ALEATORIEDADE E AGREGAÇÃO DE ESTADOS NO MODELO HIPERCUBO DE FILAS

Caio Vitor Beojone
Universidade Estadual Paulista – UNESP
Av. Eng. Luíz Edmundo C. Coube 14-01
beojone@hotmail.com

Regiane Máximo de Souza
Universidade Estadual Paulista – UNESP
Av. Eng. Luíz Edmundo C. Coube 14-01
regiane@feb.unesp.br

RESUMO

Este artigo tem por objetivo apresentar maneiras de se representar a aleatoriedade no despacho de servidores no modelo hipercubo de filas e propor um modelo com agregação de servidores para ter ganhos computacionais. O modelo hipercubo é bastante reconhecido como ferramenta de análise de Sistemas de Atendimento Emergencial (SAEs). Contudo, é notada a dificuldade de se utilizá-lo em sistemas com muitos servidores, por causa do custo computacional para resolver tais modelos. O modelo com agregação de estados mostra uma possibilidade de modelagem com menor custo computacional, mas com bons resultados, a fim de ser utilizado em pesquisas futuras.

PALAVRAS CHAVE: Agregação de servidores, modelo hipercubo, SAMU.

1. Introdução

O planejamento e a operação de Sistemas de Atendimento Emergencial (SAEs) têm impacto direto na qualidade de vida da população [Souza et al. 2015]. Nesse sentido, a Pesquisa Operacional vem sendo utilizado nos últimos 50 anos como uma importante ferramenta para o estudo de tais sistemas. [Simpson e Hancock 2009]

O modelo hipercubo é uma ferramenta reconhecida na literatura como eficaz na análise de sistemas de fila espacialmente distribuídas. Ele pode ser aplicado aos sistemas de delivery, dos bombeiros, da polícia, de guinchos, SAMU, de reparos emergenciais, dentre outros. Trata-se uma ferramenta analítica que pode ser utilizada para se obter medidas de desempenho e propor alterações nos sistemas com a finalidade de se propor um melhor atendimento às necessidades dos usuários [Larson e Odoni 2007].

Contudo, o modelo possui alto custo computacional, visto o crescimento exponencial do número de equações de equilíbrio quanto maior o número de servidores do sistema [Chiyoshi et al. 2011; Larson e Odoni 2007].

A experiência mostra que a aleatoriedade no envio de servidores é uma característica razoavelmente comum, em especial onde há co-localização de servidores não diferenciados, conforme visto em trabalhos como em Takeda [2000] e Takeda et al. [2004, 2007]. Essas características levam a não necessidade de descrever os servidores individualmente [Luque 2006].

Neste sentido, o objetivo deste trabalho é apresentar as diferentes variações do modelo hipercubo para englobar a aleatoriedade no despacho através de um modelo reduzido possibilitando a comparação entre os métodos. Além disso, é proposta uma alteração nos modelos a fim de diminuir o problema com o custo computacional do modelo utilizando agregação de servidores.

Na próxima seção tem-se uma breve apresentação do modelo hipercubo original e suas características. Em seguida, na Seção 3, é apresentado o modelo reduzido e suas principais características para a modelagem. Na Seção 4 tem-se as explicações e resultados para cada um dos modelos levantados. Adiante na Seção 5, é feita uma breve comparação dos resultados obtidos a fim de mostrar os ganhos da utilização do modelo proposto. Por fim, a Seção 6 traz as conclusões do trabalho.

2. O modelo hipercubo clássico

Desenvolvido por Larson [1974], o modelo hipercubo de filas trata-se de um modelo analítico, baseado em sistemas de filas espacialmente distribuídas em que a ideia é fazer uma expansão dos estados de um sistema *M/M/m* buscando representar os servidores individualmente. Dessa maneira, ele permite a elaboração de preferências de despacho mais complicadas. Para encontrar a solução do sistema é preciso elaborar e resolver um conjunto de equações de equilíbrio (*steady state*), os resultados são as probabilidades de ocorrência de estado de equilíbrio. A partir da solução do modelo podem ser calculadas diversas medidas de desempenho para o sistema, como: carga de trabalho dos servidores, frequências de despacho de servidores, tempos médios de viagem, tempos médios de fila, entre outras.

Larson e Odoni [2007] mostra uma série de 9 hipóteses que precisam ser satisfeitas para a aplicação do modelo hipercubo em sua forma clássica.

1. Existência de átomos geográficos: a região onde são prestados os serviços do sistema deve ser dividida em N_A átomos geográficos, sendo que cada átomo corresponde a uma fonte independente de chamados e também possuem políticas de despacho;
2. Processo de chegada conforme a distribuição de Poisson: os usuários de cada átomo solicitam chamados por meio do processo de Poisson, sendo os chamados independentes entre si. Além disso, as taxas de chegada, λ_j, de chamados de cada átomo deve ser conhecida;

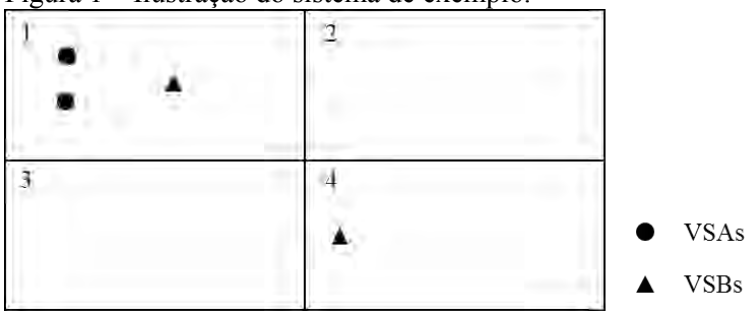
3. Tempos de viagem dos servidores: os tempos de viagem, τ_{ij}, de cada servidor *i* para o átomo *j* devem ser conhecidos ou estimados;
4. Servidores: existem *N* servidores espacialmente distribuídos ao longo do sistema, sendo que cada um pode se deslocar e atender a qualquer um dos átomos;
5. Localização do servidor: a localização do servidor no sistema deve ser conhecida ao menos probabilisticamente. Sendo que o servidor pode se mover pelos átomos ou ficar fixo em um deles;
6. Despacho simples: para atender a qualquer chamado é enviado apenas 1 servidor para o local. Se não houverem servidores disponíveis os chamados entrarão em fila ou serão considerados perdas do sistema;
7. Política de despacho dos servidores: há uma lista de preferência de despacho para cada átomo, ou seja, deve ser obedecida uma ordem de envio dos servidores para os chamados;
8. Tempo de serviço: o tempo de serviço de um servidor engloba tempo de *setup*, tempo de viagem e tempo em cena; e
9. Dependência do tempo de serviço em relação ao tempo de viagem: o tempo de viagem deve ser considerado uma variável de segunda ordem no tempo total de serviço, quando comparado ao tempo em cena e preparação da equipe.

As extensões do modelo clássico trabalham com alterações nessas hipóteses de acordo com a necessidade do sistema estudado. Exemplos de aplicação do modelo em sua forma original podem ser facilmente encontrados na literatura para a melhor compreensão do funcionamento do modelo. Alguns exemplos a serem citados são Chiyoshi et al. [2000] e Larson [1974].

3. Sistema reduzido

Considere um sistema composto por 4 átomos geográficos e 4 servidores que não admite fila de espera. Os servidores são separados em avançados (VSAs), têm preferência de envio para casos graves, e básicos (VSBs), porém sem a ocorrência de backup parcial. Sendo assim, os VSBs podem atender aos chamados graves, caso os VSAs estejam ocupados e para os chamados mais simples vice-versa. Além disso, os servidores estão distribuídos no sistema da seguinte forma: os VSAs encontram-se no átomo 1, juntamente de um VSB, enquanto o outro VSB encontra-se no átomo 4. A Figura 1 mostra essa configuração do sistema. Chamaremos o VSB localizado no átomo 1 de VSB1 e o VSB localizado no átomo 4 de VSB2.

Figura 1 – Ilustração do sistema de exemplo.



O despacho dos servidores obedece uma matriz de preferência de despacho não totalmente definida. Os servidores avançados são escolhidos aleatoriamente para atender aos chamados graves em todos os átomos do sistema, assim como, caso ambos VSBs estejam ocupados, são escolhidos aleatoriamente para o backup dos chamados básicos. Por outro lado, o VSB1 é o primeiro backup para chamados graves no átomo 1 e o VSB2 para o átomo 4, para os átomos 2 e 3 eles são escolhidos aleatoriamente. Para os chamados mais simples, o VSB1 tem preferência para o envio no átomo 1 e o VSB2 para o átomo 4, mas são escolhidos aleatoriamente para os átomos 2 e 3.

A Tabela 1 mostra as taxas de chegada e de serviço do sistema, lembrando que para a separação dos chamados graves e não-graves foi utilizada a técnica de *layering*, utilizada em Takeda et al. [2004]. Com isso, os chamados graves são representados pelos subátomos *a*, enquanto os não-graves pelos *b*.

Tabela 1 – Taxas de chegada e de serviço do sistema de exemplo.

Átomos	**Taxas (λ)**	**Servidores**	**Taxas (μ)**
1 *a*	0,20	VSA1	1,1
1 *b*	0,50	VSA2	1,1
2 *a*	0,10	VSB1	1,6
2 *b*	0,25	VSB2	1,5
3 *a*	0,10		
3 *b*	0,30		
4 *a*	0,10		
4 *b*	0,45		

4. Aplicações com aleatoriedade no modelo hipercubo

Aqui são apresentados os modelos que consideram a aleatoriedade no despacho dos servidores. Além de uma breve apresentação, eles são utilizados para modelar o sistema reduzido apresentado. Note que apenas se considerou modelos hipercubo onde os servidores podem assumir dois estados: {0} se livre e {1} se ocupado. Com isso, não foram incluídos trabalhos cujos servidores são representados por 3 ou mais estados como Boyaci e Geroliminis [2015].

4.1. Aleatoriedade em Takeda et al. [2007]

Neste trabalho, não há uma representação formal da aleatoriedade no sistema de forma que ela possa ser indicada diretamente nas equações de equilíbrio. A aleatoriedade é dada pela repetida solução do sistema utilizando preferências fixas, como no modelo hipercubo clássico de Larson [1974], e cálculo da média das probabilidades de cada estado ao final.

Esta forma possui uma série de problemas. Primeiro, não há representação formal da aleatoriedade, estando essa sujeita à algoritmos de geração de números aleatórios na criação das matrizes de preferência de despacho, e também sendo necessário um grande número de repetições para se obter uma solução satisfatória. Segundo, o modelo hipercubo possui um custo computacional muito elevado para sistemas com muitos servidores e a repetida solução do sistema aumenta ainda mais tal custo.

Dessa forma, o sistema de exemplo com seus 4 servidores e estipulando que 5 repetições da resolução são necessárias para uma representação adequada, teríamos um total de $(2^4) * 5 = 80$ equações a serem resolvidas.

Uma das possíveis matrizes de preferência de despacho seria como a dada pela Tabela 2. Lembrando que os servidores são denominados da seguinte maneira: VSAs são os servidores 1 e 2, VSB1 é o servidor 3 e o VSB2 é o servidor 4. Observe que as condições de despacho foram respeitadas, porém essa é uma tabela fixa, não representa a aleatoriedade do sistema e sim uma situação possível, para se ter uma representação plena da aleatoriedade todas as matrizes possíveis deveriam ser resolvidas e terem sua média encontrada, o que é altamente inviável computacionalmente.

Tabela 2 – Exemplo de matriz de preferência de despacho.

Átomos	**Preferências**			
	1º	**2º**	**3º**	**4º**
1 *a*	1	2	3	4
1 *b*	3	4	1	2
2 *a*	1	2	4	3
2 *b*	3	4	1	2
3 *a*	2	1	4	3
3 *b*	4	3	1	2
4 *a*	1	2	4	3
4 *b*	4	3	2	1

Com isso, uma possível matriz dos coeficientes do sistema de equações de equilíbrio para o modelo ficaria como na Tabela 3 a seguir.

Tabela 3 – Possível matriz dos coeficientes para o modelo de Takeda et al. [2007].

Estados	0000	1000	0100	0010	0001	1100	1010	1001	0110	0101	0011	1110	1101	1011	0111	1111
0000	-λ	μ1	μ2	μ3	μ4	0	0	0	0	0	0	0	0	0	0	0
1000	λ1a+λ2a+λ4a	-(λ+μ1)	0	0	0	μ2	μ3	μ4	0	0	0	0	0	0	0	0
0100	λ3a	0	-(λ+μ2)	0	0	μ1	0	0	μ3	μ4	0	0	0	0	0	0
0010	λ1b+λ2b	0	0	-(λ+μ3)	0	0	μ1	0	μ2	0	μ4	0	0	0	0	0
0001	λ3b+λ4b	0	0	0	-(λ+μ4)	0	0	μ1	0	μ2	μ3	0	0	0	0	0
1100	0	λa	λa	0	0	-(λ+μ1+μ2)	0	0	0	0	0	μ3	μ4	0	0	0
1010	0	λ1b+λ2b	0	λ1a+λ2a+λ4a	0	0	-(λ+μ1+μ3)	0	0	0	0	μ2	0	μ4	0	0
1001	0	λ3b+λ4b	0	0	λ1a+λ2a+λ4a	0	0	-(λ+μ1+μ4)	0	0	0	0	μ2	μ3	0	0
0110	0	0	λ1b+λ2b	λ3a	0	0	0	0	-(λ+μ2+μ3)	0	0	μ1	0	0	μ4	0
0101	0	0	λ3b+λ4	0	λ3a	0	0	0	0	-(λ+μ	0	0	μ1	0	μ3	0

0011	0	0	0	λb	λb	0	0	0	0	0	-(λ+μ3+μ4)	0	0	μ1	μ2	0
1110	0	0	0	0	0	λ1a+λ1b+λ2b	λa	0	λa	0	0	-(λ+μ1+μ2+μ3)	0	0	0	μ4
1101	0	0	0	0	0	λ2a+λ3a+λ3b+λ4a+λ4b	0	λa	0	λa	0	0	-(λ+μ1+μ2+μ4)	0	0	μ3
1011	0	0	0	0	0	0	λb	λb	0	0	λ1a+λ1b+λ2a+λ2b+λ3b+λ4a	0	0	-(λ+μ1+μ3+μ4)	0	μ2
0111	0	0	0	0	0	0	0	0	λb	λb	λ3a+λ4b	0	0	0	-(λ+μ2+μ3+μ4)	μ1
1111	0	0	0	0	0	0	0	0	0	0	0	λ	λ	λ	λ	-μ

A partir da média das probabilidades encontradas em cada solução é possível calcular as cargas de trabalho para os servidores, conforme a Tabela 4 adiante.

Tabela 4 – Carga de trabalho dos servidores calculada pelo modelo de Takeda et al. [2007].

Servidor	Carga de Trabalho
1	0,3170
2	0,2864
3	0,3841
4	0,4157

4.2. Aleatoriedade em Chiyoshi et al. [2011]

Neste trabalho já há um esforço para a representação formal da aleatoriedade, sendo, então, possível sua indicação diretamente nas equações de equilíbrio. Isso elimina alguns dos problemas encontrados no método de Takeda et al. [2004, 2007] como a necessidade de repetição da resolução do modelo e a dependência de algoritmos de geração de números aleatórios.

Contudo tem-se apenas a representação para um sistema de servidores centralizados e com aleatoriedade total, ou seja, em nenhum caso há a preferência pela escolha de um servidor, essa dependendo exclusivamente dos servidores disponíveis, visto que a matriz de preferência de despacho é totalmente descartada.

Como fim de resolver esta questão far-se-á uma adaptação ao modelo, uma matriz de preferência de despacho semi-definida. A Tabela 5 mostra como fica essa matriz. Note que não é mais definido o servidor para a preferência e sim a preferência para o servidor. Existem servidores com a mesma prioridade de envio (servidores 1 e 2 para o átomo 1*a*), isso significa que eles têm a mesma preferência e sua escolha é aleatória caso ambos estejam disponíveis.

Tabela 5 – Matriz de preferência de despacho para Chiyoshi et al. [2011] adaptado.

Átomos	**Servidores**			
	1	**2**	**3**	**4**
1 *a*	1º	1º	2º	3º
1 *b*	3º	3º	1º	2º
2 *a*	1º	1º	2º	2º
2 *b*	2º	2º	1º	1º
3 *a*	1º	1º	2º	2º
3 *b*	2º	2º	1º	1º
4 *a*	1º	1º	3º	2º
4 *b*	3º	3º	2º	1º

Como a escolha de envio do servidor pode ser aleatória, os servidores dividirão por igual a taxa de chegada. Por exemplo, supondo o estado {1000}, ou seja, VSBs livres, a taxa de entrada para o estado {1010} a partir de um chamado do átomo 2*b*, será $\lambda_{2b}/2$ e para o estado {1001} também. Dessa forma pode-se construir a matriz dos coeficientes para esse modelo, conforme a Tabela 6.

Tabela 6 – Matriz dos coeficientes para o modelo de Chiyoshi et al. [2011] adaptado.

Estados	0000	1000	0100	0010	0001	1100	1010	1001	0110	0101	0011	1110	1101	1011	0111	1111
0000	-λ	μ1	μ2	μ3	μ4	0	0	0	0	0	0	0	0	0	0	0
1000	λa/2	-(λ+μ1)	0	0	0	μ2	μ3	μ4	0	0	0	0	0	0	0	0
0100	λa/2	0	-(λ+μ2)	0	0	μ1	0	0	μ3	μ4	0	0	0	0	0	0
0010	λ1b+(λ2b+λ3b)/2	0	0	-(λ+μ3)	0	0	μ1	0	μ2	0	μ4	0	0	0	0	0
0001	(λ2bλ3b)/2+λ4b	0	0	0	-(λ+μ4)	0	0	μ1	0	μ2	μ3	0	0	0	0	0
1100	0	λa	λa	0	0	-(λ+μ1+μ2)	0	0	0	0	0	μ3	μ4	0	0	0
1010	0	λ1b+(λ2b+λ3b)/2	0	λa/2	0	0	-(λ+μ1+μ3)	0	0	0	0	μ2	0	μ4	0	0
1001	0	(λ2b+λ3b)/2+λ4b	0	0	λa/2	0	0	-(λ+μ1+μ4)	0	0	0	0	μ2	μ3	0	0
0110	0	0	λ1b+(λ2b+λ3b)/2	λa/2	0	0	0	0	-(λ+μ2+μ3)	0	0	μ1	0	0	μ4	0
0101	0	0	(λ2b+λ3b)/2+λ4b	0	λa/2	0	0	0	0	-(λ+μ2+μ4)	0	0	μ1	0	μ3	0
0011	0	0	0	λb	λb	0	0	0	0	0	-(λ+μ3+μ4)	0	0	μ1	μ2	0
1110	0	0	0	0	0	λ1+(λ2+λ3)/	λa	0	λa	0	0	-(λ+μ1	0	0	0	μ4

						2						+μ2+μ3)				
1101	0	0	0	0	0	(λ2+λ3)/2+λ4	0	λa	0	λa	0	0	-(λ+μ1+μ2+μ4)	0	0	μ3
1011	0	0	0	0	0	0	λb	λb	0	0	λ/2	0	0	-(λ+μ1+μ3+μ4)	0	μ2
0111	0	0	0	0	0	0	0	0	Λb	λb	λ/2	0	0	0	-(λ+μ2+μ3+μ4)	μ1
1111	0	0	0	0	0	0	0	0	0	0	0	λ	λ	λ	λ	- μ

Diretamente a partir da matriz obtém-se o resultado, sem novas repetições, ou seja, apenas 16 equações são resolvidas, contra as 80 resolvidas anteriormente. Calcula-se agora as cargas de trabalho para os servidores (Tabela 7). Note que os VSAs possuem a mesma carga de trabalho, o que já era esperado, já que são escolhidos aleatoriamente em todos os casos e são homogêneos.

Tabela 7 – Carga de trabalho dos servidores para o modelo de Choyoshi et al. [2011] adaptado.

Servidor	**Workload**
1	0,3026
2	0,3026
3	0,4006
4	0,3995

4.3 Proposta de representação da aleatoriedade

A agregação de servidores é a ferramenta proposta para situações onde dois ou mais servidores não precisam ser representados individualmente. As características que possibilitam isso são a co-localização e a indiferença na escolha para despacho (mesma preferência). Usando a nomenclatura de estados para representar as situações possíveis para dois servidores nessa situação tem-se {00}, {10}, {01} e {11} como não há diferenças entre os servidores, não precisa das representações {10} e {01} (LUQUE, 2006). Neste caso, a representação pode ser feita por três estados, {0} – os dois servidores livres –, {1} – um dos servidores ocupados – e {2} os dois servidores ocupados.

Com isso, cria-se um agrupamento de servidores eliminando estados com a desnecessária diferenciação dos servidores. Não devendo mais se tratar os servidores individualmente, mas sim por agrupamento. Uma mudança no cálculo do número de estados do sistema ocorre. No lugar da equação apresentada por Larson e Odoni (2007) $2^N + Q$, tem-se a Equação (1) a seguir, onde m é o número de agrupamentos e n_k é o número de servidores do agrupamento k. Note que não é necessário haver mais de um servidor no agrupamento, caso não se consiga formar nenhum agrupamento tem-se o mesmo número de estados que anteriormente.

$$M = \prod_{k=1}^{m} (n_k + 1) + Q \qquad (1)$$

A mudança do número de estados também se reflete no espaço de estados. Agregar servidores faz com que cada dimensão do espaço seja referente a um agrupamento e não mais a um servidor individualmente. Sendo assim, o número de dimensões é igual ao número de agrupamentos.

A partir do espaço de estados, é possível trabalhar com as transições entre estados. O sistema ainda obedece uma matriz de preferência de despacho, porém esta possui alterações. Ao

invés de indicar o servidor (agrupamento) de uma dada preferência, inverte-se, representa-se a preferência do servidor, conforme mostra a Figura 2 adiante. Note que aqui podem existir servidores (agrupamentos) com a mesma preferência, neste caso, eles são escolhidos aleatoriamente conforme a disponibilidade. No caso de agrupamentos com mais de um servidor disponível, o agrupamento tem probabilidade de escolha proporcional ao número de seus servidores livres. Por exemplo, dois agrupamentos possuem servidores para despacho, um agrupamento com um servidor livre e o outro com dois servidores livres, o primeiro agrupamento receberá $1/3$ dos chamados, enquanto o outro agrupamento receberá $2/3$.

Figura 2 – Alterações na matriz de preferência de despacho.

Átomos	Preferências		
	1º	2º	3º
1	1	2	3

→

Átomos	Servidores		
	1	2	3
1	1º	2º	3º

Agora já se pode obter as probabilidades de estado para o sistema e, com isso, as cargas de trabalho dos servidores. Contudo, é importante lembrar que agora não se saberá qual servidor de um agrupamento estará ocupado. Neste caso, como a escolha entre os servidores do agrupamento é aleatório, assume-se que a carga é dividida entre os servidores do agrupamento. Por exemplo, em um agrupamento com dois servidores em um estado onde apenas um servidor está ocupado, tem-se que cada um ficou ocupado metade das vezes naquele estado. Por fim, a equação (5) mostra como é feito o cálculo da carga de trabalho ρ_{ki} do servidor ki. Onde, B é um estado pertencente ao conjunto de estados M; n_{kB} é o número de servidores do agrupamento k que se encontram ocupados no estado B e P_B é a probabilidade do estado B.

$$\rho_{ki} = \sum_{B \epsilon M} \frac{n_{kB} \cdot P_B}{n_k} \tag{2}$$

Por fim, além das cargas de trabalho pode-se obter as frequências de despacho dos servidores. Elas podem ser calculadas através da Equação (3) adiante. Primeiramente observa-se que ela é separada em frequência de despacho onde não há espera em fila e onde há espera em fila. Na frequência de despacho sem espera, λ_j/λ é a fração de chamados do átomo j, μ_{ki}/μ_k é a fração da taxa de serviço do agrupamento k e P_D é a probabilidade de envio do servidor k ao átomo j. Note que a probabilidade de envio deve ser dividida entre os servidores com mesma preferência disponíveis para o atendimento. Na frequência de despacho com espera tem-se novamente a fração dos chamados do átomo j, P_S é a probabilidade de saturação do sistema e μ_{ki}/μ é a fração da taxa de serviço total representada pelo servidor ki.

$$\begin{aligned} f_{ki,j} &= f_{ki,j}^{(nq)} + f_{ki,j}^{(q)} \\ f_{ki,j}^{(nq)} &= \frac{\lambda_j}{\lambda} \cdot \frac{\mu_{ki}}{\mu_k} \cdot \sum_{D \in E_{k,j}} P_D \\ f_{ki,j}^{(q)} &= \frac{\lambda_j}{\lambda} \cdot P_S \cdot \frac{\mu_{ki}}{\mu} \end{aligned} \tag{3}$$

O cálculo dos tempos de viagem não sofre alterações neste modelo.

A fim de diminuir o número equações de equilíbrio pode-se pensar num conjunto de servidores homogêneos, co-localizados e com as mesmas preferências de envio entre si como um agrupamento de servidores. Conforme apresentado por Luque [2006], a aglutinação de estados é válida como forma de conseguir modelos com menos equações, todavia não a aplicou ou desenvolveu. Por exemplo, num sistema com dois servidores nessas condições não precisam ser representados pelos estados {00}, {10}, {01} e {11} mas sim pelos estados {0} (nenhum servidor ocupado), {1} (um servidor ocupado) e {2} (dois servidores ocupados), que dizem respeito ao agrupamento, onde não há diferença no envio de um servidor ou outro.

Com isso, o número de estados desses sistemas dependerá do número de agrupamentos e do número de servidores por agrupamento. Para tanto, apresenta-se a Equação (4), onde M é o

número de estados do sistema, m é o número de agrupamentos, n_k é o número de servidores do agrupamento k e Q o número de estados das filas.

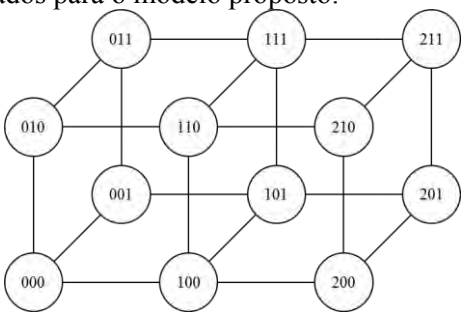
$$M = \prod_{k=1}^{m}(n_k + 1) \quad (4)$$

No exemplo dado, não há diferenças entre os VSAs, visto que são homogêneos, co-localizados e não há diferença quanto às preferências de despacho. Isso possibilita a criação de um agrupamento dos dois servidores onde a taxa de serviço será igual à soma das taxas individuais. Lembrando que nem sempre será possível realizar agrupamentos, de maneira a ter-se agrupamentos com apenas um servidor. Dessa forma, aplicando a Equação (4), teremos:

$$M = \underset{VSAs}{(2+1)} \cdot \underset{VSB1}{(1+1)} \cdot \underset{VSB2}{(1+1)} = 12\ estados$$

O espaço de estados do sistema sofre alterações, criando-se uma dimensão para cada agrupamento, diferentemente do modelo hipercubo original onde cada novo servidor origina uma nova dimensão. No exemplo, teríamos um espaço com 4 dimensões, já que possuímos 4 servidores, porém com o agrupamento teremos apenas 3. A Figura 3 mostra como fica o espaço para esse caso.

Figura 3 – Espaço de estados para o modelo proposto.



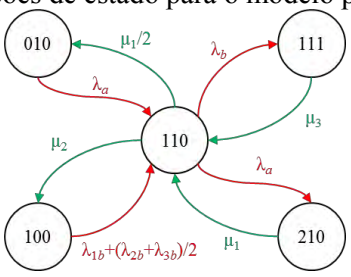
A matriz de preferência de despacho tem as mesmas características apresentadas na adaptação do modelo de Chiyoshi et al. [2011], porém ao invés de trabalhar com os servidores individualmente, utilizam-se os agrupamentos, conforme a Tabela 8 a seguir.

Tabela 8 – Matriz de preferência de despacho para o modelo proposto.

Átomos	**Agrupamentos**		
	1	**2**	**3**
1 *a*	1º	2º	3º
1 *b*	3º	1º	2º
2 *a*	1º	2º	2º
2 *b*	2º	1º	1º
3 *a*	1º	2º	2º
3 *b*	2º	1º	1º
4 *a*	1º	3º	2º
4 *b*	3º	2º	1º

Com o mostrado, é possível separar um estado e observar as suas transições para os outros. A Figura 4 mostra as possíveis transições para o estado {110}.

Figura 4 – Exemplo de transições de estado para o modelo proposto.



Agora é possível apresentar a matriz dos coeficientes para o modelo (Tabela 9). Note que agora tem-se apenas 12 estados.

Tabela 9 – Matriz dos coeficientes para o modelo proposto.

Estados	000	100	010	001	200	110	101	011	210	201	111	211
000	-(λ)	μ1/2	μ3	μ4	0	0	0	0	0	0	0	0
100	λa	-(λ+μ1/2)	0	0	μ1	μ3	μ4	0	0	0	0	0
010	λ1b+(λ2b+λ3b)/2	0	-(λ+μ3)	0	0	μ1/2	0	μ4	0	0	0	0
001	(λ2b+λ3b)/2+λ4b	0	0	-(λ+μ4)	0	0	μ1/2	μ3	0	0	0	0
200	0	λa	0	0	-(λ+μ1)	0	0	0	μ3	μ4	0	0
110	0	λ1b+(λ2b+λ3b)/2	λa	0	0	-(λ+μ1/2+μ3)	0	0	μ1	0	μ4	0
101	0	(λ2b+λ3b)/2+λ4b	0	λa	0	0	-(λ+μ1/2+μ4)	0	0	μ1	μ3	0
011	0	0	λb	λb	0	0	0	-(λ+μ3+μ4)	0	0	μ1/2	0
210	0	0	0	0	λ1+(λ2+λ3)/2	λa	0	0	-(λ+μ1+μ3)	0	0	μ4
201	0	0	0	0	(λ2+λ3)/2+λ4	0	λa	0	0	-(λ+μ1+μ4)	0	μ3
111	0	0	0	0	0	λb	λb	λ	0	0	-(λ+μ1/2+μ3+μ4)	μ1
211	0	0	0	0	0	0	0	0	λ	λ	λ	-μ

Com isso pode-se calcular as probabilidades de cada estado para o modelo, mas para o cálculo da carga de trabalho de cada servidor é preciso utilizar a Equação (5) abaixo. Onde, B é um estado pertencente ao conjunto de estados M do sistema, n_{kB} é o número de servidores do agrupamento k que se encontram ocupados no estado B, n_k é o número de servidores pertencentes ao agrupamento k, P_B é a probabilidade do estado B.

$$\rho_{ki} = \sum_{B \epsilon M} \frac{n_{kB} \cdot P_B}{n_k} \tag{5}$$

Dessa forma, as cargas de trabalho dos servidores do exemplo estão presentes na Tabela 10 a seguir. Note que os servidores 1 e 2, que são pertencentes ao agrupamento 1 possuem a mesma carga de trabalho, já que são homogêneos e não há diferença na preferência de envio entre eles, assim como visto para o modelo adaptado de Chiyoshi et al. [2011].

Tabela 10 – Cargas de trabalho para o modelo proposto.

Servidor	Workload
1	0,3026
2	0,3026
3	0,4006
4	0,3995

5. Comparação dos resultados

Também é importante mostrar as diferenças nos resultados de cada desses modelos que buscam representar aleatoriedade dentro do modelo hipercubo. A Tabela 11 traz um resumo dos resultados obtidos. Note que há uma redução significativa do número de equações necessárias para cada modelo, lembrando que quanto maior o sistema a ser analisado, maiores são as oportunidades de ganho com essa redução. Além disso, nota-se que não houve perda na precisão dos resultados em cada modelo, sendo que os dois com representação formal de aleatoriedade não tiveram diferença nenhuma nas cargas de trabalho observadas.

Tabela 11 – Resumo dos resultados apresentados.

Método	Nº de eq.	Workload			
		Serv. 1	Serv. 2	Serv. 3	Serv. 4
Takeda et al. [2007]	80	0,3170	0,2864	0,3841	0,4157
Chiyoshi et al. [2011] adaptado	16	0,3026	0,3026	0,4006	0,3995
Proposta	12	0,3026	0,3026	0,4006	0,3995

6 Conclusão

Neste artigo foram apresentadas algumas formas de incluir a aleatoriedade do despacho de servidores utilizando o modelo hipercubo de filas. Além das formas já encontradas na literatura, apresentou-se uma nova extensão utilizando agregação de servidores como forma de diminuição do custo computacional.

Os modelos mostram uma evolução em relação ao modelo original ao incluírem uma característica comum a SAEs urbanos. Também mostram-se evoluções entre si com o ganho em precisão e quantidade de equações resolvidas.

O modelo proposto consegue atender a uma demanda apresentada em trabalhos anteriores como Larson e Odoni [2007], Chiyoshi et al. [2011].

Para pesquisas futuras visa-se aplicar o modelo proposto em sistemas reais juntamente de outras extensões do modelo hipercubo como backup parcial e prioridade em fila. Além disso, propõe-se a inclusão do modelo em métodos de otimização para modelos de localização de servidores.

Os autores agradecem à FAPESP pelo apoio financeiro.

Referências

Boyaci B. e Geroliminis N. (2015). Approximation methods for large-scale spatial queueing systems. *Transportation Research Part B* 74:151-181.

Chiyoshi F., Galvão R. D. e Morabito R. (2000). O uso do modelo hipercubo na solução de problemas de localização probabilísticos. *Gestão & Produção* 7(2):146-174.

Chiyoshi F., Iannoni A. P., Morabito R. (2011) A tutorial on hypercube queueing models and some practical applications in emergency service systems. *Pesquisa Operacional* 31(2):271-299.

Larson R. C. (1974). Hypercube queuing model for facility location and redistricting in urban emergency services. *Computers and operations research* 1:67-95.

Larson R. C. e Odoni A. R. (2007). Urban Operations Research. 2 ed. Dynamic Ideas, Belmont, Massachusett.

Luque L. (2006). Análise da aglutinação de estados em cadeias de markov do modelo hipercubo de filas com servidores co-localizados. Dissertação de Mestrado. INPE – São José dos Campos.
Simpson N. C. e Hancock P. G. (2009). Fifty years of operational research and emergency response. *Journal of the Operational Research Society* 60:126-139.
Souza R., Morabito R., Chiyoshi F. e Iannoni A. P. (2015). Incorporating priorities for waiting customers in the hypercube queuing model with application to an emergency medical service system in Brazil. *European Journal of Operational Research* 242:274-285.
Takeda R. (2000). Uma contribuição para avaliar o desempenho de sistemas de transporte emergencial de saúde. Universidade de São Paulo. *Tese* (doutorado em Transportes) – Escola de Engenharia de São Carlos.
Takeda R. A., Widmer, J. A. e Morabito, R. (2004). Aplicação do modelo hipercubo de filas para avaliar a descentralização de ambulâncias em um sistema urbano de atendimento médico de urgência. *Pesquisa Operacional* 24(1):39-72.
Takeda R. A., Widmer, J. A. e Morabito, R. (2007). Analysis of ambulance decentralization in an urban emergency medical service using the hypercube queueing model. *Computers &Operations Research* 34:727-741.