

5

Efeitos do buffer de capacidade e da priorização de atividades

Neste capítulo se analisam os efeitos que a implementação dos *buffers* de capacidade – BC têm sobre o sistema. Para tal fim, construiu-se um modelo de simulação que representa a execução de múltiplos projetos, permitindo separar os atrasos causados por conflitos de recursos daqueles causados pela aleatoriedade das durações das atividades.

Neste capítulo também são analisados os efeitos que o tipo de priorização de atividades utilizado durante a etapa de execução tem sobre o sistema. Com essa finalidade foi implementado, no modelo anterior, um processo decisório que permite executar programas baseados em prioridades fixas e dinâmicas.

A apresentação dos resultados é iniciada com a definição dos objetivos, parâmetros, cenários e lógica de simulação a cada experimento. Finalmente se apresentam os gráficos e tabelas relevantes para o entendimento dos resultados. Outras informações complementares são apresentadas no apêndice B.

5.1.

Efeitos dos *buffers* de capacidade - BC

5.1.1.

Objetivos

O modelo de simulação tem como objetivo representar a etapa de execução dos múltiplos projetos do máster, permitindo a identificação e entendimento dos efeitos que os BC têm sobre os objetivos do sistema e dos projetos individuais. Com o modelo, pretende-se esclarecer se efetivamente há necessidade de dimensionar os BC de acordo com cada projeto e não como uma porcentagem fixa (da capacidade do recurso gargalo) para todos os projetos.

Do mesmo modo, busca-se explicar o desempenho do BC como complemento de segurança do BP e analisar o compromisso (*trade-off*) entre

atingir um sistema de maior estabilidade (menos atrasos por falta de recursos) e estabelecer um sistema de menor *makespan*.

5.1.2. Estrutura da Simulação

O modelo de simulação é baseado na programação híbrida (definida no Capítulo IV), em que os BC são inseridos somente no “programa projetado”. A simulação representa a interação entre o “programa base”, definido pelo recurso gargalo, e a execução probabilística das atividades do “programa projetado” com os BC inseridos.

A simulação é realizada em processos que permitem corridas de até 40 horas de duração, o que é equivalente a programas que se estendam até 40 unidades de tempo (t). Para cada cenário são efetuadas 80 réplicas com valores aleatórios distintos entre si. Os valores aleatórios das 80 réplicas são similarmente reproduzidos em cada cenário, permitindo, que sua análise seja sob as mesmas condições. A decisão de 80 replicações foi arbitrária e poderá ser estendida para menor variância nos resultados.

- **Programa de base.** - O modelo é baseado na programação base, obtida na abordagem do recurso estratégico ou gargalo, no exemplo o recurso “A” é definido como gargalo do sistema.

Com a finalidade de isolar o comportamento dos BC em relação aos BP, são observadas algumas particularidades no programa *master* da Figura 5.1

- São representadas apenas as atividades críticas e os respectivos BP para cada projeto. Isso porque, para a finalidade da análise, as atividades não críticas e seus respectivos BA representam um aumento desnecessário da complexidade do modelo.
- Para manter um significativo compartilhamento de recursos entre os projetos do sistema, as disponibilidades de recursos do exemplo são redefinidas: “A” = 6, “B” = 4, “C” = 3.
- Mesmo que o programa de base seja nivelado em relação ao recurso “A”, o modelo exige que a execução das atividades satisfaça o nivelamento de todos os recursos. Dessa maneira se consegue estudar os BC num

ambiente de exigência máxima. Esse procedimento de execução, ao desconsiderar as atividades não críticas, não entra em conflito com o programa básico original.

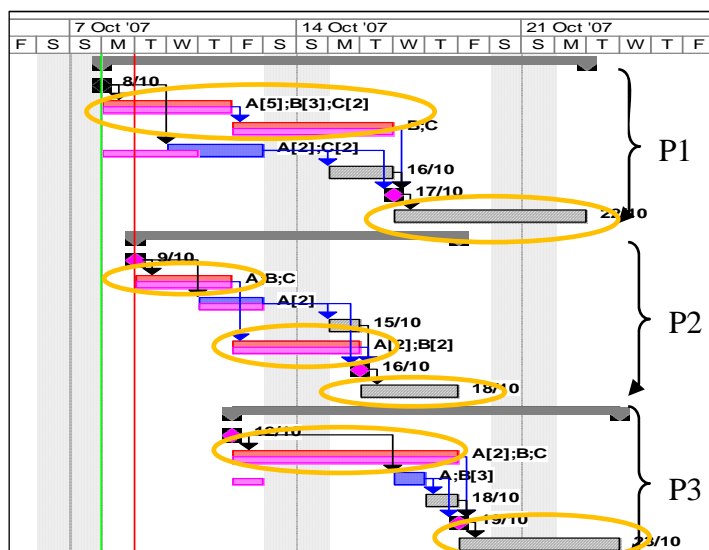


Figura 5.1: Programa base do modelo de simulação

A definição do programa de base é feita através de variáveis que, de acordo com a seqüência estabelecida na Figura 5.1, determinam o início e fim de cada atividade. Dessa maneira o programa de base se mantém fixo durante o processo de simulação e é usado como referência na determinação do estado de execução das atividades. Como exemplo se mostra na Tabela 5.1 as variáveis definidas para as atividades do projeto um.

Tabela 5.1: Descrição das variáveis que definem o programa de linha de base

Variáveis do programa de linha de base do projeto um
$P1A1_TIB = 0$
O instante zero é atribuído ao tempo de início básico da primeira atividade.
$P1A1_TFB = P1A1_TIB + 4$
O tempo final básico da primeira atividade está definido pelo tempo de início mais a duração fixa da atividade.
$P1A2_TIB = P1A1_TFB$
Ao tempo de início básico da segunda atividade é atribuído o mesmo instante em que a primeira atividade é terminada, existe uma restrição de precedência.
$P1A2_TFB = P1A2_TIB + 3$
O tempo de término básico da segunda atividade é definido pelo tempo de início mais a duração fixa da atividade.

- **Programa Projetado.** - O programa projetado é determinado por meio de variáveis, que mudam dinamicamente com o avanço da simulação. A inserção do espaço definido pelos BC é realizada no momento um da execução ($t = 1$), sendo que a simulação é iniciada no momento zero.

A Tabela 5.2 mostra a definição das variáveis correspondentes ao projeto dois. O código completo pode ser encontrado no apêndice B.

Tabela 5.2: Descrição das variáveis que definem o programa projetado

Variáveis do programa projetado do projeto dois
$P2A1_TIP = P2A1_TIB + P2A1_varTIP + P2_BC$
O tempo de início projetado da primeira atividade é definido pelo tempo de início básico mais o incremento de uma unidade de tempo, a qual é somada quando a atividade é efetivamente executada. A partir do momento um da simulação se adiciona ao cálculo o valor fixo do <i>buffer</i> de capacidade correspondente.
$P2A1_TFP = P2A1_TIP + 3 - P2A1_varTFP$
O tempo final projetado da primeira atividade é definido pelo tempo de início projetado mais a duração fixa da atividade. Quando a atividade é efetivamente executada se diminui uma unidade de tempo ao valor calculado. Isso permite que o tempo final projetado se mantenha fixo e que o cálculo da quantidade de trabalho não completado seja diminuído.
$P2A3_TIP = P2A1_TFP + P2A3_varTIP$
O tempo inicial projetado da terceira atividade é definido pelo tempo final projetado da atividade anterior, (mantendo a restrição de precedência definida no programa base). O incremento da unidade de tempo tem a mesma lógica que a atividade P2A1_TIP. O BC é aumentado unicamente na primeira atividade e transmitido na cadeia através das dependências de seqüenciamento.
$P2A3_TFP = P2A3_TIP + 2 - P2A3_varTFP$
O tempo de término da terceira atividade tem a mesma lógica da atividade P2A1_TFP.

- **Distribuição de Probabilidade.** – O modelo utiliza a função de distribuição binomial negativa para descrever o comportamento da duração das atividades na etapa de execução.

A binomial negativa é uma distribuição de probabilidade discreta que pode ser usada para descrever experimentos que consistem numa seqüência de tentativas independentes. Cada tentativa pode resultar em sucesso ou falha, sendo a probabilidade de sucesso, p , constante para todas as tentativas do experimento.

A binomial negativa pode ser definida tanto em função do número de tentativas como em função do número de falhas (num experimento de Bernoulli) requeridas até se obter um dado número de sucessos. (CASELLA & BERGER, 2002)

A lógica do modelo de simulação usa a definição da distribuição baseada no número de falhas. Cada iteração simulada corresponde a uma tentativa de execução da atividade num dado intervalo de tempo, podendo esta resultar em uma correta execução do trabalho planejado (sucesso) ou na perda desse tempo (fracasso). Não são considerados atrasos de frações de tempo, permitindo somente durações de valores múltiplos (t) da unidade de tempo..

O processo de simulação é repetido em cada atividade até que o número de sucessos iguale à duração planejada da atividade. Dessa maneira, o numero de tentativas (soma do número de falhas e do número de sucessos) determina a duração total da atividade.

A definição formal da função de distribuição binomial negativa em função da variável aleatória Y = número de falhas antes do $r^{\text{ésimo}}$ sucesso, é dada por:

$$P(Y = y) = \binom{r + y - 1}{y} p^r (1 - p)^y, \quad y = 0, 1, \dots$$

Parâmetros	$p = \text{probabilidade de sucesso}$ $r = \text{número de sucessos alvo}$
Média	$= r \frac{(1 - p)}{p}$
Variância	$= r \frac{(1 - p)}{p^2}$

Onde: $Y = X - r$; X = número de tentativas até o $r^{\text{ésimo}}$ sucesso.

- A forma da distribuição binomial negativa para o problema modelado (Figura 5.2) é similar à distribuição que descreve as atividades de um projeto, definida por Leach (2005). Nesse tipo de distribuições as durações extremamente curtas como as durações extremamente longas têm uma chance

aceitável de acontecer. Para valores grandes de r a distribuição tende a ser mais simétrica.

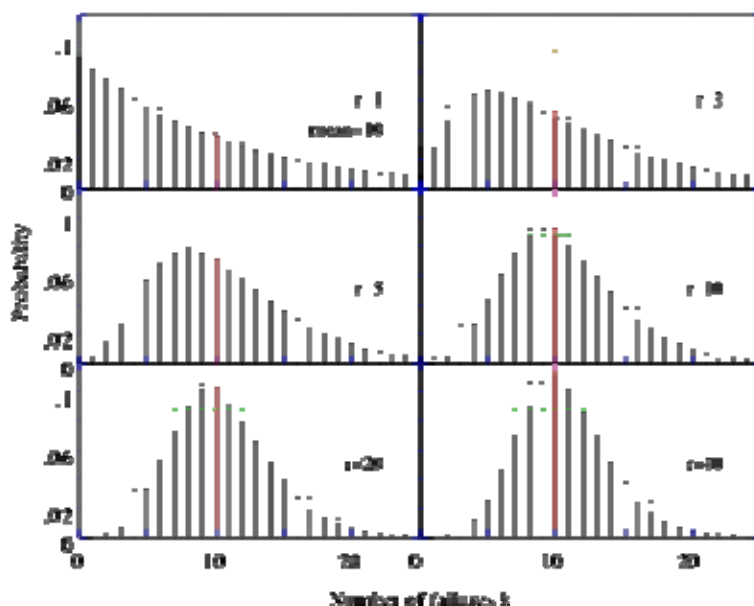


Figura 5.2: Distribuição Binomial negativa
Fonte: <http://en.wikipedia.org>

Para implementar a distribuição binomial negativa no modelo, faz-se uso da propriedade das distribuições binomiais de que a soma de duas ou mais variáveis binomiais independentes com o mesmo parâmetro p é ainda uma variável binomial com parâmetros r igual ao somatório dos r_i e o mesmo p .

Desse modo se considera que as probabilidades de sucesso das diversas atividades de um mesmo projeto são independentes e de mesmo valor p . Conseqüentemente, o valor de r é dado pela duração enxuta (sem considerar o buffer) do projeto.

Com esses pressupostos, calcula-se o valor de p em função da probabilidade média de atraso (ou falhas) aceitável na estimação da duração do projeto. Ou seja, com base na probabilidade de ter tantos atrasos na execução das atividades do projeto quanto o buffer do projeto possa suportar. Esse cálculo é feito na planilha eletrônica (*Excel*) e utilizado numa ferramenta de simulação (*ProccessModel*) que onde se insere o caráter probabilístico do modelo. Diferentes valores de probabilidade de atrasos foram definidos para a análise dos modelos.

- **Os parâmetros e cenários de simulação.** – Os diferentes cenários de simulação são definidos por dois parâmetros principais:

1. A porcentagem utilizada para dimensionar o BC: O cálculo dessa porcentagem considera que o tempo reservado pelo *buffer* é complemento do tempo que o recurso investe na atividade anterior da sequência crítica (porcentagem que completa o 100% da disponibilidade do recurso). Lembrar que os BC são inseridos entre duas atividades que iniciem consecutivamente dentro da sequência crítica e que pertençam a projetos diferentes. Portanto, no modelo são inseridos dois BC calculados segundo o exemplo:

$P2_BC = ((P1A1_TFB - P1A1_TIB) * BC2_tam) / (1 - BC2_tam)$
<p>O dimensionamento do BC que protege ao projeto dois é definido pela multiplicação da duração da primeira atividade do projeto um e a porcentagem do dimensionamento do BC, tudo dividido pelo complemento de tal porcentagem.</p>

2. O valor da probabilidade de sucesso (execução conforme programa) e falha (sem progresso) que define o comportamento aleatório da duração real das atividades: Devido que a ferramenta de software não suporta a implementação direta da distribuição binomial negativa, ela foi implementada dentro da lógica de programação. A ferramenta de simulação permite definir duas probabilidades independentes (de sucesso e falha) para determinar o sorteio aleatório da execução de atividades. Por meio da lógica de programação se dispõe que o sorteio seja efetuado até que o número de sucessos aleatórios seja igual à duração planejada das atividades, tal e como é o processo gerador da distribuição binomial negativa.

A combinação desses dois parâmetros define o conjunto de cenários avaliados. Para diferentes probabilidades de atraso do projeto (25%, 20%, 15% e 10%) foram experimentados diferentes tamanhos de BC (0%, 15%, 25%, 35% e 45%), definindo 20 cenários de avaliação.

- **A lógica do processo de execução.** – O modelo define o “Tempo” como a principal entidade que aciona o processo de execução de atividades, este se desloca através das atividades, ativando a execução daquelas que segundo o

programa de linha de base e o programa projetado estão prontas para ser trabalhadas. Para simplificar a representação dos projetos, define-se que uma hora é equivalente a uma jornada laboral (comumente dias de 8 horas).

A Figura 5.3 mostra a representação gráfica do modelo. As esferas correspondem às unidades de tempo que se deslocam através do processo de execução de atividades. A existência de três entidades “Tempo” se movimentando em forma paralela indica a simultaneidade da execução dos projetos. Três tipos de saídas indicam a mudança de ciclo de processamento, ou seja, o término de um dia de trabalho.

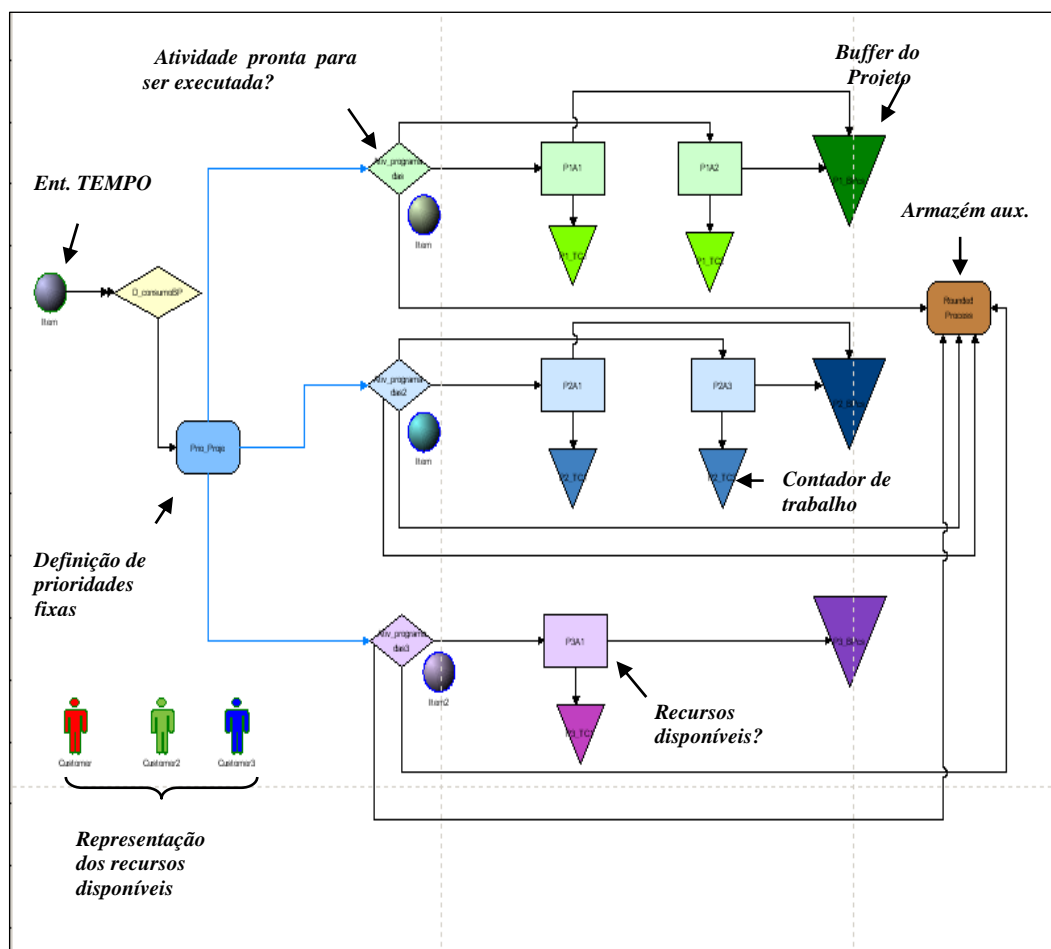


Figura 5.3: Modelo de simulação dos efeitos do BC

A entidade “Tempo” pode sair do sistema através de um “armazém” ou “contador do trabalho efetivamente realizado” (quando a execução da atividade não é retardada), através do buffer do projeto (quando a execução da atividade é retardada) ou por meio de um “armazém auxiliar” (quando não

existe nenhuma atividade pronta para ser executada), esta última opção não implica mudanças nos tempos projetados.

O processo de execução é iniciado com a definição das prioridades de cada projeto, neste caso as prioridades são determinadas segundo a ordem de chegada (1,2,3) e mantidas fixas durante todo o processo.

A partir das prioridades estabelecidas, se define uma lógica interna que permite liberar primeiro as entidades de tempo dos projetos prioritários. O “*delay*” aplicado às outras atividades, não afeta a cronometragem do modelo, mas, sim, permite que os recursos sejam alocados às atividades segundo as prioridades definidas.

A primeira decisão de execução é tomada nos losangos do diagrama, cada projeto tem entre três e quatro caminhos opcionais pra entidade tempo. Para poder entrar na primeira atividade (de qualquer projeto) a condição: Tempo de início base da atividade (TIB) \leq Momento de execução atual, ou, Tempo de início projetado (TIP) = Momento de execução, tem que ser cumprida. Para entrar numa atividade, é preciso que as atividades anteriores tenham sido completadas. As outras duas opções chegam ao armazém auxiliar, trasladando a entidade quando o projeto tiver terminado e quando o projeto é deslocado no tempo por causa dos BC (condição: TIB \leq Momento de execução e TIP $>$ momento de execução).

Uma vez que a entidade “Tempo” aciona a atividade, significa que esta está pronta para ser executada, não obstante, é preciso se avaliar a disponibilidade dos recursos. Para que a atividade possa ser executada, a necessidade de todos seus recursos tem que ser satisfeita, caso não seja assim, tal atividade será atrasada e o BP será consumido numa unidade de tempo.

Depois de alocar os recursos necessários para trabalhar a atividade e de atualizar o número disponível de cada recurso, procede-se com o sorteio aleatório que representa o resultado (sucesso ou falha) da execução. Caso esse resultado seja negativo (falha), a entidade “Tempo” será trasladada ao BP (consumindo-o). Caso o resultado seja positivo (sucesso), a entidade Tempo será trasladada ao armazém do trabalho realizado.

Este processo é efetuado de forma simultânea nos três projetos, assim também as atualizações de todos os tempos de início e fim projetados são feitas cada vez que a entidade “Tempo” encontre uma saída do processo.

Ademais do consumo do BP originado pelos atrasos das atividades, existe um consumo imediato originado pela inserção dos BC.

Os BP podem ser consumidos infinitamente, sendo que outro contador (*delay* ou atraso do projeto) é ativado no momento em que o consumo ultrapassa o limite do *buffer*.

Repare-se que o modelo permite identificar e contabilizar as diferentes naturezas do atraso do projeto, facilitando o isolamento do efeito dos BC.

5.2.

Efeitos da priorização de atividades.

5.2.1.

Objetivos

O modelo de simulação tem como objetivo representar o processo e as conseqüências que a aplicação de diferentes critérios de priorização de atividades durante a execução tem sobre o sistema. O propósito é esclarecer se o uso do critério de priorização dinâmica, com base no estado de consumo dos *buffers* do projeto – BP, obtém melhores resultados (em termos de tempo) do que a priorização fixa.

Neste contexto, o modelo busca, também, proporcionar a informação necessária para entender a essência das diferenças entre ambos os critérios e o porquê de seus efeitos no sistema.

5.2.2.

Estrutura da simulação

O modelo utilizado é uma adequação do modelo anterior. Adiciona-se uma decisão no início do processo para determinar o critério a utilizar. O modelo permite apenas a utilização de um único critério em cada simulação. Na priorização dinâmica se avalia o estado de consumo dos BPs, definindo as novas prioridades como em termos da ordem crescente dos tempos neles restantes. A

priorização mista é utilizada como referência de comportamento. Sua utilização é iniciada segundo as prioridades fixas dos projetos, esse critério passa a ser dinâmico só depois de que algum BP atinja a área crítica de consumo.

A simulação está baseada na execução do mesmo programa utilizado no modelo anterior. É assim, que tanto o programa base, o programa projetado e a distribuição de probabilidade são os mesmos definidos para o modelo anterior.

- **Os parâmetros e cenários de simulação.** - Para isolar o efeito das priorizações de algum outro efeito que pudesse distorcer os resultados, desconsiderou-se a inserção de *buffers* de capacidade. Assim os diferentes cenários são definidos pelos valores das probabilidades de sucesso e falha que determinam a duração real das atividades e pela ativação do tipo de critério de priorização a utilizar.

3. Os valores da probabilidade de sucesso (evolução de um dia da tarefa) e falha (não evolução da tarefa no dia) que define o comportamento aleatório da duração real das atividades: A implementação da distribuição binomial negativa por meio das probabilidades de sucesso e falha é igual ao modelo anterior. Neste caso se trabalha com um intervalo maior de probabilidade de atraso do projeto, isso, com a finalidade de observar o comportamento dos critérios de priorização num intervalo que abarque cenários pessimistas e otimistas.
4. Os critérios de priorização de atividades: O critério de priorização de atividades é a lógica utilizada na escolha da seguinte atividade a executar. A “priorização fixa” toma essa decisão considerando unicamente as prioridades dos projetos definidas na etapa de planejamento, ou seja, levam-se em conta as características das atividades (se são críticas ou não) e se despreza o estado de consumo dos *buffers*.

A “priorização dinâmica” escolhe a seguinte atividade a executar considerando tanto o nível crítico das atividades como o estado de consumo dos *buffers*. Desde que o modelo representa unicamente as atividades críticas, o critério da escolha é baseado somente no estado de consumo dos BP.

O conjunto de cenários avaliados está formado pela combinação de ambos os parâmetros. Para diferentes probabilidades de atraso do projeto (80%, 60%, 47%, 25%, 20%, 15%, 10% e 5%) são experimentados os três critérios de priorização (fixa, dinâmica e mista), estabelecendo 24 diferentes cenários.

- **A lógica do processo de execução.** - Ademais da lógica explicada na simulação anterior, o modelo adiciona um processo inicial que avalia o estado dos BP. Os projetos são organizados em forma decrescente, segundo o consumo relativo dos respectivos *buffers*. Para tal fim, utiliza-se o cálculo do consumo dos BP em função da sua dimensão planejada. O consumo relativo dos BP é dado por:

$$CR_i = \frac{\text{consumo_do_BP}_i}{\text{duração_planejada_do_BP}_i}$$

Deste modo se assegura que as prioridades outorgadas aos projetos sejam efetivamente baseadas no estado crítico dos *buffers* do projeto.

Para implementar o critério de priorização mista se define que a partir do instante que algum BP atinge o consumo do 45% da sua capacidade, o critério de priorização deixa de ser fixo para adotar o critério de escolha de atividades dinâmica. A representação gráfica do modelo é mostrada na Figura 5.4

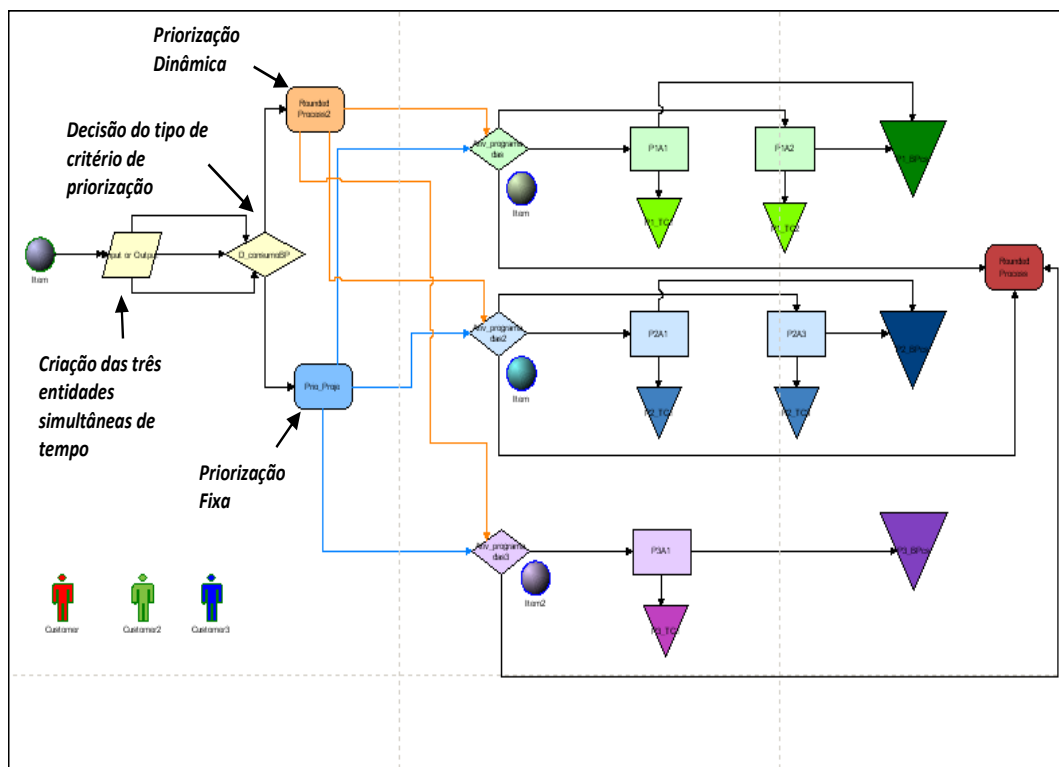


Figura 5.4: Modelo de simulação dos efeitos do critério de priorização de atividades

5.3. Análise dos resultados

5.3.1. Efeitos dos BC

Antes de apresentar a análise dos resultados é importante lembrar que o experimento é baseado na programação “híbrida” que insere os BC dentro do programa projetado, mantendo-se intacto o programa base. Isso ocasiona que os BP tenham um consumo inicial equivalente ao tamanho do BC. É assim que a análise dos efeitos do BC é avaliada em função da quantidade consumida do BP.

5.3.1.1. Dimensionamento dos BC em função do BP

No início da pesquisa, definiu-se que os BC agem como um complemento dos BP, adicionando a segurança que possa absorver os atrasos causados pelo compartilhamento de recursos entre projetos do mesmo sistema. A partir de essa definição, surge a necessidade de esclarecer se, contrariamente ao estabelecido

nas ferramentas de software, os BC teriam que ser dimensionados em base às necessidades específicas de cada projeto e não como uma porcentagem fixa da carga do recurso gargalo.

No capítulo IV se concluiu que quando o consumo do BP ocasionado pela inserção do BC é pequeno, em relação à vizinhança do ponto de início do BP, é um indício de que o BC estaria sub-dimensionado. Similarmente, quando a inserção do BC consome totalmente ou excede a capacidade do BP, é um indício de que este estaria super-dimensionado.

Com essa lógica, estabelece-se que o ponto de referência mínimo de análise está no ponto inicial das curvas de consumo de todos os casos de probabilidade de atraso do projeto. Esse ponto inicial corresponde ao consumo dos BP quando o tamanho do BC é zero, ou seja, que é o consumo ocasionado unicamente pelos atrasos aleatórios das atividades. Deste modo, tamanhos de BC que gerem consumos do BP próximos ao valor de referência são desconsiderados por não cobrir as necessidades de segurança do projeto.

O ponto de referência que indica o super-dimensionamento dos BC é aquele, que ao estar mais longe do ponto inicial, marca o início de uma tendência crescente. Tamanhos de BC que gerem pontos posteriores a esse, também são desconsiderados por exceder a necessidade de segurança do projeto.

Os gráficos do comportamento dos três BP na presença de diferentes níveis de probabilidade de atraso e de diferentes tamanhos de BC são mostrados nas Figura 5.5 a Figura 5.8.

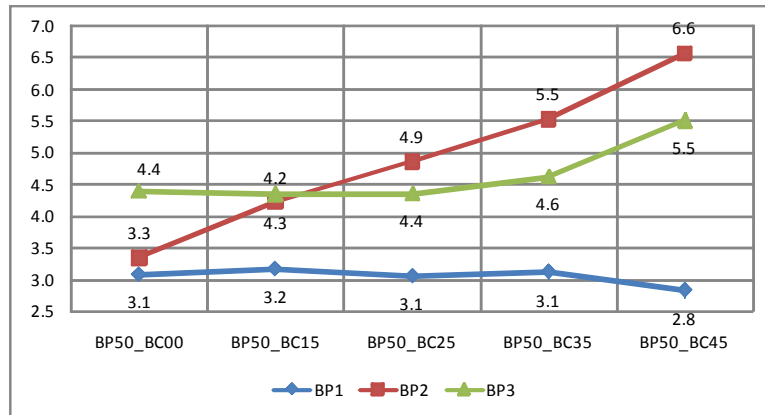


Figura 5.5: Consumo de BP - 25% prob. Atraso

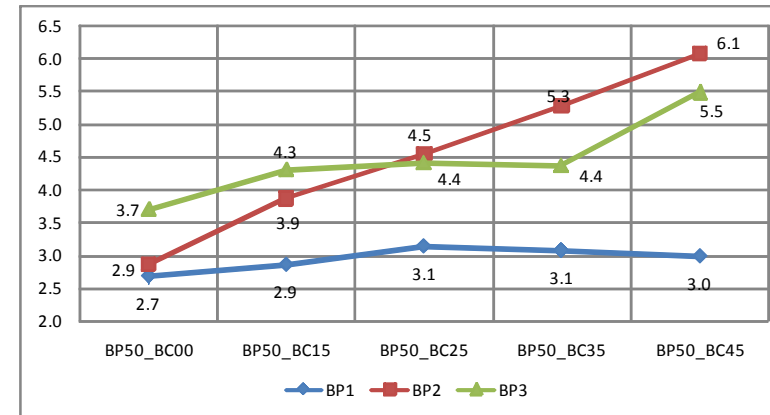


Figura 5.6: Consumo de BP - 20% prob. Atraso

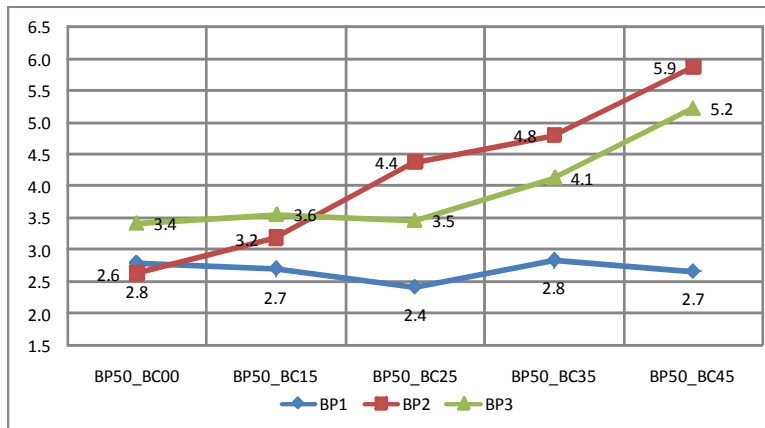


Figura 5.7: Consumo de BP - 15% prob. Atraso

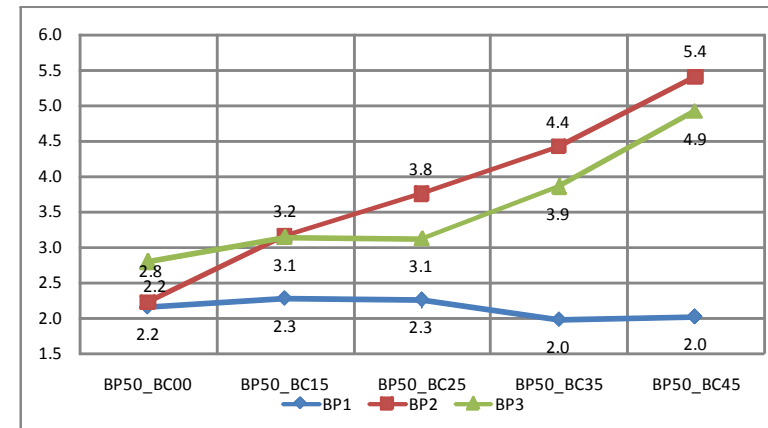


Figura 5.8: Consumo de BP - 10% prob. Atraso

O projeto um (linha azul), ao ser sempre o projeto de maior prioridade, não apresenta atrasos causado por conflitos de recursos nem pelo efeito da inserção de BC. A utilização da mesma série de variações aleatórias em todos os cenários faz com que a curva de consumo do BP1 seja quase constante.

O projeto dois, em todos os casos, apresenta uma curva linearmente crescente (vermelha), indicando que as inserções dos BC atingem rapidamente o ponto de sobre-dimensionamento. Esta condição não permite uma definição clara do tamanho de BC adequado, não obstante, pode-se afirmar que a necessidade de segurança extra do projeto é mínima ou nula.

O projeto três apresenta um comportamento totalmente distinto. Na Figura 5.5 o consumo gerado pelos BC de 15%, 20% e 25%, desenham uma curva quase constante ao redor do ponto de referência. O BC dimensionado a 35% é o primeiro ponto que se desloca da vizinhança do ponto inicial de referência, indicando que este seria o dimensionamento mais adequado para as condições do projeto.

Para o caso de 20% de probabilidade de atraso dos projetos (Figura 5.6), o BC mais adequado oscila entre tamanhos de 25% e 35%. Para o caso de 15% e 10% de probabilidade de atraso, o tamanho de BC mais adequado oscila entre 15 e 25%. Repare-se que com a diminuição da probabilidade de atraso do projeto a distância entre os primeiros pontos da curva define uma tendência constante, similar à gerada pelas variações aleatórias do projeto um. Isso indicaria que a necessidade de adicionar segurança ao projeto é mínima ou nula.

Esta análise, ademais, de dar indícios para a determinação dos BC, permite esclarecer que as características de cada projeto e o grau de dependência que existe entre os diferentes projetos do sistema, fazem que a necessidade de segurança extra dos projetos seja distinta. É assim, que projetos definidos com a menor prioridade, ao depender de um maior número de projetos, não podem utilizar a mesma porcentagem de dimensionamento de BC do que aqueles projetos de maior prioridade. Se esse fosse o caso, certamente o projeto incorreria atrasos. Da mesma maneira, ao dimensionar todos os BC com a porcentagem definida para o projeto menos prioritário, os projetos de prioridade intermediária serão excessivamente deslocados, prejudicando sua competitividade no mercado e provocando o desaproveitamento da disponibilidade dos recursos.

5.3.1.2.

Análise das causas de atraso das atividades

A análise é completada com a decomposição das causas que ocasionam o atraso das atividades. O atraso pode ser dado devido às condições aleatórias do experimento ou à falta de disponibilidade de recursos necessários para executá-las. A última causa é a que deve ser mitigada através da inserção dos BC.

Comparando o efeito que cada tamanho de BC têm na mitigação de atrasos causados por falta de recurso e o atraso efetivo dos projetos, determina-se claramente o tamanho de BC que cada projeto necessita. Assim, também se permite o entendimento das conseqüências que essa mitigação tem no cumprimento da data prometida dos projetos.

As figuras Figura 5.9 a Figura 5.16 mostram, para cada projeto e cada probabilidade de atraso, a relação entre o atraso efetivo dos projetos (dado pelo consumo do BP sem considerar o consumo ocasionado pela inserção dos BC) e o atraso ocasionado por falta de recursos. Ambos os eixos estão descritos em unidades gerais de tempo (t).

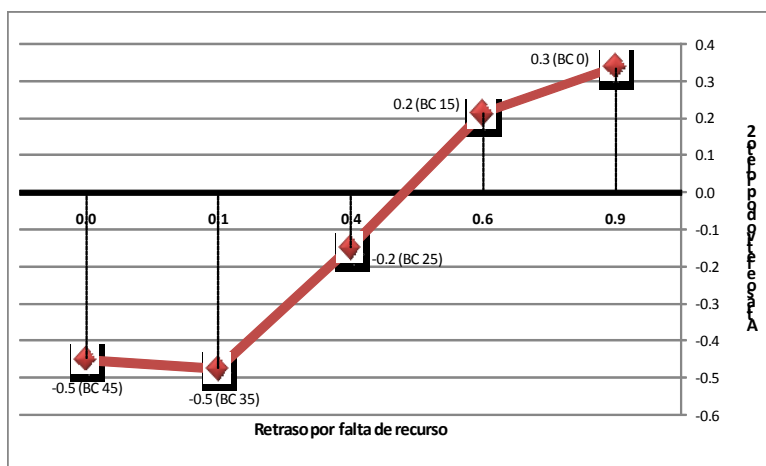


Figura 5.9: Mitigação dos atrasos por falta de recurso
(Projeto dois - 25% de probabilidade de atraso)

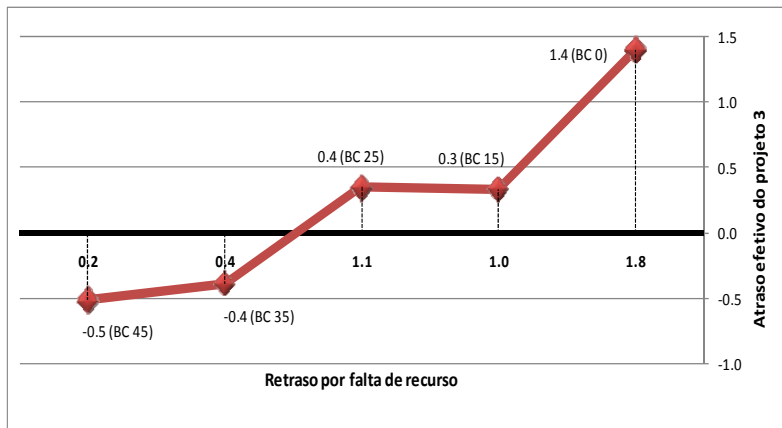


Figura 5.10: Mitigação dos atrasos por falta de recurso
(Projeto três - 25% de probabilidade de atraso)

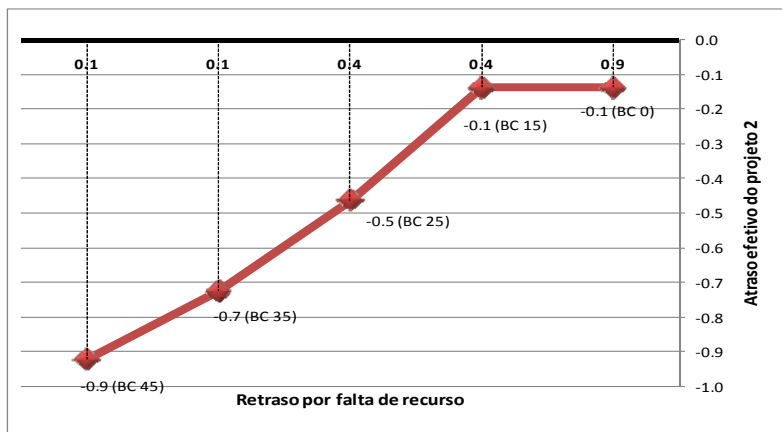


Figura 5.11: Mitigação dos atrasos por falta de recurso
(Projeto dois - 20% de probabilidade de atraso)

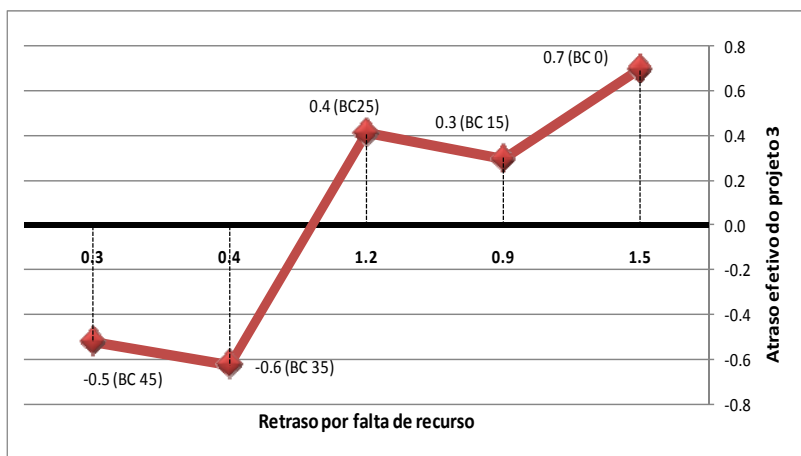


Figura 5.12: Mitigação dos atrasos por falta de recurso
(Projeto três - 15% de probabilidade de atraso)

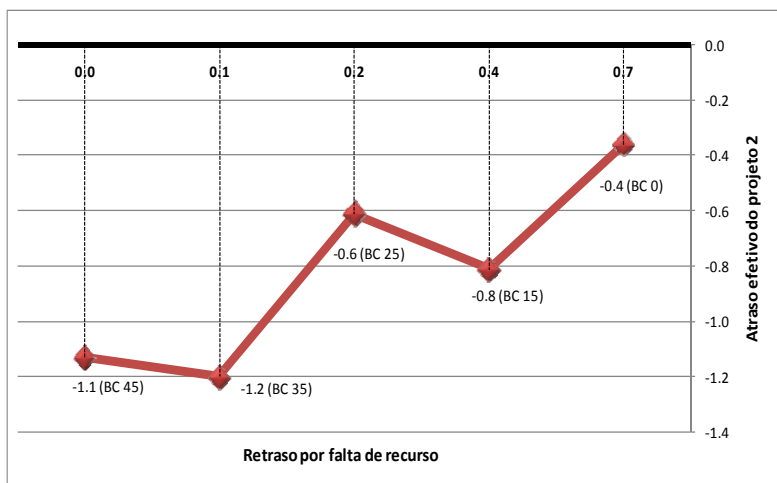


Figura 5.13: Mitigação dos atrasos por falta de recurso
(Projeto dois - 15% de probabilidade de atraso)

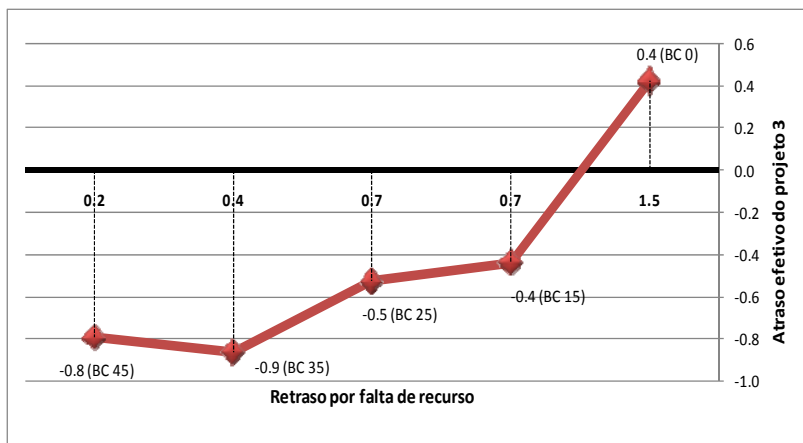


Figura 5.14: Mitigação dos atrasos por falta de recurso
(Projeto três - 15% de probabilidade de atraso)

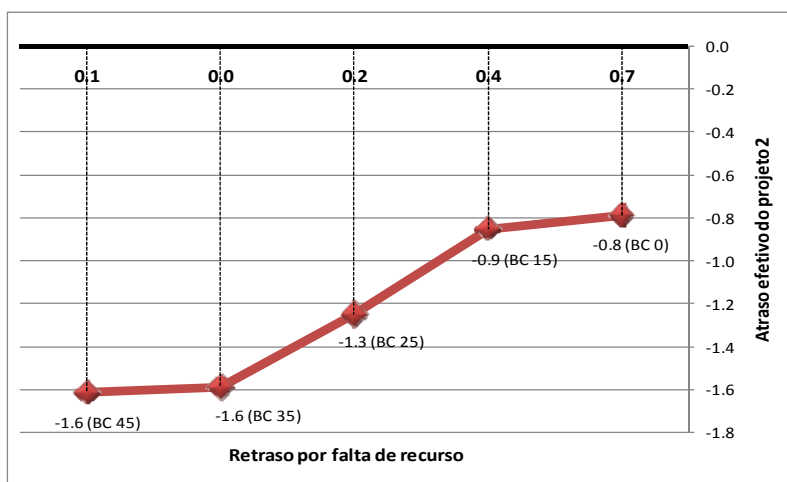


Figura 5.15: Mitigação dos atrasos por falta de recurso
(Projeto dois - 10% de probabilidade de atraso)

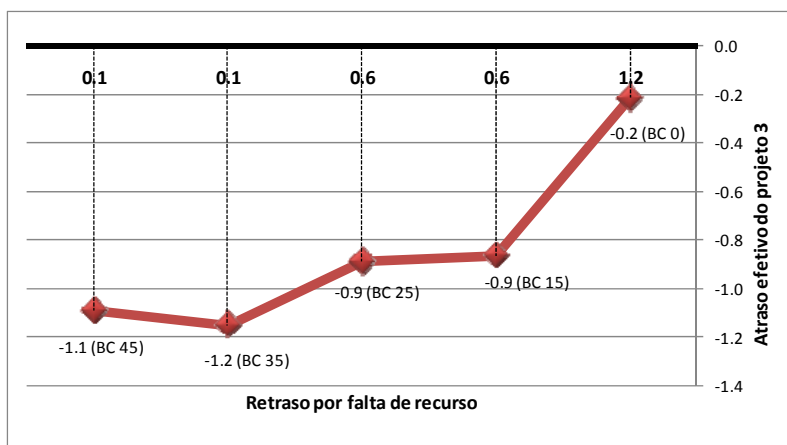


Figura 5.16: Mitigação dos atrasos por falta de recurso
(Projeto três - 10% de probabilidade de atraso)

A Figura 5.9 mostra que, efetivamente como a análise anterior indicou, o projeto dois tem uma necessidade ínfima de aumentar sua segurança. Com 25% de probabilidade de atraso do projeto, os atrasos dos projetos são descritos por frações menores à unidade mínima de medida. Contudo, a determinação de um BC de 25% asseguraria o término do projeto antes da data prometida.

Probabilidades de atraso do projeto menores de 25% não requerem a inserção do BC para o projeto 2 (Figura 5.11, Figura 5.13, Figura 5.15). Repare que em todos os casos do projeto dois a inserção de BC de tamanhos maiores que 25% não geram efeitos significantes na redução do tempo atrasado por falta de recursos (eixo horizontal).

Como se determinou na análise prévia, o projeto três apresenta uma necessidade maior de segurança que o proteja das interdependências com os projetos de maior prioridade. Note-se na Figura 5.10 como a inserção de um BC dimensionado em 15% reduz o tempo de atraso do projeto de 1,4 a 0,3 unidades de tempo, o que é equivalente à redução de 1,8 a 1,0 unidades de tempo atrasadas por causa da falta de recursos. O efeito de mitigação dos atrasos por falta de recurso diminui gradualmente até a inserção de BC dimensionados a 35%. Nesse ponto se atinge o término do projeto antes da data prometida. Incrementos maiores que 35% não geram efeitos significantes na redução de atrasos por falta de recursos.

A necessidade de segurança extra do projeto três, somente muda quando a probabilidade de atraso é definida em 15% (Figura 5.12). Nesse cenário os atrasos

do projeto são menores à unidade de tempo e a inserção de um BC de 15% faz que o projeto seja terminado antes da data prometida. Probabilidades de atraso menores que 15% não requerem a inserção do BC.

5.3.1.3.

“Trade-off” entre mitigação de atrasos ou diminuição do “makespan”

Na análise do efeito de mitigação do BC se utilizou como referência o tamanho de BC que atinge o término antecipado da data prometida. Certamente, em termos de mitigação, esses tamanhos de BC seriam adequados para as características do projeto e do cenário analisado. Contudo, vale à pena lembrar que a data prometida atingida é aquela data resultante da deslocação do início dos projetos. Ou seja, uma data que inclui a inserção dos BC sugere o aumento do *makespan* dos projetos, situação que, poderia resultar desfavorável para algumas empresas.

A inserção dos BC leva consigo a decisão do intercâmbio (*trade-off*) entre: a obtenção de programas que mitiguem consideravelmente os atrasos causados por falta de recursos, criando projetos mais estáveis; e a obtenção de programas que mesmo correndo algum risco de atraso, criam projetos de menor permanência no sistema, aumentando a taxa de rotação de projetos.

A análise contrapõe os atrasos ocasionados pelo compartilhamento de recursos entre projetos (atrasos por falta de recurso) com o tempo inserido pelo BC na deslocação dos projetos (aumento do *makespan*). Ambas as medidas são expressas como uma percentagem da duração enxuta dos projetos. Entendendo-se como duração enxuta àquela que não considera os BP.

È assim que desde a Figura 5.17 à Figura 5.20 se mostra o incremento da percentagem do *makespan* e a diminuição da percentagem do atraso ocasionado por falta de recursos, para cada tamanho de BC. O cálculo de ditas variações é sempre feito em relação ao ponto de início.

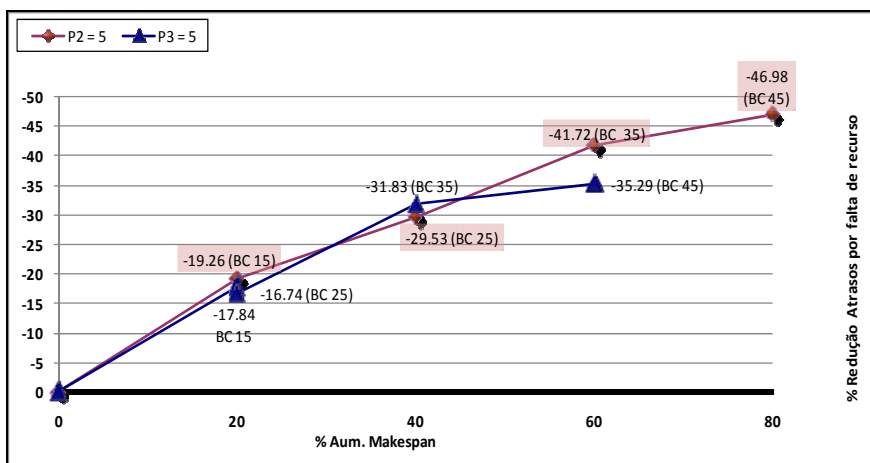


Figura 5.17: Trade-off entre *makespan* e redução de atrasos por falta de recurso (25% probabilidade de atraso)

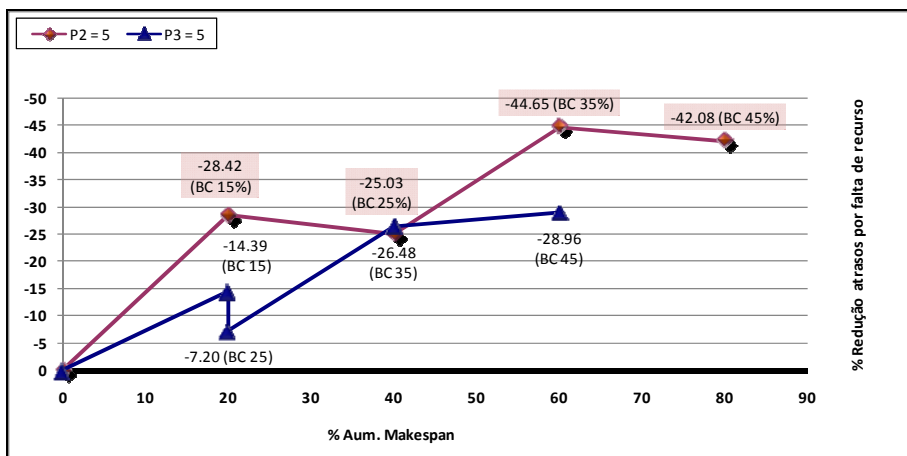


Figura 5.18: Trade-off entre *makespan* e redução de atrasos por falta de recurso (20% probabilidade de atraso)

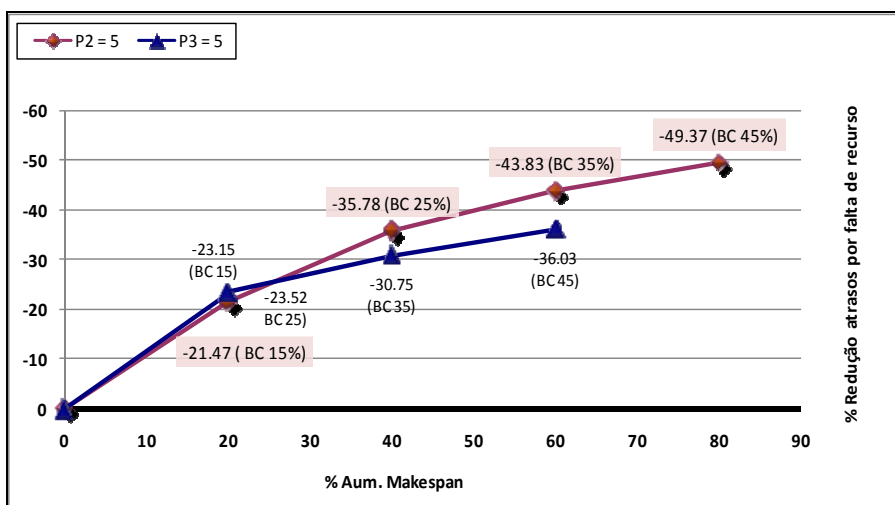


Figura 5.19: Trade-off entre *makespan* e redução de atrasos por falta de recurso (15% probabilidade de atraso)

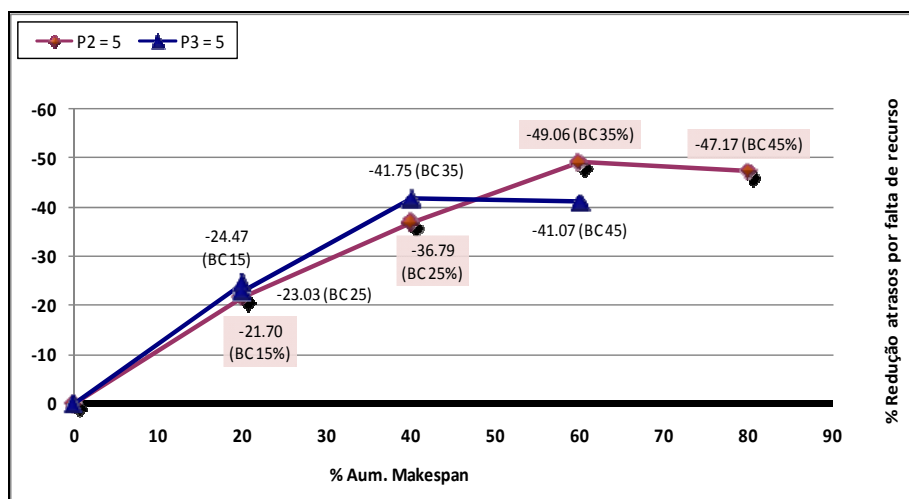


Figura 5.20: Trade-off entre *makespan* e redução de atrasos por falta de recurso (10% probabilidade de atraso)

Repare-se que a porcentagem de tempo que os BC incrementam no *makespan* não é a mesma em todos os projetos. Lembre-se que o BC é determinado em base à capacidade do recurso gargalo. Ou seja, para obter um BC dimensionado a 25%, faz-se o tempo ocupado pelo gargalo nas atividades previas da “seqüência crítica” (as que pertencem a outro projeto), igual a 75%.

O projeto dois tem BC equivalentes a: 15% = 1t, 25% = 2t, 35% = 3t e 45% = 4t.
--

O projeto três tem BC equivalentes a: 15% = 1t., 25% = 1t, 35% = 2t e 45% = 3t.

Como o modelo não contempla atrasos em frações de tempo, o cálculo do BC considera arredondamento superior para qualquer excesso do valor inteiro.

A combinação do *makespan* e do nível de compartilhamento de recursos que possa ser definida como “apropriada”, carece de regras e padrões. Cada situação, incluso cada projeto imerso dentro de um mesmo cenário, tem diferentes características e necessidades que determinam um conceito particular do “apropriado”.

Repare-se como para o projeto dois (Figura 5.17) o incremento de 20% no *makespan* gera a maior **taxa de diminuição** dos atrasos por falta de recurso. Conforme vimos na análise anterior, incrementos do *makespan* maiores de 20% (ou BC maiores de 15%) ocasionam datas de entrega antecipadas.

Não obstante, o objetivo da empresa poderia ser justamente a definição da data de entrega mais antecipada. Em situações onde o projeto não tem urgência de implementação, este poderia ser programado longe dos projetos conflitantes, de tal maneira, que sua duração efetiva seja menor. Se esse fosse o caso, escolher-se-ia o BC de 35%, *buffer* que gera a maior taxa de diminuição de atrasos em relação aos incrementos do *makespan*.

No projeto três a maior taxa de diminuição de atrasos é definida pelo incremento de 40% do *makespan* (ou BC de 35%). Contudo, o projeto poderia ser caracterizado por não permitir grandes postergações da data de início e ser mais permissível com extensões do término do trabalho. Essa situação é comum em projetos de manutenções de chão de fábrica. Se esse fosse o caso, teria que se escolher um BC de 15%, que é o dimensionamento que produz o menor adiamento da data de início, mesmo sacrificando o cumprimento da data de entrega prometida.

A Figura 5.18 mostra como a aleatoriedade na execução das atividades dos projetos predecessores altera o efeito de mitigação dos BC, definindo um comportamento singular da curva de *trade-off* entre o *makespan* e a diminuição de atrasos por falta de recurso.

Tal efeito é mais bem explicado na abstração da Figura 5.21. A área conflitante do projeto um está formada pelas atividades que ademais de ter um alto compartilhamento de recursos com o projeto dois, também concentra a maior quantidade de atrasos aleatórios do projeto. Nesse contexto, a inserção de BC intermediários, como o caso de 25%, faz que as atividades mais conflitantes do projeto dois (marcadas em azul sólido) sejam programadas durante mais tempo dentro da área de conflito do projeto um. Isso ocasiona um menor efeito de mitigação de atrasos do que com a inserção de *buffers* menores, tal e como é visto na análise dos projetos dois e três da Figura 5.18.

Deste modo, conclui-se que o desempenho esperado dos BC depende fortemente das características do cenário no qual são inseridos. É assim, que inserção de BC maiores não significa que necessariamente se produza um melhor resultado na mitigação dos atrasos por falta de recurso. Especialmente, quando o nivelamento da carga de trabalho é baseado num único recurso gargalo.

Obviar os efeitos que os cenários têm sobre o comportamento dos BC pode levar a dimensionar BC que criem conflitos de recursos tão importantes, que acabem gerando um novo gargalo.

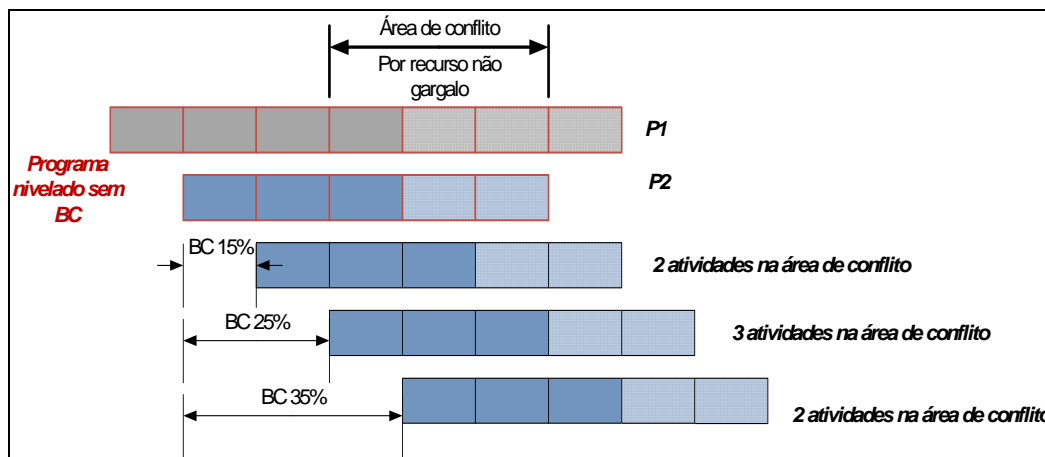


Figura 5.21: Caso especial dos efeitos do BC

5.3.2. Efeitos dos critérios de priorização

Ao estudar o comportamento das abordagens de aplicação durante a etapa de execução (capítulo IV), provocaram-se atrasos em certas atividades que, de acordo com a lógica do programa, permitiam gerar situações específicas de análise. A partir dessas situações, observou-se que as decisões de execução das atividades são baseadas nas prioridades definidas inicialmente para os projetos. Mostrando-se que ao dar prioridade de execução às atividades que correspondiam aos BP mais consumidos, poder-se-ia produzir um resultado melhor.

Esse critério de priorização foi também utilizado como regra de execução da CC/BM na implementação do experimento desenhado por Cohen *et al.* (2004). No artigo se compara o desempenho da metodológica CC/BM em relação a outras regras de programação de múltiplos projetos. Concluindo-se que o desempenho da metodologia, nas condições avaliadas, não supera o resultado obtido através de algumas das regras observadas.

Desta maneira, com base nas observações da análise anterior e na existência de referências literárias que sustentem a validade dos fenômenos observados, levanta-se a hipótese de que o uso de prioridades de execução dinâmicas faz com que os projetos possam ser terminados em menor tempo.

Para poder responder à dito questionamento, analisa-se os resultados do experimento em função do cumprimento da data de entrega prometida dos projetos (incluindo o BP). Os critérios de priorização são avaliados através de uma série de cenários definidos por probabilidades de atraso do projeto entre 5% e 80%.

As Figura 5.22 e Figura 5.23 mostram as curvas que descrevem o efeito que os critérios de priorização fixa, dinâmica e mista tem no sistema.

O eixo vertical expressa o atraso dos projetos, no qual se interpretam os valores negativos como se fossem termos anteriores à data de entrega prometida. O eixo horizontal indica a probabilidade de atraso do projeto com que foram geradas as durações das atividades.

Para facilitar a análise, o gráfico é apresentado em duas partes. Na primeira parte se mostram os cenários altamente pessimistas, descritos por probabilidades de 80%, 60% e 47% (Figura 5.22). Na segunda parte se mostram os cenários descritos por probabilidades de 25%, 20%, 15%, 10% e 5% (Figura 5.23).

O elevado número de atrasos totais do sistema era de se esperar. Esses atrasos são ocasionados tanto pela alta probabilidade de atraso imersa nos cenários, como pela ausência de BC nos projetos. Contudo, o uso da priorização fixa produz um resultado ligeiramente melhor em situações onde a probabilidade média de atraso é superior a 60%.

Conforme diminui a probabilidade de atraso, as curvas de prioridades fixas e dinâmicas tendem a coincidir, obtendo o mesmo resultado (ou resultados muito próximos) na consecução da data de término prometida.

Com base nesses resultados, certamente, não pode se afirmar que o uso de prioridades dinâmicas melhora o desempenho da programação de projetos. Pelo contrario, o desempenho dos projetos através da utilização de prioridades dinâmicas tende a ser pior do que o resultado obtido através da manutenção das prioridades fixas.

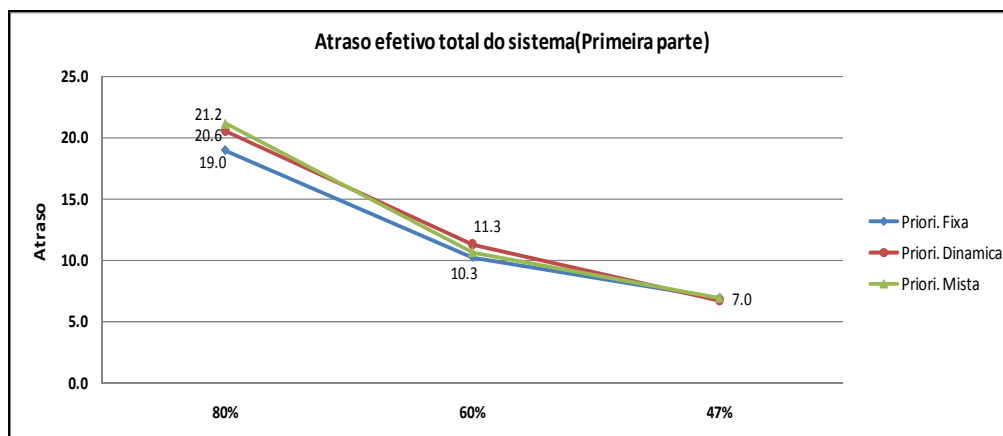


Figura 5.22: Efeitos dos critérios da priorização no sistema

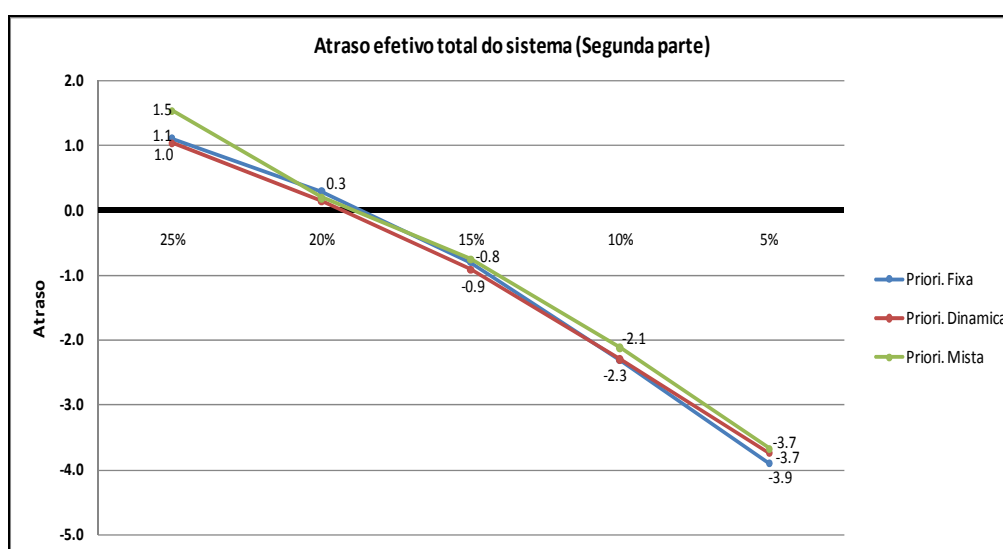


Figura 5.23 : Efeitos dos critérios da priorização no sistema

Para entender o porquê do critério de priorização dinâmica não gerar os resultados esperados, compara-se os efeitos que os critérios de priorização têm sobre cada projeto num ambiente de elevadas probabilidades de atraso.

Quando os projetos são executados segundo suas prioridades definidas inicialmente (priorização fixa), as atividades do projeto um são sempre as primeiras em serem executadas. Ao aplicar o critério de priorização dinâmica ou mista, as datas de término do projeto um acabam sendo muito mais atrasadas do que o adiantamento ganho nas execuções dos projetos dois e três. (Figura 5.24, Figura 5.25 e Figura 5.26)

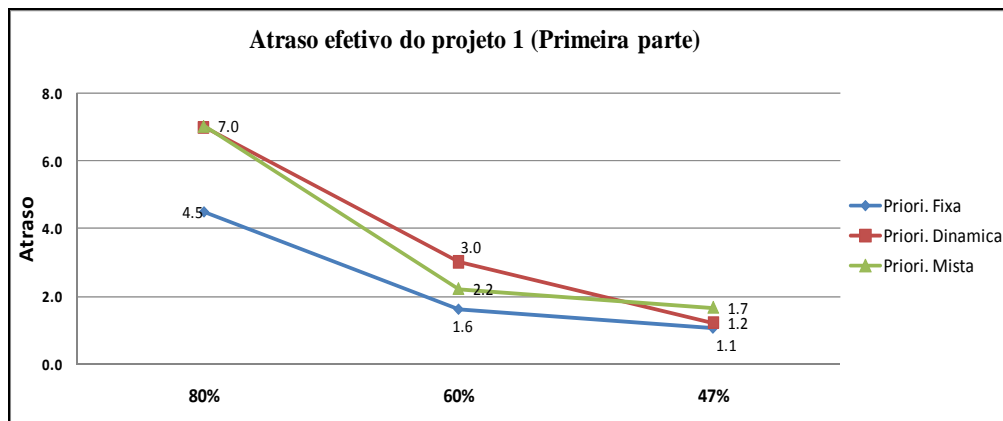


Figura 5.24 : Efeitos dos critérios da priorização no projeto 1

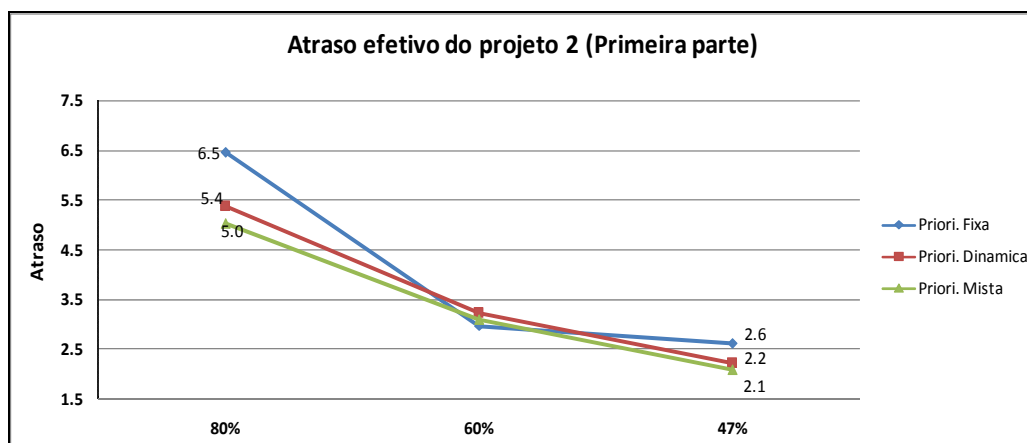


Figura 5.25 : Efeitos dos critérios da priorização no projeto dois

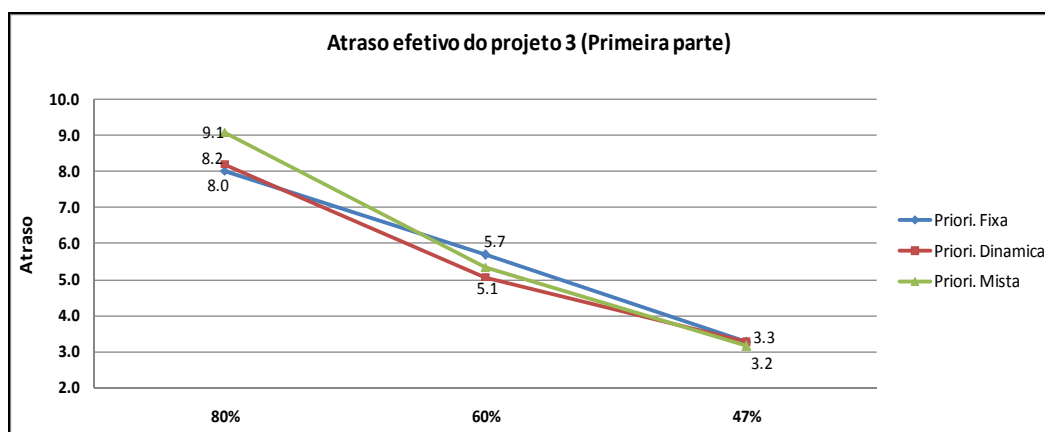


Figura 5.26: Efeitos dos critérios da priorização no projeto três.

Esse fenômeno pode ser explicado pelos conceitos de variações de *causas comuns* e variações de *causas especiais*. Segundo Leach (2005) as variações são de causas comuns quando a fonte da variação é o próprio sistema, e são de causas

especiais quando as variações são provocadas por eventos provenientes fora do sistema.

É na presença dessas últimas variações que a gerência de projetos (e de qualquer sistema produtivo) deve tomar ações corretivas. Leach (2005) enfatiza que o fato de tratar as variações comuns do sistema como se fossem de causas especiais, gera o incremento da variância total do sistema, e conseqüentemente seu atraso.

Na execução de múltiplos projetos, os atrasos ocasionados tanto pela aleatoriedade da duração das atividades como pelo compartilhamento de recursos são variâncias comuns do sistema. Ditas variações, são mitigadas através da inserção dos diferentes *buffers* durante a etapa de planejamento do programa. É assim que, a tentativa de melhorar os resultados da execução do sistema, dando prioridade às atividades dos projetos mais críticos (BP mais consumido) ocasiona o desequilíbrio do programa. Gerando-se uma situação caótica, na qual os recursos vão de um projeto a outro criando um tipo de múltiplas tarefas (*multi-tasking*).

A razão pela qual tais efeitos não foram percebidos na análise anterior, é que as variações de execução foram geradas segundo uma lógica similar aos casos de probabilidade de atraso menores de 25%. A simplicidade do problema estudado deixa identificar os efeitos de usar prioridades dinâmicas, somente na presença de altas probabilidades de atraso.

Contudo, a negatividade dos efeitos incrementa-se conforme a complexidade do problema aumenta. Do mesmo modo, as conseqüências empioram quando se agregam BC na análise.