

Análise de Dados do Simulador MobFogSim

V. D. M. Pinho

L. F. Bittencourt

Relatório Técnico - IC-PFG-20-23

Projeto Final de Graduação

2020 - Dezembro

UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE COMPUTAÇÃO

The contents of this report are the sole responsibility of the authors.
O conteúdo deste relatório é de única responsabilidade dos autores.

Análise de Dados do Simulador MobFogSim

Vitoria Dias Moreira Pinho*

Luiz Fernando Bittencourt†

Resumo

Com a expansão e popularização do conceito de Internet das Coisas, a abordagem de *Fog Computing* passa a entrar em destaque como uma solução tecnológica para atender essa nova demanda. Entretanto, a mobilidade de dispositivos gera dificuldade para o desenvolvimento desta tecnologia, principalmente relacionada às decisões e políticas de migração. Diante deste cenário, a utilização do MobFogSim, um simulador de mobilidade em *Fog Computing*, pode ser de extremo auxílio para escolha da configuração de migração. O objetivo deste trabalho é explorar o uso deste simulador, com diferentes parâmetros de entrada, para avaliar como estes influenciam nos resultados da simulação.

1 Introdução

Nas últimas décadas, com o desenvolvimento de novas tecnologias e aprimoramento das redes de computadores, a internet tem se tornado um mecanismo rápido e poderoso de comunicação entre uma grande variedade de dispositivos ao redor do mundo. Assim, em 1999, surgiu o termo Internet das Coisas (IoT - do inglês *Internet of Things*), proposto por Kevin Ashton [1]. IoT, como o próprio nome sugere, consiste em diversos dispositivos diferentes, como eletrodomésticos, celulares, computadores, meios de transporte, entre outros, se comunicando entre si através da internet.

O crescente advento da IoT gera um grande volume de dados e também uma alta demanda por processamento. No modelo de *Cloud Computing* (Computação em Nuvem) [2], o processamento é realizado utilizando recursos de uma máquina ociosa (nuvem - do inglês *cloud*), conectada ao usuário através da rede. Apesar do modelo de computação em nuvem ser largamente utilizado atualmente, há muitas dificuldades em utilizá-lo para IoT pois a demanda por processamento sobrecarregaria o tráfego na rede. Nesse contexto surge a ideia de *Fog Computing* (Computação em Névoa) [3], que estende a ideia de computação em nuvem acrescentando pontos de processamento (névoa - do inglês *fog*) para a borda da rede. Ou seja, é adicionada uma camada de processamento mais próxima do dispositivo, de forma que o tráfego da rede é significativamente reduzido. A figura 1 ilustra a diferença entre *Cloud Computing* e *Fog Computing* através de dois diagramas que mostram como os dispositivos se conectam na rede, ilustrando os processos descritos anteriormente.

Uma das dificuldades enfrentadas ao implementar o modelo de *Fog Computing* é a mobilidade dos dispositivos. Já que, conforme o dispositivo se locomove pode ser necessário

*Instituto de Computação, Universidade Estadual de Campinas, 13081-970 Campinas, SP.

†Instituto de Computação, Universidade Estadual de Campinas, 13081-970 Campinas, SP.

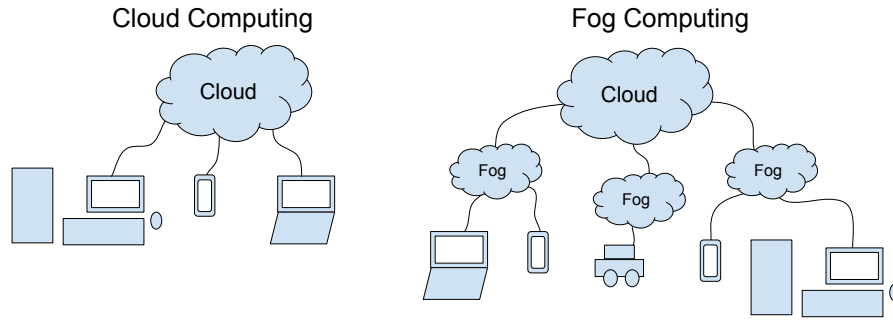


Figura 1: *Cloud Computing* ilustrado à esquerda e *Fog Computing* ilustrado à direita. Os dispositivos conectados à rede podem ser celulares, computadores pessoais, meios de transporte, eletrodomésticos etc.

migrar os dados do próprio para outro ponto de processamento, pois quanto mais longe o dispositivo, maior o tráfego na rede. Esses pontos de processamento, denominados *cloudlets*, podem executar as tarefas utilizando máquinas virtuais (VMs, do inglês *Virtual Machine*) ou *containers*.

Nesse contexto, foi desenvolvido o simulador de mobilidade MobFogSim[4], sendo uma extensão do simulador iFogSim[5] com o acréscimo da modelagem da mobilidade de usuário e da migração de *containers* e máquinas virtuais ao longo das *cloudlets*. Esse simulador permite variar diversos parâmetros de mobilidade, como a política de migração, a estratégia de migração ou a abordagem para escolha do ponto de migração e apresenta uma grande variedade de resultados que podem ser utilizados para avaliar qual a melhor combinação de parâmetros para modelagem estudada.

O objetivo deste projeto foi avaliar como a variação dos parâmetros de entrada do simulador MobFogSim altera os resultados de mobilidade obtidos. Este trabalho foi dividido de forma que a seção 2 apresenta alguns conceitos básicos sobre *Fog Computing* e sobre o processo de migração de dados num contexto de mobilidade do usuário, a seção 3 discute como os dados foram coletados e organizados, já a seção 4 mostra os resultados obtidos e, por fim, na seção 5 é apresentada as conclusões do estudo.

2 Conceitos Básicos

Essa seção tem como objetivo apresentar conceitos básicos de *Fog Computing* 2.1 e do processo de Migração 2.2.

2.1 *Fog Computing*

Como descrito anteriormente, o modelo de *Fog Computing* apresenta pontos de processamento na borda da rede, denominados *cloudlets*. Essas *cloudlets* podem se conectar a outras *cloudlets* ou diretamente com a nuvem (*clouds*), sendo que a infraestrutura e organização

da rede depende do tipo de aplicação. As *cloudlets* são capazes de armazenar ou processar dados e devem estar próximas do dispositivo, para garantir um processamento rápido. Dessa forma, quando o usuário se locomove é esperado que haja um processo de migração dos seus dados para uma outra *cloudlet*, mais próxima deste. No simulador MobFogSim é considerado que esse processo de migração de dados e de processamento é realizado utilizando VMs e/ou *containers*, que encapsulam o acesso ao computador e aos recursos de rede. Ou seja, os nós *fog* da rede tomam a decisão de quando realizar migrações de uma *cloudlet* para outra.

2.2 Migração

A decisão de realizar ou não a migração e como vai ser realizado esse processo depende de diversos fatores tais como a posição das *cloudlets*, a conexão de rede, a direção e velocidade do usuário, entre outros.

No momento de realizar uma migração diferentes estratégias podem ser adotadas, como uma migração proativa, em que a migração é iniciada antes que o processo de *handoff* comece, baseando-se em informações da mobilidade do usuário para prever um bom ponto de migração, ou, alternativamente, pode-se adotar a estratégia de migração reativa, em que a migração só é iniciada uma vez que o processo de *handoff* já foi iniciado, sendo que o novo ponto de acesso do usuário é descoberto de forma determinística.

A decisão de para onde migrar pode ser baseada na distância geográfica, na topologia da rede, ou em outros critérios. Ainda, se não tem *cloudlets* disponíveis, é possível que não haja uma migração. Três diferentes critérios podem ser utilizados no simulador MobFogSim para escolher uma próxima *cloudlet* para realizar uma migração: menor distância entre o usuário e o ponto de acesso (ou seja, escolhe a *cloudlet* conectada ao ponto de acesso mais próximo ao usuário), menor distância entre o usuário e a nova *cloudlet* (ou seja, escolhe a *cloudlet* mais próxima ao usuário), menor latência (escolhe a *cloudlet* com a menor latência para o usuário).

Por fim, a preparação para realização da migração envolve checar o tamanho da VM, estabelecer uma conexão entre a *cloudlet* de destino e a *cloudlet* fonte, e depende da técnica de migração utilizada. As políticas de migração utilizadas pelo MobFogSim são a *post-copy migration* e a *cold migration*. O processo de *cold migration* inicialmente interrompe a VM ou *container*, posteriormente a transfere, e por fim continua o processamento da própria na *cloudlet* de destino. Essa técnica apresenta um alto *downtime* (tempo que serviço está indisponível por estar em migração) e pode coincidir com o tempo total de migração. Já o processo de *post-copy migration* realiza a migração enquanto a VM ou *container* ainda está rodando, sendo que esta só é interrompida por um tempo mínimo, apenas para salvar e transferir uma pequena porção do estado geral.

3 Metodologia

O objetivo do estudo foi verificar como a variação dos parâmetros de entrada do simulador MobFogSim afetava os resultados da simulação. Para isso foi utilizado o aplicativo de

exemplo, fornecido no repositório¹ do projeto.

O repositório do projeto fornece diversos arquivos de entrada, disponibilizados no diretório inputs. A entrada utilizada foi a do arquivo 1702log.csv, representada na figura 2. Os pontos azuis representam a posição das *cloudlets* ao longo do trajeto, que sempre têm o mesmo valor padrão, definido pelo simulador, enquanto os pontos vermelhos representam o caminho realizado pelo dispositivo móvel, definido no arquivo de entrada.

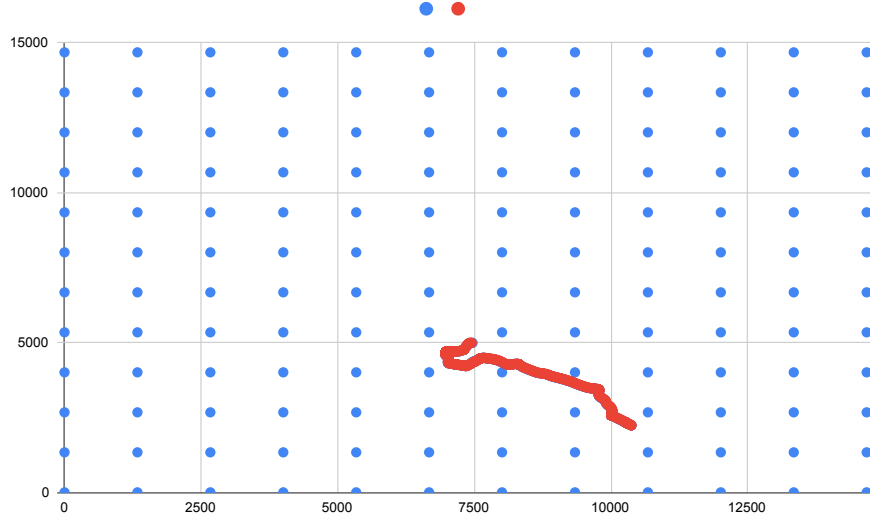


Figura 2: Trajeto realizado pelo dispositivo móvel na entrada 1702.

O simulador MobFogSim permite a variação de 10 parâmetros de entrada, descritos na tabela 1, que mostra os possíveis valores assumidos para cada um deles. No projeto, os parâmetros do número de usuários, semente para geração de números aleatórios, predição da mobilidade do usuário e imprecisão da predição de mobilidade do usuário não sofreram variações, sendo fixados com os valores da tabela 2.

Para os demais parâmetros foram considerados dois cenários como base, que se diferenciam pela abordagem do ponto de migração utilizada, denominados Cenário 1 (Subseção 3.1) e Cenário 2 (Subseção 3.2). Para cada cenário foram variados outros parâmetros de simulação para avaliar como estes influenciavam nos resultados da simulação.

Foram realizadas 3 medidas para cada configuração de entrada utilizada e foi calculada a média aritmética dos resultados obtidos. A tabela 3 mostra a descrição dos resultados avaliados no projeto.

3.1 Cenário 1

As variações dos parâmetros de entrada utilizados neste cenário encontram-se na tabela 4. Os dados em negrito indicam os parâmetros estudados para a respectiva medida.

¹MobFogSim encontra-se disponível em <https://github.com/diogomg/MobFogSim>

Parâmetro	Valores Assumidos
Processo de Migração	Permitido ou Negado
Semente usada na geração de números aleatórios	Número inteiro
Abordagem do ponto de migração	Fixa ou Baseada na velocidade do usuário
Estratégia de migração para selecionar Cloudlet de destino	Menor latência ou Menor distância entre o usuário e a Cloudlet ou Menor distância entre o usuário e o ponto de acesso
Número de usuários	Número inteiro
Largura de banda entre Cloudlets	Número inteiro
Política de Migração	Complete VM/Cold migration ou Complete Container migration ou Container Live Migration
Predição da mobilidade do usuário (em segundos)	Número inteiro
Imprecisão da predição de mobilidade do usuário (metros)	Número inteiro
Latência da rede entre Cloudlets	Número inteiro

Tabela 1: Descrição dos parâmetros disponíveis no simulador MobFogSim

Parâmetro	Valor Assumido
Semente usada na geração de números aleatórios	290538
Número de usuários	1
Predição da mobilidade do usuário (segundos)	0
Imprecisão da predição de mobilidade do usuário (metros)	0

Tabela 2: Valores fixados utilizados no estudo.

Resultados da Simulação	Descrição
Average delay after new connection	Média do atraso depois de uma nova conexão
Average Downtime	Média do tempo em que a máquina está indisponível (migração)
Energy history device	Energia consumida pelo dispositivo
Average migration time	Média do tempo total de migração
Cloudlet network usage	Uso da rede pela <i>cloudlet</i>
Device network usage	Uso da rede pelo dispositivo
Total network usage	Uso da rede total
VM size sended	Tamanho da VM utilizado na migração
Total handoff	Número total de <i>handoff</i> realizados
Total my countless latency	Total de latências que não foram contabilizadas

Tabela 3: Descrição dos resultados obtidos no simulador.

3.2 Cenário 2

As variações dos parâmetros de entrada utilizados neste cenário encontram-se na tabela 5. A intenção do cenário 2 foi estudar se a variação dos parâmetros seguia os padrões do cenário 1 ao se fixar a abordagem do ponto de migração baseada na velocidade do usuário. Os dados em negrito indicam os parâmetros estudados para a respectiva medida.

4 Resultados

Como explicado anteriormente, foram coletados 3 instâncias de dados para cada medida e calculada a média aritmética de três resultados para cada categoria de dados obtida. Por exemplo, para medida 1 da tabela 4, executou-se a simulação três vezes e coletou-se o dado Device network usage realizando a média dos valores obtidos referentes a este dado. Devido ao fato do caminho realizado pelo usuário ser o mesmo para todas as medidas, alguns dados tais como o valor do clock da aplicação e do total de handoffs foram constantes, sendo, neste caso, 704000 e 5 respectivamente. A fim de facilitar a análise os dados foram separados nas subseções 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, e 4.9 com suas respectivas análises. Os dados referentes a medida 1 e 5 serão discutidos posteriormente, na subseção de considerações finais 4.10.

4.1 Average delay after new connection

Os dados obtidos para Average delay after new connection para o Cenário 1 para o Cenário 2 encontram-se na tabela 6.

Do cenário 1 é possível perceber que a maior parte dos valores ficaram na faixa de 382000, portanto a análise é focada nas medidas que não acompanharam essa tendência. Já no cenário 2 a maior parte das medidas ficou na faixa de 336000. A medida 2 obteve um resultado menor (em torno de 336000) devido a abordagem do ponto de migração ser baseado

Medida	Processo de Migração	Abordagem do Ponto de Migração	Estratégia de Migração	Largura de banda entre cloudlets	Política de Migração	Latência de rede entre cloudlets
base	permitido	fixo	menor latência	11	Complete VM/Cold migration	61
1	negado	fixo	menor latência	11	Complete VM/Cold migration	61
2	permitido	baseado na velocidade do usuário	menor latência	11	Complete VM/Cold migration	61
3	permitido	fixo	menor distância entre usuário e cloudlet	11	Complete VM/Cold migration	61
4	permitido	fixo	menor distância entre usuário e ponto de acesso	11	Complete VM/Cold migration	61
5	permitido	fixo	menor latência	7	Complete VM/Cold migration	61
6	permitido	fixo	menor latência	11	Complete Container migration	61
7	permitido	fixo	menor latência	11	Container Live Migration	61
8	permitido	fixo	menor latência	11	Complete VM/Cold migration	50

Tabela 4: Parâmetros de entrada utilizados no Cenário 1.

Medida	Processo de Migração	Abordagem do Ponto de Migração	Estratégia de Migração	Largura de banda entre cloudlets	Política de Migração	Latência de rede entre cloudlets
base	permitido	baseado na velocidade do usuário	menor latência	11	Complete VM/Cold migration	61
9	permitido	baseado na velocidade do usuário	menor distância entre usuário e cloudlet	11	Complete VM/Cold migration	61
10	permitido	baseado na velocidade do usuário	menor distância entre usuário e ponto de acesso	11	Complete VM/Cold migration	61
11	permitido	baseado na velocidade do usuário	menor latência	100	Complete VM/Cold migration	61
12	permitido	baseado na velocidade do usuário	menor latência	11	Complete Container migration	61
13	permitido	baseado na velocidade do usuário	menor latência	11	Container Live Migration	61
14	permitido	baseado na velocidade do usuário	menor latência	11	Complete VM/Cold migration	15

Tabela 5: Parâmetros de entrada utilizados no Cenário 2.

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	382147,27	base	336277,52
1	0,00	9	336277,69
2	336277,52	10	336277,52
3	382147,09	11	423694,55
4	382147,27	12	348659,93
5	0,00	13	454357,89
6	386854,70	14	336231,52
7	354341,84	-	
8	382141,77	-	

Tabela 6: Average delay after new connection para o Cenário 1 e Cenário 2.

na velocidade do usuário, fato que pode ser confirmado no cenário 2, em que os valores seguiram essa mesma tendência. A política de migração Container Live Migration pareceu reduzir essa média no cenário 1 (medida 7), porém aumentou no cenário 2 (medida 13). Isto pode indicar que essa política apenas diminuí o valor quando acompanhada de abordagem de ponto de migração fixo, entretanto seriam necessários mais testes para confirmar isso. Já a política de migração Complete Container Migration, das medidas 6 e 12, aumentou o valor da média em ambos os casos. Por fim, um aumento da largura de banda entre as *cloudlets* gerou um aumento na média.

Em suma, os resultados indicam que a abordagem do ponto de migração baseada na velocidade do usuário diminui o atraso depois de uma nova conexão. Já a política de migração CompleteVM/Coldmigration parece ser a melhor escolha, para um menor atraso, entretanto seria necessário uma maior amostra de dados para confirmar essa tendência.

4.2 Average downtime

Os dados obtidos para Average downtime para o Cenário 1 e para o Cenário 2 encontram-se na tabela 7.

Do cenário 1 e do cenário 2 é possível perceber que a maior parte dos valores ficaram na faixa de 102000, portanto a abordagem do ponto de migração parece não ter influência sobre o Average downtime. A medida 11 teve uma grande diminuição deste parâmetro, o que é condizente já que, quanto maior a largura de banda entre as *cloudlets*, mais rápida é a transferência de dados, e, portanto, menor o Average downtime. As políticas de migração Container Live Migration (das medidas 7 e 13) e Complete Container Migration (das medidas 6 e 12) parecem diminuir esse parâmetro também, o que é condizente já que a política do Coldmigration é a que deveria apresentar maior Average downtime.

Em suma, os resultados indicam que uma maior largura de banda entre as *cloudlets* e as política de migração Complete Container Migration e Container Live Migration apresentam resultados melhores do que a política CompleteVM/Coldmigration.

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	102269,71	base	102269,34
1	0,00	9	102269,34
2	102269,34	10	102269,34
3	102269,71	11	10348,00
4	102269,71	12	59853,38
5	0,00	13	19942,26
6	60550,39	14	102269,34
7	19942,30	-	
8	102269,71	-	

Tabela 7: Average downtime para o Cenário 1 e para o Cenário 2.

4.3 Energy history device

Os dados obtidos para Energy history device para o Cenário 1 e para o Cenário 2 encontram-se e na tabela 8.

Do cenário 1 é possível perceber que a maior parte dos valores ficaram na faixa de 60800000, enquanto no cenário 2 a maior parte das medidas ficou na faixa de 60900000. Tal resultado indica que a abordagem do ponto de migração ser baseado na velocidade do usuário leva a um pequeno aumento no consumo de energia (também evidenciado na medida 2). A política de migração Container Live Migration (das medidas 7 e 13) parece provocar uma redução no consumo de energia enquanto a política de migração Complete Container Migration (das medidas 6 e 12) parece aumentar esse parâmetro.

Em suma, os resultados indicam que a abordagem do ponto de migração baseada na velocidade do usuário leva a um leve aumento no consumo de energia. Já a política de migração Container Live Migration parece ser a melhor alternativa, para redução desse parâmetro.

4.4 Average migration time

Os dados obtidos para Average migration time para o Cenário 1 e para o Cenário 2 encontram-se na tabela 9.

Do cenário 1 e do cenário 2 é possível perceber que a maior parte das medidas ficaram na faixa de 102269, indicando que a abordagem do ponto de migração não tem influência sobre esse parâmetro. A política de migração Container Live Migration (das medidas 7 e 13) parece aumentar essa medida, enquanto a política de migração Complete Container Migration (das medidas 6 e 12) tem o efeito de redução na própria. Por fim, o aumento da largura de banda entre *cloudlets* da medida 11 apresentou uma redução ainda maior da média do tempo de migração, o que é condizente, já que quanto maior a banda, menor o tempo que leva para migração ocorrer.

Em suma, os resultados indicam que a largura da banda entre *cloudlets* é o parâmetro

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	60852489,30	base	60927591,54
1	60269690,35	9	60930187,97
2	60927591,54	10	60927113,80
3	60851312,46	11	60645728,65
4	60853094,63	12	59351762,12
5	63278982,57	13	62141440,05
6	59469695,08	14	60919824,90
7	61014933,08	-	
8	60896628,44	-	

Tabela 8: Energy history device para o Cenário 1 e para o Cenário 2.

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	102269,71	base	102269,34
1	0,00	9	102269,46
2	102269,34	10	102269,34
3	102269,71	11	10348,00
4	102269,71	12	61363,90
5	0,00	13	132948,42
6	61364,27	14	102269,34
7	132948,66	-	
8	68254,53	-	

Tabela 9: Average migration time para o Cenário 1 e para o Cenário 2.

de maior influência na redução do tempo de migração. Já a política de migração Complete Container Migration parece ser a melhor escolha para redução dessa medida.

4.5 Cloudlet network usage

Os dados obtidos para Cloudlet network usage para o Cenário 1 e para o Cenário 2 encontram-se na tabela 10.

Do cenário 1 é possível perceber que a maior parte dos valores ficaram na faixa de 1.343.967.574, enquanto no cenário 2 a maior parte das medidas ficou na faixa de 2.733.612.902, indicando que a abordagem do ponto de migração ser baseado na velocidade do usuário leva a um aumento do uso de rede da *cloudlet*. A política de migração Container Live Migration (das medidas 7 e 13) e a política Complete Container Migration (das medidas 6 e 12) parecem ter um efeito no aumento do uso da rede pela *cloudlet* também, entretanto no cenário 2 a política Container Live Migration não parece ter influenciado esse parâmetro, sendo

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	1.343.967.574,36	base	2.733.612.902,73
1	0,00	9	2.717.266.072,75
2	2.733.612.902,73	10	2.733.612.902,73
3	1.343.967.574,36	11	26.624.368.486,73
4	1.343.967.574,36	12	6.939.483.995,20
5	807.096.662,36	13	2.733.612.902,73
6	4.088.605.964,77	14	2.733.612.902,73
7	2.733.612.902,73	-	
8	1.343.967.574,36	-	

Tabela 10: Cloudlet network usage para o Cenário 1 e para o Cenário 2.

necessários mais testes. O aumento da largura de banda entre as *cloudlets*, representado pela medida 11, proporcionou um grande aumento nesse parâmetro, o que é condizente, já que com uma maior largura de banda é esperado que a utilização da rede seja maior.

Os parâmetros que parecem ter mais influência sobre essa medida são a abordagem de migração, a política de migração e a largura de banda entre as *cloudlets*. Dessa forma, visando economia de uso de rede, as melhores escolhas seriam manter a abordagem de migração fixa, escolher uma largura de banda entre as *cloudlets* relativamente baixa e uma política de migração CompleteVM/Coldmigration.

4.6 Device network usage

Os dados obtidos para Device network usage para o Cenário 1 e para o Cenário 2 encontram-se na tabela 11.

Do cenário 1 é possível perceber que a maior parte dos valores ficaram na faixa de 10.000.000, já no cenário 2 a maior parte das medidas ficou na faixa de 6.200.000, evidenciando que a abordagem do ponto de migração ser baseado na velocidade do usuário leva a uma redução do uso da rede no dispositivo. As políticas de migração Container Live Migration (das medidas 7 e 13) e Complete Container Migration (das medidas 6 e 12) levaram a um aumento nesse parâmetro também, sendo a última responsável por um aumento mais proeminente. O aumento da largura de banda entre as *cloudlets*, da medida 11, também proporcionou um aumento neste parâmetro.

Dessa forma, os resultados indicam que a abordagem do ponto de migração baseada na velocidade do usuário é a melhor escolha para uma redução deste parâmetro. Ainda, a política de migração CompleteVM/Coldmigration parece também apresenta significativa contribuição em sua redução.

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	10.364.202,00	base	6.223.774,00
1	29.177.604,00	9	6.216.158,00
2	6.223.774,00	10	6.226.718,00
3	10.352.704,00	11	11.001.914,00
4	10.346.076,00	12	7.849.550,00
5	9.197.348,00	13	25.718.446,00
6	12.650.258,00	14	6.208.436,00
7	27.169.116,00	-	
8	10.363.216,00	-	

Tabela 11: Device network usage para o Cenário 1 e para o Cenário 2.

4.7 Total network usage

Os dados obtidos para Total network usage para o Cenário 1 e para o Cenário 2 encontram-se na tabela 12.

Do cenário 1 é possível perceber que a maior parte dos valores ficaram na faixa de 1.354.000.000, enquanto que no cenário 2 a maior parte das medidas ficou na faixa de 2.739.000.000. Indicando que a abordagem do ponto de migração ser baseado na velocidade do usuário provoca um aumento no uso total da rede. A política de migração Container Live Migration (medidas 7 e 13) pareceu aumentar essa medida apenas no caso em que a abordagem de ponto de migração é fixa. Já a política de migração Complete Container Migration (medidas 6 e 12), aumentou esse valor em ambos os casos. Por fim, o aumento da largura de banda entre as *cloudlets* (medida 11) também proporcionou um aumento no uso total da rede.

Logo, os resultados indicam que a abordagem do ponto de migração fixa é uma boa abordagem para diminuir o uso total de rede. Além disso, a política de migração CompleteVM/Coldmigration também é uma boa escolha para manter os valores baixos.

4.8 VM size sended

Os dados obtidos para VM size sended para o Cenário 1 e para o Cenário 2 encontram-se na tabela 13.

Do cenário 1 é possível perceber que a maior parte dos valores ficaram na faixa de 128, enquanto no cenário 2 a maior parte das medidas ficou na faixa de 256. Indicando que a abordagem do ponto de migração ser baseado na velocidade do usuário tem influência sobre o tamanho de VM enviado. As políticas de migração Container Live Migration (das medidas 7 e 13) e Complete Container Migration (das medidas 6 e 12) parecem ter um efeito de aumentar esse parâmetro, sendo que o último causa um aumento maior do que o primeiro.

Em suma, os resultados indicam que a abordagem do ponto de migração e a política de

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	1.354.331.776,36	base	2.739.836.676,73
1	29.177.604,00	9	2.723.482.230,75
2	2.739.836.676,73	10	2.739.839.620,73
3	1.354.320.278,36	11	26.635.370.400,73
4	1.354.313.650,36	12	6.947.333.545,20
5	816.294.010,36	13	2.759.331.348,73
6	4.101.256.222,77	14	2.739.821.338,73
7	2.760.782.018,73	-	
8	1.354.330.790,36	-	

Tabela 12: Total network usage para o Cenário 1 e para o Cenário 2.

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	128	base	256
1	0,00	9	256
2	256	10	256
3	128	11	256
4	128	12	640
5	128	13	256
6	364	14	256
7	256	-	
8	128	-	

Tabela 13: VM size sended para o Cenário 1 e para o Cenário 2.

migração são os principais parâmetros para alteração do tamanho da VM enviado.

4.9 Total migrations

Os dados obtidos para Total migrations para o Cenário 1 e para o Cenário 2 encontram-se na tabela 14.

Do cenário 1 e do cenário 2 é possível perceber que a maior parte dos valores de migração foram 1, indicando que a abordagem do ponto de migração não tem influência sobre o número de migrações. A política de migração Complete Container Migration (das medidas 6 e 12) parecem ter um efeito de aumentar esse número de migrações.

Logo, dentre todos os parâmetros, apenas a política de migração Complete Container Migration parece ter influência sobre o número de migrações.

Cenário 1		Cenário 2	
Medida	Resultado	Medida	Resultado
base	1	base	1
1	0	9	1
2	1	10	1
3	1	11	1
4	1	12	4
5	0	13	1
6	3	14	1
7	1	-	
8	1	-	

Tabela 14: Total migrations para o Cenário 1 e para o Cenário 2.

4.10 Considerações finais

As medidas 1 e 5 obtiveram grande parte dos seus resultados zerados. No caso da medida 1, era esperado esse resultado já que o processo de migração não era permitido. Em relação à medida 5 houve apenas uma diminuição da largura de banda entre as *cloudlets*, entretanto não houve nenhuma migração (o que explica os valores zerados). A diminuição do número de migrações pode ter ocorrido pois, com a diminuição da largura de banda, o tempo de migração ficou muito alto, levando o algoritmo escolher como mais vantajoso não realizar nenhuma migração.

Por fim, é possível constatar que, dependendo do resultado que é almejado, a escolha dos parâmetros da simulação é de suma importância para alcançar o próprio. Ainda, mesmo com uma escolha otimizada de parâmetros, é extremamente difícil conseguir uma solução ótima para todos os resultados. Como exemplo, a medida Average downtime obtém resultados melhores ao não utilizar a política de ColdMigration, entretanto essa mesma política é a que apresenta um melhor resultado para o uso total da rede.

5 Conclusão

Frente às diversas dificuldades enfrentadas em um cenário de mobilidade num modelo de *Fog Computing*, principalmente relacionadas com as decisões e políticas de migração, é possível concluir que a análise dos resultados, gerados pelo simulador MobFogSim, são extremamente úteis para auxiliar na escolha das configurações de modelagem de um sistema desse tipo.

Os resultados obtidos no estudo permitem extrair algumas possíveis combinações de parâmetros para se obter uma modelagem desejada. Ainda, devido a natureza combinatorial do problema, algoritmos ou mecanismos para encontrar uma combinação de parâmetros são um desafio a ser estudado.

Referências

- [1] Kevin Ashton, *That ‘internet of things’ thing*. RFID journal, 22.7, 97–114 (2009).
- [2] B. Hayes, *Cloud computing*. (2008).
- [3] Flavio Bonomi, *Fog computing and its role in the internet of things*. In Proceedings of the first edition of the MCC workshop on Mobile cloud computing, pp. 13–16 (August 2012).
- [4] Carlo PULIAFITO, *MobFogSim: Simulation of mobility and migration for fog computing*. Simulation Modelling Practice and Theory, v. 101, p. 102062 (2020).
- [5] Harshit Gupta, *iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments*. Software: Practice and Experience, 47.9, 1275–1296 (2017).