-rotinas que estavam em execução no estado recebido.

Para garantir que a monitoração da execução dos fluxogramas vai sempre representar o estado atual e que será possível controlar a execução dos processos, foi utilizada a localização de serviço. Ao receber um pedido de conexão de um cliente, o servidor MPA replicado pede ao seu replicador a referência para o servidor MPA principal. Essa referência é retornada para o cliente MPA que, consequentemente, não precisa saber que o servidor MPA está sendo replicado ou qual é o servidor que está executando de fato. Dessa forma, os comandos que o cliente enviar para o servidor serão replicados indiretamente, pois irão gerar alterações no estado de execução que, por sua vez, irão ser replicadas para os demais servidores.

Quanto a confirmação da suspeita de falha de uma servidor MPA principal, é possível utilizar a lógica de informação de atividade que o MPA gera para informar que está operando para usuários do sistema. Essa lógica de detecção de atividade é chamada de lógica de *watchdog* e consiste em executar, na máquina que disponibiliza a base de dados, um procedimento que busca indicativos de atividade do MPA e revoga a autorização para execução dos fluxogramas caso não haja atividade por um intervalo de tempo determinado. Essa lógica pode ser adaptada para auxiliar a confirmação de falha de réplica primária. Nesse caso o campo reservado para o *watchdog* poderia ser associado a um temporizador que seria zerado periodicamente em um fluxograma. Sendo assim, o servidor principal ficaria encarregado de reiniciar a contagem e, caso uma réplica consulte esse campo e verifique que o temporizador passou do intervalo de tempo para suspeita, os servidores secundários confirmariam a falha do servidor principal.



## 5.3 Testes de Desempenho

Como discutimos no capítulo 3, a replicação de um sistema se baseia na repetição de chamadas e na instanciação de cópias de um processo ou objeto, podendo gerar sobrecarga na execução desse sistema. Sendo assim, é necessário avaliar o impacto dessa sobrecarga no desempenho do sistema replicado e determinar se esse impacto não interfere de forma significativa na execução dos processos. Para isso, foram elaborados testes de desempenho que comparam a execução do sistema sem replicação com a execução replicada do sistema, na presença de falhas ou não. Além desses casos, foram realizados testes para verificar o impacto da entrada de uma nova réplica secundária, visando determinar o custo da inclusão de uma réplica nova.

### 5.3.1
### Modelo de Processos Usado para Testes

O modelo usado no teste de desempenho representa um uso possível do sistema. Esse modelo apresenta um conjunto de fluxogramas que exercita as construções possíveis do MPA e características identificadas para sistemas de gerenciamento de *workflow*. O conjunto de fluxogramas contém ações não-determinísticas, execuções em paralelo, sincronização, término implícito, chamada de fluxogramas função, cancelamento de fluxograma aplicação, iteração sobre instâncias, acesso a bases de dados, interação com um usuário e lógica para ativação de *watchdog*.

Para testar o comportamento na presença de ações não-determinísticas foram descritas ações para determinar o valor de abertura de válvulas, o nível de um tanque e a sua temperatura. Além das leituras, foi definido um procedimento que ajusta a abertura de uma válvula em passos, a cada intervalo de tempo. Foram definidos dois fluxogramas aplicação: um para o gerenciamento de válvulas de tanque e outro para gerenciamento de *watchdog* e chave de habilitação.

O fluxograma para gerenciamento de *watchdog* e chave de habilitação é responsável por verificar se a chave que autoriza a atuação do MPA está habilitada, iniciando a execução do fluxograma de gerenciamento de tanques, ou desabilitada, cancelando a execução do fluxograma de gerenciamento de tanques. O gerenciamento de *watchdog* consiste em acionar o *watchdog* a cada 10 segundos. Essa chamada foi modelada de forma a escrever o valor zero em um valor da base de dados que corresponde a um temporizador.

O fluxograma de gerenciamento de válvulas de tanque executa o mesmo procedimento para todos os tanques configurados que, nesse teste, são cinco. Nesse procedimento, o valor de abertura da válvula de entrada, o nível e temperatura de cada tanque é lido para determinar a abertura da válvula de saída e a ativação do sistema de controle de temperatura. Caso a válvula de entrada esteja fechada, o procedimento ignora o controle para aquele tanque. Para os demais tanques, o fluxograma irá iniciar o controle do nível e da temperatura do tanque de acordo com limites estabelecidos para essas variáveis. Cada um desses controles é realizado até que a variável correspondente atinja os valores de operação entre os limites estabelecidos. Uma vez que as variáveis de todos os tanques tenham atingido os valores de operação o fluxograma é considerado concluído, e o tempo desde o início da execução do fluxograma é armazenado. O fluxograma está representado na figura 5.2. O controle do nível está localizado na parte superior do fluxograma e o controle de temperatura está localizado na parte inferior.

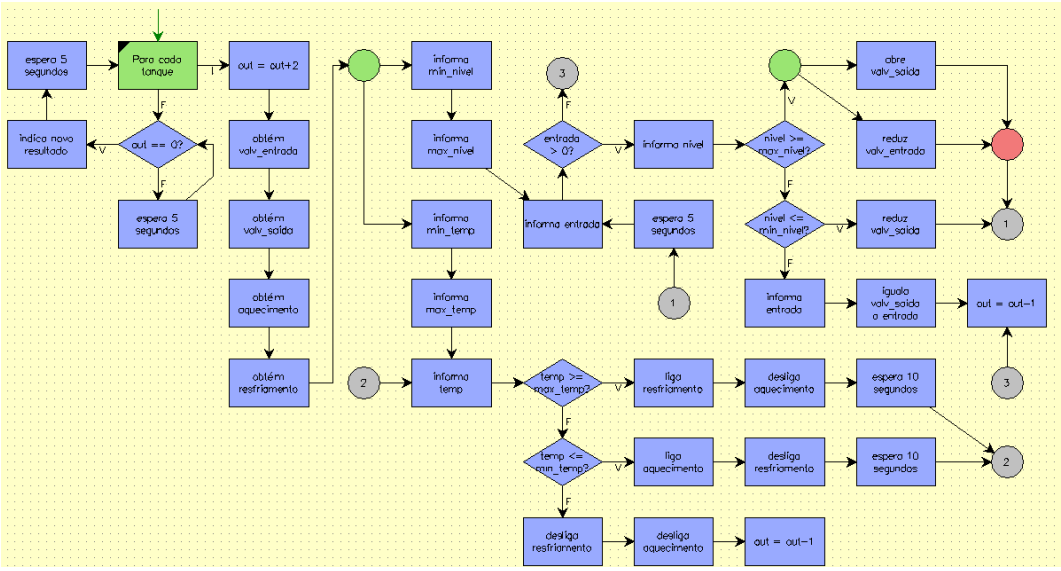


Figura 5.2: Fluxo de controle de temperatura e nível de tanques.

O controle da temperatura atua em três casos: se a temperatura estiver acima do limite máximo, o refrigerador é ligado; se a temperatura estiver abaixo do limite mínimo, o sistema de aquecimento é ligado; se a temperatura estiver na faixa de operação tanto o aquecimento quanto o resfriamento são desligados e o controle é concluído.

O controle do nível irá observar o nível do tanque e a abertura da válvula de entrada, atuando de acordo. Caso o nível do tanque esteja abaixo do nível mínimo, a válvula de saída será fechada até um quarto. Se o nível do tanque estiver na faixa de operação a válvula de saída será mantida no mesmo nível da válvula de entrada. Se o nível do tanque estiver acima do máximo, o fluxograma irá, paralelamente, abrir a válvula de saída o máximo e fechar a válvula de entrada até a metade, sincronizando a execução ao final. A abertura da válvula é feita separadamente no fluxograma função representado na figura 5.3. Esse fluxograma irá receber por parâmetro a abertura final, intervalo entre aberturas e variação de abertura para a válvula. Sendo assim, o fluxograma altera a abertura da válvula até que fique a uma diferença menor que variação da abertura final. Esse fluxograma é concluído uma vez que aplique o valor de abertura desejado.

Apesar de conter chamadas não-determinísticas, para que possamos comparar o resultado entre duas execuções do controle o estado inicial de cada um dos tanques será sempre o mesmo para todos os testes. Sendo assim, cada tanque irá realizar sempre a mesma sequência de ações. Configuramos um fluxograma adicional para executar separadamente na máquina em que

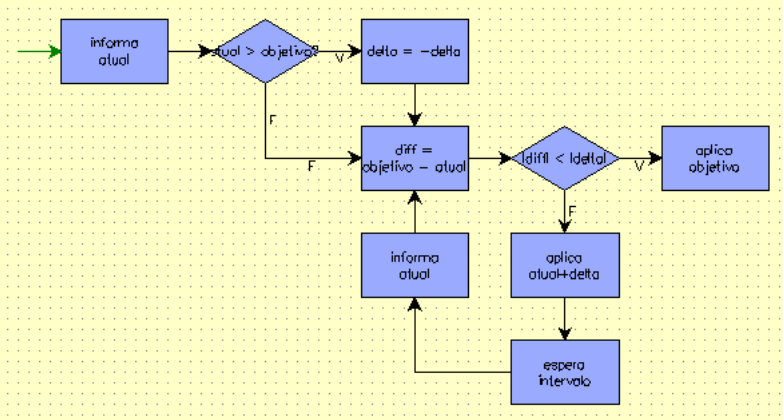


Figura 5.3: Fluxo movimentação de válvulas.

está localizada a base de dados para simular a resposta da planta, alterando os valores das variáveis do tanque de acordo com o estado dos seus equipamentos. Isso foi feito de forma que o nível do tanque varie de acordo com a abertura das suas válvulas e o nível atual, enquanto a sua temperatura varie de acordo com a operação dos sistemas de aquecimento e o valor da temperatura atual. Serão modelados, ao todo, cinco tanques com os mesmos limites para temperatura e nível, mas com diferentes valores iniciais entre si. O controle é considerado como concluído uma vez que tanto o nível quanto a temperatura tenham atingido o valor de operação entre os limites. Entretanto, como estamos verificando em intervalos de tempo se o controle foi concluído, é possível que haja uma variação de até um intervalo entre duas medidas.

Para testar o impacto no desempenho de uma execução sem falhas foram analisados os seguintes casos:

- execução do conjunto de fluxogramas sem modificações no servidor de execução
- execução do conjunto de fluxogramas sem nenhum replicador
- execução do conjunto de fluxogramas apenas com o replicador principal
- execução do conjunto de fluxogramas com 1 réplica secundária
- execução do conjunto de fluxogramas com 2 réplicas secundárias
- execução do conjunto de fluxogramas com 3 réplicas secundárias

Para testar o impacto da entrada de uma réplica secundária foram analisados os seguintes casos:

- execução com entrada da primeira réplica secundária
- execução com entrada da segunda réplica secundária

– execução com entrada da terceira réplica secundária

Para testar o impacto de falhas e recuperação de réplicas primárias foram analisados os seguintes casos:

– execução com falha de réplica primária e 1 réplica secundária
– execução com falha de réplica primária e 2 réplicas secundárias
– execução com falha de réplica primária e 3 réplicas secundárias

## 5.4 Resultados

Realizamos os testes descritos anteriormente instalando o sistema MPA em quatro máquinas Windows de uma rede local. Em cada caso de teste partimos os servidores com o mesmo estado inicial. Para cada um desses servidores inicializamos um replicador, um como líder e os demais com a localização deste. Uma vez que os replicadores já estavam em operação, iniciamos a execução do fluxo ligando a chave de habilitação.

O fluxograma de gerenciamento de tanque que propusemos realizava o controle aproximadamente em um minuto. Nos casos em que era necessário, inserimos falhas enquanto o controle estava sendo realizado e observávamos quanto tempo a mais era necessário para que o controle fosse concluído. Cada caso de teste foi realizado 30 vezes e a seguir iremos incluir a média e desvio padrão para cada caso de teste realizado.

Para esses testes, estabelecemos que o intervalo para identificação de falha deveria ser de 30 segundos. Esse intervalo alto foi definido pois observamos que a infra-estrutura de comunicação deixava de enviar mensagens de *keepalive* enquanto tentava recuperar o canal de comunicação com a réplica em falha. Sendo assim, nesse intervalo de tempo uma réplica secundária poderia ser incorretamente suspeita de falha.

O primeiro grupo de testes foi feito visando determinar o tempo em que o fluxograma levaria para retornar a temperatura e nível de cada tanque para os valores de operação, entre a faixa de máximo e mínimo. Nesses testes não inserimos falhas nas réplicas, observando apenas o quanto a replicação interferiria na execução do fluxograma. Como referência incluímos um caso com uma versão do servidor de execução sem as modificações necessárias para a utilização do mecanismo.

Observando os resultados da tabela 5.4 percebemos que o tempo médio para alcançar a faixa de operação não varia de forma significativa à medida que número de réplicas utilizadas aumenta. Em especial, o resultado foi próximo aos demais no caso em que foi utilizada uma versão do servidor de execução

| Replicadores | Média | Desvio padrão |
|---|---|---|
| - | 55.5 | 2.3 |
| 0 | 56.5 | 3.5 |
| 1 | 55.7 | 2.7 |
| 2 | 55.7 | 2.7 |
| 3 | 56.4 | 2.7 |
| 4 | 56.1 | 2.3 |

Tabela 5.2: Resultados de execução sem falhas

sem as modificações necessárias para utilização do mecanismo. Além disso, fica claro que o desvio padrão é pequeno em relação a média observada. Entretanto, acreditamos que o valor do desvio padrão está associado a forma que o servidor MPA realiza o controle proposto. Configuramos os fluxogramas para aguardar cinco segundos entre duas verificações da faixa de operação dos tanques. Por esse motivo, o tempo para conclusão do controle pode ser identificado entre dois intervalos de espera para verificação, em alguns casos sendo identificado antes e outro depois da espera. Sendo assim, médias da ordem de um ou dois segundos de diferença podem ser consideradas como próximas e o desvio padrão abaixo desses cinco segundos fica dentro do esperado.

| Secundária | Média | Desvio padrão |
|---|---|---|
| 1a | 51.6 | 4.3 |
| 2a | 55.7 | 3.5 |
| 3a | 59.3 | 3.4 |

Tabela 5.3: Resultados de execução com entrada de secundária

Diferente dos resultado observados para a execução sem falhas, percebe-se que a entrada de uma réplica nova interferiu no tempo para conclusão do controle. Observando os resultados descritos na tabela 5.4 podemos concluir que um número maior de réplicas implica no aumento do tempo gasto para conclusão do controle. Usando a entrada da primeira réplica secundária como referência, percebemos que a entrada da segunda réplica adia a conclusão do controle por aproximadamente um ciclo de verificação (quatro segundos). Além disso, o aumento no tempo gasto para a entrada da terceira réplica é de aproximadamente dois ciclos (oito segundos). Consideramos que esse tempo a mais está relacionado ao tempo que o servidor principal gasta para registro de uma nova réplica, obtenção de estado por parte da réplica principal, envio e atualização do novo estado para as réplicas secundárias.

O maior impacto no tempo para conclusão do controle foi observado na falha da réplica principal. Esse tempo está diretamente relacionado com o intervalo para identificação da sua falha. Percebemos que os resultados apresentados na tabela 5.4 indicam um aumento no tempo médio para conclusão

| Réplicas | Média | Desvio padrão |
|---|---|---|
| 2 | 78.7 | 7.2 |
| 3 | 87.9 | 10.7 |
| 4 | 87.5 | 14.3 |

Tabela 5.4: Resultados de execução com falha na réplica principal

do controle em torno de 30 segundos, o valor estabelecido pelo *timeout*. Entretanto, a variação do número de réplicas parece não interferir com o tempo médio para concluir o controle em casos de falha da réplica principal.

Nesse caso de teste, observamos que o desvio padrão para as medições é alto em relação ao intervalo entre verificações. Acreditamos que esse aumento no desvio padrão esteja relacionado a simulação desenvolvida para o teste de desempenho. A simulação varia o nível e a temperatura de acordo com a abertura das válvulas do tanque e o estado de operação do aquecedor e do refrigerador. Sendo assim, durante intervalo entre a falha do servidor principal e a partida de um novo servidor principal, as variáveis do tanque podem ser levadas a um estado não observado em condições sem falha, por sua vez, levando a mais passos para retorno a faixa de operação. Além disso, em casos de falha na réplica principal, os intervalos de espera podem ser reiniciados. Isso acontece devido ao fato do MPA aguardar a conclusão dos intervalos de espera para enviar um novo estado. Por exemplo, supondo que o servidor principal falhe quando faltar um segundo para um espera de cinco segundos ser concluída, a réplica que assumir o papel de principal irá iniciar novamente a espera quando retomar a execução, levando outros cinco segundos para executar a próxima ação. Ou seja, além do tempo para identificação da falha e eleição, a espera pode levar até o dobro do tempo definido.

Um outro fator que também explica o desvio padrão maior que nos casos anteriores é a questão da coleta de lixo realizada nos servidores secundários. A todo momento são enviadas estruturas de dados para atualização e, eventualmente, o estado completo. Todas essas estruturas, uma vez utilizadas para atualizar o estado, são descartadas e passíveis de remoção no próximo ciclo do coletor de lixo. Sendo assim, como os testes foram realizados em sequência, algumas medições provavelmente foram influenciadas por coletas realizadas durante a execução do fluxograma de controle.