



Análise de uso da plataforma Konker para simulação de aplicações IoT

L. S. Ramos *L. F. Bittencourt*

Relatório Técnico - IC-PFG-21-7
Projeto Final de Graduação
2021 - Julho

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 da plataforma Konker para simulação de aplicações IoT

Lucas dos Santos Ramos*

Luiz Fernando Bittencourt†

Resumo

O conceito de Internet das Coisas (IoT) tem inúmeras aplicações na sociedade atual tendo um grande potencial de revolucionar diferentes setores das nossas vidas. Com essa capacidade e previsão de expansão tão grande a garantia de escalabilidade de uma aplicação IoT não é trivial e é de extrema importância. Logo a disponibilidade de simulações para testar a escalabilidade dessas aplicações é de grande valia para o desenvolvimento da área. Diante disso, este trabalho visa analisar a plataforma Konker como uma opção para simulação de aplicações IoT, explorando a possibilidade de testes de escalabilidade com ela e mensurando quais recursos computacionais limitam ou não a escala dessas simulações. Após a implementação de uma aplicação de casa inteligente simples na plataforma Konker e a execução de um experimento de escalabilidade criando várias instâncias dessa aplicação e medindo os recursos computacionais utilizados durante o processo, concluímos que a Konker é uma plataforma viável para simulação de aplicações IoT. Em cenários com simulações de escalabilidade que se aproximem de casos reais de uso, com milhares de usuários, o processamento de CPU foi o recurso limitante e simulações em ambientes de nuvem com essa plataforma seriam uma extensão natural da análise aqui apresentada.

1 Introdução

O avanço da tecnologia nos últimos anos distribuiu no mundo centenas de milhões de dispositivos eletrônicos que já estão dentro das residências, do ambiente de trabalho e do cotidiano de grande parte das pessoas. Muitos desses dispositivos já estão conectados entre si através da Internet como computadores e celulares, sendo que hoje temos cerca de metade da população mundial (em torno de 4 bilhões de usuários) ativos na internet e estima-se que em um futuro próximo quase toda a população mundial estará conectada. Com essa conectividade surge a chamada *Internet of Things* (IoT), ou Internet das Coisas, que se trata da conexão de todos os dispositivos e objetos ao nosso redor para criar dados e possibilidades infinitas de melhorias no gerenciamento da vida nos setores residenciais, comerciais, industriais, e até no gerenciamento mais eficiente de cidades e países [1]

Com todas essas aplicações, a área de IoT tem um grande e rápido crescimento previsto que já está acontecendo. Dado o potencial de conexão de muitos dispositivos, a

*Instituto de Computação, Universidade Estadual de Campinas

†Instituto de Computação, Universidade Estadual de Campinas

escalabilidade e disponibilidade de sistemas IoT são aspectos fundamentais para o seu funcionamento. Para alcançar esses aspectos, a realização de simulações de ambientes de IoT são primordiais, já que isso torna possíveis testes de *stress* sem a necessidade de dispositivos físicos. A necessidade por simuladores dessas aplicações é também reforçada por Puliafito et al. [2], que apresentam um simulador com o objetivo de testar o aspecto de disponibilidade de uma aplicação IoT. Nessa mesma linha, Zeng et al. [3] apontam novamente para a importância da simulação de aplicações IoT apresentando o simulador IoTSim, que visa testar o aspecto de conectividade e escalabilidade de uma aplicação IoT. Nesse trabalho temos o objetivo de analisar a plataforma Konker como uma alternativa para simulação de aplicações IoT, apresentando testes de escalabilidade de uma aplicação funcionando nessa plataforma, entendendo como recursos computacionais podem limitar a escalabilidade dos cenários avaliados.

Este relatório começa explicando brevemente a plataforma utilizada e os conceitos importantes que envolvem esse trabalho na Seção 2. Em seguida, na Seção 3 é apresentada uma descrição da aplicação implementada juntamente com um detalhamento dos procedimentos para a realização do experimento de escalabilidade. Então, são apresentados os resultados do experimento realizado e uma discussão sobre eles na Seção 4. Por fim, na Seção 5 finalizamos o relatório com uma discussão sobre as análises realizadas.

2 Conceitos

Em primeiro lugar é importante introduzir brevemente a plataforma Konker utilizada neste relatório.

2.1 Konker

A Konker é uma plataforma para desenvolvimento de aplicações IoT genérica com objetivo de facilitar e agilizar a criação dessas aplicações, sendo uma plataforma *open-source* que se encontra disponível em <https://github.com/KonkerLabs/konker-platform>. A Konker apresenta alguns conceitos básicos que servem de base para o entendimento do seu funcionamento e são essenciais para estruturar uma aplicação nessa plataforma. Durante este relatório, os principais conceitos da Konker que serão utilizados são os conceitos de mensagem, dispositivo, canal e rota. Cada um desses conceitos é explicado abaixo.

2.1.1 Mensagens

Uma mensagem é uma informação enviada de um dispositivo físico para sua representação virtual na Konker ou entre os dispositivos virtuais existentes na plataforma. É muito comum essa mensagem ser enviada no padrão JSON.

2.1.2 Dispositivos

Um dispositivo na plataforma da Konker é uma representação virtual de um dispositivo físico, como por exemplo uma lâmpada, um sensor de presença ou um sensor de temperatura.

Um dispositivo possui canais de entrada e canais de saída e um canal é uma via com a qual o dispositivo na Konker pode se comunicar com outros dispositivos. Nos canais de entrada, um dispositivo na Konker pode receber dados externos, como por exemplo receber dados de temperatura do sensor de temperatura físico que ele representa. Já nos canais de saída, o dispositivo pode enviar dados para outros dispositivos físicos ou que existam na plataforma utilizando uma rota. A comunicação entre os dispositivos físicos e os dispositivos virtuais na Konker pode ser feita através do protocolo HTTP com a realização de chamadas REST ou através do protocolo MQTT.

2.1.3 Rota

Neste trabalho uma rota foi utilizada somente como uma via de comunicação entre dois dispositivos que existem na plataforma da Konker. Uma rota necessita que seja especificado o dispositivo e canal de entrada e um dispositivo e canal de saída. A função da rota é realizar o encaminhamento de uma mensagem vinda do dispositivo e canal de entrada para o dispositivo de saída no canal de saída especificado, sendo possível fazer filtros de quais mensagens encaminhar e quais não encaminhar.

Com os conceitos apresentados, a Konker permite a criação de ambientes variados. Por exemplo, é possível criar um dispositivo virtual na Konker que represente um sensor de temperatura físico fazendo com que eles se comuniquem através do protocolo MQTT, então com a adição de outro dispositivo na Konker que represente um atuador, como um motor, fazendo novamente com que eles se comunicando através do protocolo MQTT. Podemos por fim adicionar uma rota que encaminhe uma ação ao motor quando a temperatura atinja um valor específico.

2.1.4 Rota lógica

Apesar de não ser um conceito da Konker, neste relatório também foi utilizado o conceito de Rota lógica. O conceito similar na Konker é chamado de Transformação, entretanto como neste trabalho não foram utilizadas as transformações da Konker, definimos o conceito de rota lógica.

Uma rota lógica neste relatório é uma conexão entre dois canais de quaisquer dispositivos na Konker que ocorre condicionalmente e externamente à plataforma da Konker, ou seja, é quando um dispositivo físico recebe uma mensagem e ele mesmo seguindo uma lógica programada nele envia um dado condicionalmente para um canal de algum dispositivo qualquer na Konker. Esse conceito é usado por simples praticidade das simulações e será abordado com mais detalhes na Seção 3.

3 Metodologia

O objetivo do experimento realizado neste relatório foi de validar a utilização da Konker para simulações de escalabilidade de aplicações IoT bem como entender os recursos computacionais que limitam o aumento de escala dessa simulação. Todos os arquivos utilizados para

realização desse experimento podem ser encontrados em <https://github.com/lucasR23/iot-simulation>

3.1 Aplicação

Para a execução desse experimento foi desenvolvida uma aplicação na plataforma da Konker que exemplifica o funcionamento de uma casa inteligente simples. Essa aplicação implementa um cenário em que a casa possui uma iluminação qualquer, um robô de limpeza, um sensor de presença e o usuário possui um celular em que ele pode controlar o estado da iluminação e do robô de limpeza. A casa possui a funcionalidade de automaticamente desligar a iluminação se o usuário sair da casa e a iluminação estiver ligada, e também de notificar o usuário em seu celular perguntando se ele deseja ligar o robô de limpeza caso ele esteja desligado quando o usuário sair da casa.

Essa aplicação possui a arquitetura apresentada na Figura 1, sendo composta por 4 dispositivos, cada um representando um dispositivo físico do mundo real.

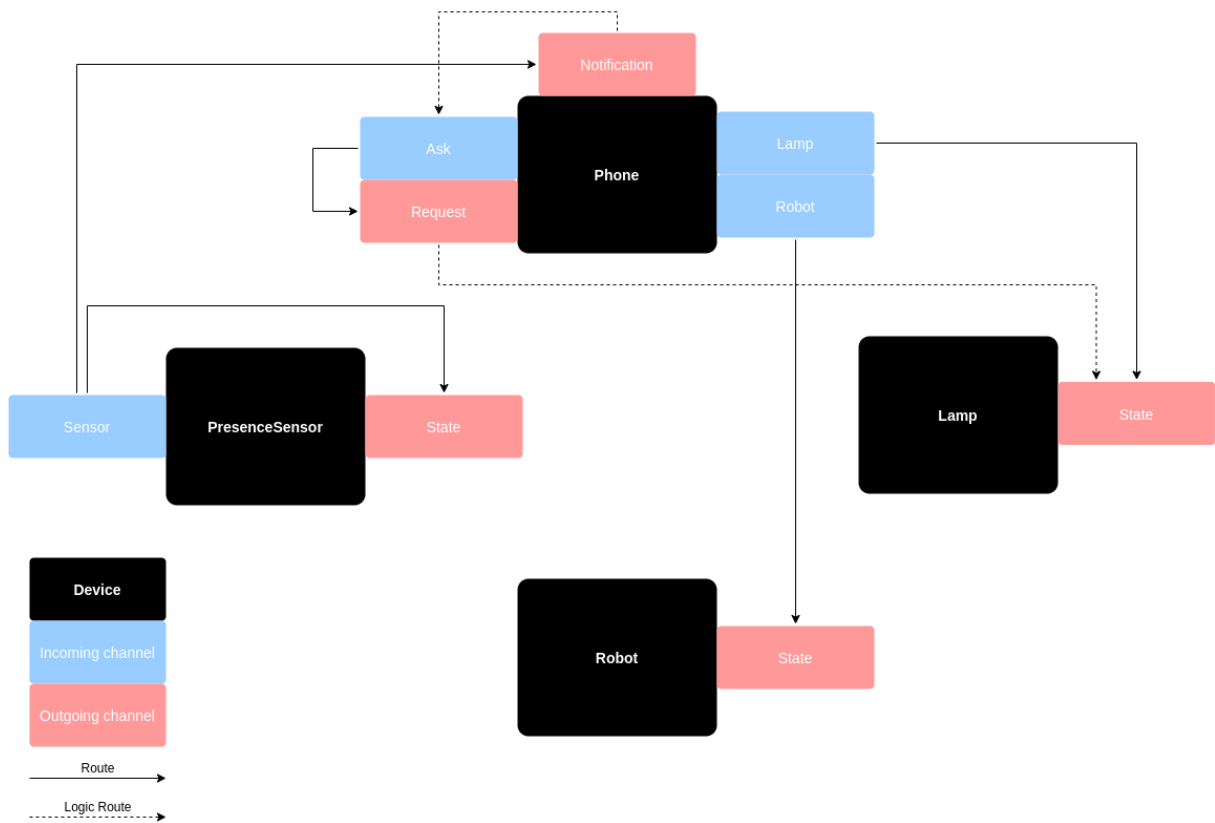


Figura 1: Estrutura da aplicação simulada na plataforma Konker

O dispositivo *phone* possui o canal de entrada *lamp* que recebe os comandos referentes à iluminação casa; o canal de entrada *robot* de forma análoga recebe os comandos referentes ao robô de limpeza. O dispositivo *Lamp* possui somente um canal de saída que envia os

dados para o dispositivo físico de iluminação e , novamente, de forma análoga o dispositivo *Robot* possui somente um canal de saída que envia os dados para o robô de limpeza físico. Existe uma rota entre os canais de entrada *lamp* e *robot* com os canais de saída *state* dos dispositivos *Lamp* e *Robot* respectivamente, essas rotas são as responsáveis por encaminhar a mensagem de alteração de estado de algum desses dispositivos.

O último dispositivo dessa aplicação é o *PresenceSensor*, que representa o sensor de presença físico existente na casa. Ele tem um canal de entrada *sensor* que recebe os dados do sensor de presença real e esses dados são encaminhados para um canal de saída chamado *state*, que representa a presença física do nosso usuário na casa. Outra rota conecta o canal de entrada *sensor* no canal de saída *notification* do dispositivo *Phone*. Essa rota de saída é usada para notificar o usuário caso seja necessário requisitar que ele tome alguma decisão.

Por fim, o dispositivo *Phone* possui uma rota de entrada *ask* que recebe as decisões que usuário tomou e as encaminha através de uma rota para o canal de saída *Request*, que por sua vez encaminha a ação necessária a ser tomada aos devidos dispositivos.

As rotas lógicas que aparecem nessa estrutura foram implementadas em um programa desenvolvido para esse trabalho que simula os dispositivos reais que existiriam em uma aplicação real. Essa rota lógica poderia ter sido feita diretamente na plataforma da Konker, através de transformações, porém neste projeto essa funcionalidade da Konker não foi utilizada por simples praticidade de desenvolvimento já que os dispositivos utilizados foram simulados por um programa em Python, fazendo com que seja possível adicionar qualquer lógica desejada ao dispositivo de forma fácil. Essa decisão de projeto naturalmente não altera ou invalida os experimentos realizados.

3.2 A simulação dos dispositivos reais

Tendo a arquitetura da aplicação na Konker, agora necessitamos de dispositivos físicos que enviem dados e recebam dados para mudar de estado. Esses dispositivos foram simulados através de um programa, desenvolvido para exemplificar a aplicação, que envia dados simulando o usuário tomando as ações possíveis e apresentando a mudança no estado de outros dispositivos após cada ação. A comunicação entre esses dispositivos simulados com a plataforma da Konker foi realizada utilizando o protocolo MQTT.

3.3 Experimento

Com a plataforma da Konker configurada e os dispositivos físicos sendo simulados foi possível realizar testes de *stress* em cima dessa estrutura na tentativa de mensurar a escalabilidade da aplicação bem como encontrar os recursos computacionais que limitam essa escalabilidade.

Para a realização desse experimento foi desenvolvido um novo programa em python que configura as seguintes variáveis:

- Quantidade de residências simuladas;
- Taxa de atualização dos dispositivos simulados.

Sendo assim, podemos escolher quantas instâncias da aplicação da casa inteligente descrita queremos que estejam em execução e qual a taxa de atualização de cada dispositivo.

Nesse experimento, são criados dados de teste que são enviados para os canais de entrada dos dispositivos *Phone* e *PresenceSensor* e são capturadas as respostas que chegam nos canais de saída de cada dispositivo, medindo-se o tempo que levou entre o envio da mensagem no canal de entrada até o recebimento da resposta em cada canal de saída. Durante todo esse processo são medidos os recursos computacionais utilizados pela aplicação como um todo, isto é, tanto a plataforma da Konker quanto os dispositivos simulados gerando dados. Com isso, os seguintes dados foram obtidos:

- O tempo médio de entrega de uma mensagem;
- O tempo máximo de entrega de uma mensagem;
- A porcentagem média de memória utilizada;
- A porcentagem máxima de memória utilizada;
- A taxa média de escrita e leitura do disco;
- A taxa máxima de escrita e leitura do disco;
- A porcentagem média de utilização da CPU;
- A porcentagem máxima de utilização da CPU.

Foi controlado o número de mensagens que os dispositivos devem enviar durante o teste para garantir que todas as instâncias enviem uma mesma quantidade de mensagem, simulando um uso uniforme por cada instância.

A plataforma da Konker foi rodada localmente em uma máquina juntamente com programa que simula os dispositivos físicos enviando dados para a plataforma. O programa simulando os dispositivos foi responsável por medir o tempo de entrega de uma mensagem e os recursos computacionais utilizados foram anotados utilizando o analisador do sistema operacional dos processos referentes a plataforma Konker e a simulação dos dispositivos sendo que os dados anotados foram a soma de cada recurso que ambos os processos estavam utilizando.

A máquina na qual todos esses processos foram executados possui as seguintes especificações:

- Sistema: Ubuntu 20.04 LTS
- CPU: Intel(R) Core(TM) i7-8565U CPU 1.80GHz - 8 Threads, 4 Cores
- Memória RAM: 8GB DDR4 2400MHZ
- Disco rígido: 931GB 7200RPM - Leitura sequencial de 104MB/s, Escrita sequencial de 97 MB/s, Escrita e Leitura aleatória de 10MB/s

Em resumo o experimento teve um fluxo no seguinte formato:

- N instâncias da aplicação da casa inteligente são iniciadas cada uma em uma *thread* com os dispositivos configurados em uma taxa de atualização específica.
- Cada dispositivo na Konker recebe nos seus canais de entrada mensagens de teste com a taxa de atualização especificada, que são então processadas pela estrutura desenvolvida na Konker até chegar nos canais de saída;
- Foi medido o tempo decorrido entre o envio da mensagem nos canais de entrada até a entrega em cada canal de saída dos dispositivos. Esses dados geram a média e o pico do tempo de entrega de uma mensagem;
- Durante esse processo os recursos computacionais utilizados pelos processos da plataforma da Konker e dos dispositivos simulados são monitorados, gerando as outras medições.

4 Resultados

Após a realização do experimento descrito na Seção 3 foram obtidos os dados como descrito na metodologia.

4.1 Tempo de entrega de uma mensagem

Visualizando os dados obtidos em relação ao tempo de entrega de uma mensagem na tabela 1 e nos gráficos 2, 3, 4 fica claro que o tanto o tempo médio de entrega de uma mensagem quanto o tempo de pico de entrega de uma mensagem tem uma tendência de aumento linear. Além disso, é possível notar que o crescimento do tempo de pico é similar ao crescimento do tempo médio.

Não foi possível notar diferenças significativas nas tendências de crescimento para diferentes taxas de atualização dos dispositivos. Entretanto, analisando a tabela 1 é possível perceber uma tendência de queda tanto do tempo médio quanto do tempo de pico ao se aumentar a taxa de atualização dos dispositivos. Esse fato ocorre pois, com uma taxa de atualização maior dos dispositivos, existe um tempo maior entre o envio de cada mensagem, por exemplo, os dispositivos com taxa de atualização de 0,5 segundos vão enviar uma mensagem a cada 0,5 segundos, se estivéssemos simulando somente 1 dispositivo enviando mensagens isso daria 2 mensagens por segundo que a Konker teria que processar. Já para uma taxa de atualização de 2,5 segundos, simulando novamente somente 1 dispositivo enviando mensagens, a Konker teria que processar 2 mensagens a cada 5 segundos somente. Isso faz com que para taxas de atualizações menores o processamento fique mais concentrado em um período menor de tempo e o contrário para taxas de atualizações maiores, consequentemente fazendo com que a simulação possa ser executada mais rapidamente com taxas de atualizações maiores dos dispositivos. Mais a frente esse fato será analisado à luz do uso da CPU para confirmar se houve ou não uma menor carga para a máquina.

Por fim, podemos concluir que o tempo de resposta para o caso com 200 residências simuladas, utilizando o recurso computacional específico deste cenário avaliado, teve um valor elevado de tempo de resposta com picos que resultariam em uma espera de alguns

minutos. Assim, uma simulação de cenários com tais características podem demandar um maior poder computacional para ser capaz de gerar dados e processar requisições em tempo reduzido. Mais à frente neste relatório serão discutidas as limitações de desempenho que foram encontradas, principalmente no caso com 200 residências simuladas.

Tabela 1: Tempo de entrega de uma mensagem da plataforma da Konker e do script de simulação dos dispositivos para diferentes quantidades de residências simuladas e taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Tempo resp. médio (s)	Tempo de resp. pico (s)
1	0,5	0,034	0,111
1	1,5	0,030	0,144
1	2,5	0,036	0,124
10	0,5	0,221	1,032
10	1,5	0,209	1,036
10	2,5	0,202	1,188
50	0,5	17,287	28,407
50	1,5	7,092	10,350
50	2,5	1,111	6,058
75	0,5	20,806	44,987
75	1,5	9,875	35,189
75	2,5	1,646	9,237
100	0,5	25,408	45,949
100	1,5	14,454	38,536
100	2,5	11,075	34,282
150	0,5	37,325	69,834
150	1,5	27,729	59,365
150	2,5	21,628	53,489
200	0,5	60,757	78,945
200	1,5	59,250	69,517
200	2,5	55,640	68,189

4.2 Uso de memória

Visualizando os dados do uso de memória nos gráficos 5, 6, 7 foi possível notar que a memória foi pouco usada e não teve variações significativas ao se variar a quantidade de residências simuladas. Ao ser analisado na tabela 2 a variação do uso de memória em relação a taxa de atualização dos dispositivos, nenhuma variação foi encontrada.

Podemos concluir que esse recurso não limitou a escala das simulações e sequer teve variações significativas. Isso ocorre pois as simulações não trabalham com processamentos pesados ou uma quantidade grande de informações armazenadas, sendo que a principal tarefa de toda a aplicações é a redistribuição de mensagens para os dispositivos eventualmente com algum processamento básico de uma mensagem.

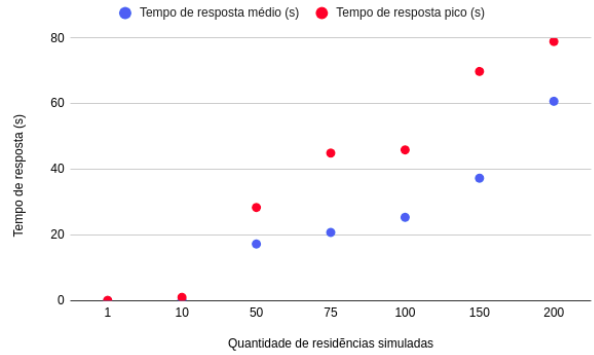


Figura 2: Tempo de entrega de uma mensagem (tempo de resposta) em segundos x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 0,5s

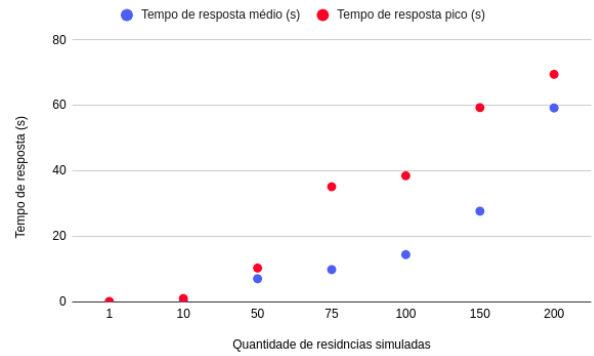


Figura 3: Tempo de entrega de uma mensagem (tempo de resposta) em segundos x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 1,5s

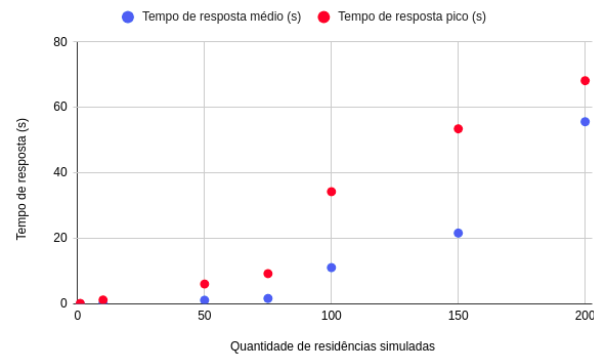


Figura 4: Tempo de entrega de uma mensagem (tempo de resposta) em segundos x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 2,5s

Tabela 2: Porcentagem de uso da memória da plataforma da Konker e do script de simulação dos dispositivos para diferentes quantidades de residências simuladas e taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Uso médio de mem. (%)	Uso mem. pico (%)
1	0,5	2,0	2,0
1	1,5	1,9	1,9
1	2,5	1,9	1,9
10	0,5	1,9	1,9
10	1,5	1,8	1,8
10	2,5	1,8	1,8
50	0,5	1,8	1,8
50	1,5	1,9	1,9
50	2,5	1,9	1,9
75	0,5	1,0	1,0
75	1,5	1,0	1,0
75	2,5	1,2	1,2
100	0,5	1,3	1,4
100	1,5	1,3	1,4
100	2,5	1,4	1,5
150	0,5	1,5	1,5
150	1,5	1,0	1,0
150	2,5	1,1	1,1
200	0,5	1,1	1,1
200	1,5	1,1	1,1
200	2,5	1,1	1,2

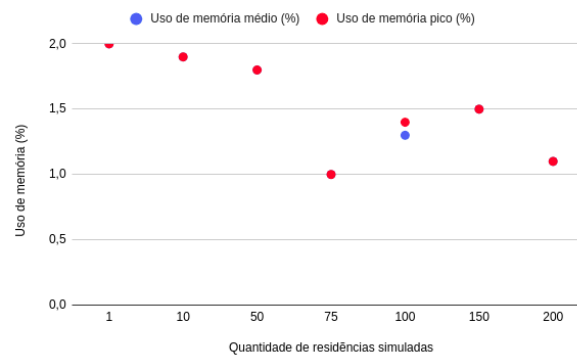


Figura 5: Porcentagem do uso de memória x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 0,5s

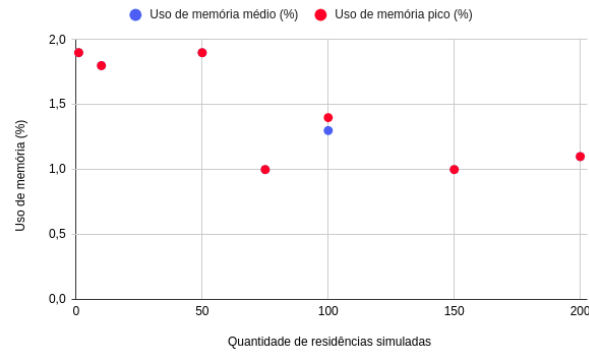


Figura 6: Porcentagem do uso de memória x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 1,5s

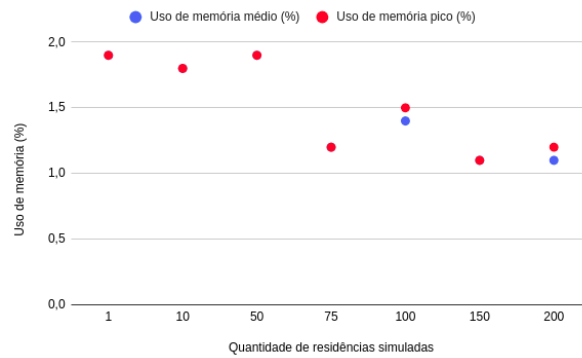


Figura 7: Porcentagem do uso de memória x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 2,5s

4.3 Uso do Disco Rígido

A utilização do disco rígido foi medido em KB/s (Kilo bytes por segundo) e não foi diferenciada a taxa de escrita e de leitura. Nos resultados apresentados no gráficos 8, 9, 10 é possível perceber primeiro uma tendência de aumento na taxa de leitura e escrita no disco; entretanto, essa tendência é quebrada principalmente no caso com 200 residências simuladas e também no caso com 150 residências simuladas quando a taxa de atualização dos dispositivos foi de 2,5 segundos. Isso ocorreu devido a uma carga muito alta no processamento que fez a simulação deteriorar – mais detalhes sobre esse problema serão discutidos na Subseção de Uso da CPU.

Além disso, a variação em relação a taxa de atualização dos dispositivos observando a Tabela 3 foi possível notar uma leve tendência de aumento da média na taxa de leitura e escritura e uma leve queda no pico de leitura e escrita, o que condiz com o fato do processamento ficar mais espaçado para os casos com maior taxa de atualização, fazendo acesso ao disco ficar mais uniforme.

Por fim podemos concluir que o uso do disco variou consideravelmente com o número de residências simuladas, e também variou com a taxa de atualização dos dispositivos. Entretanto, não foi um recurso que limitou a escala das simulações apesar de ter sido afetado pelo limite observado no uso da CPU.

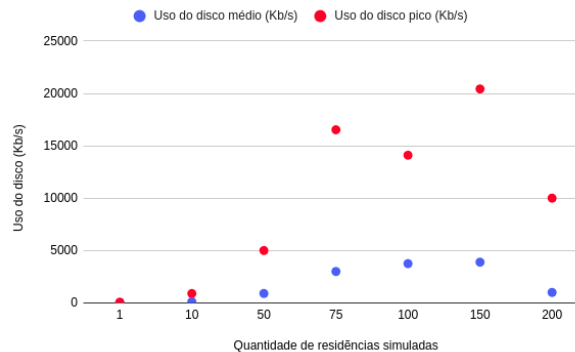


Figura 8: Uso disco rígido em KB/s x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 0,5s

4.4 Uso da CPU

Em primeiro lugar, é importante ressaltar que o uso da CPU medido em porcentagem pode ultrapassar 100% pois a mediação feita é por núcleo de processamento logo um processador com mais de um núcleo pode apresentar medidas maiores do 100% caso o uso de vários núcleos estejam altos o suficiente.

Uma vez com essa informação, analisando os dados dos gráficos 11, 12, 13 podemos perceber claramente o aumento do uso da CPU com o aumento no número de residências simuladas tanto da média de uso quanto do pico.

Tabela 3: Taxa de escrita e leitura no disco rígido da plataforma da Konker para diferentes quantidades de residências simuladas e taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Uso médio de disco (KB/s)	Uso disco pico (KB/s)
1	0,5	49	65
1	1,5	15	21
1	2,5	10	22
10	0,5	120	900
10	1,5	96	511
10	2,5	85	390
50	0,5	900	5000
50	1,5	998	3600
50	2,5	1200	3000
75	0,5	3000	16530
75	1,5	3200	12970
75	2,5	3420	14580
100	0,5	3750	14100
100	1,5	3880	13050
100	2,5	3550	14420
150	0,5	3890	20430
150	1,5	4000	15780
150	2,5	4100	4800
200	0,5	1000	10000
200	1,5	1200	12000
200	2,5	1150	1030

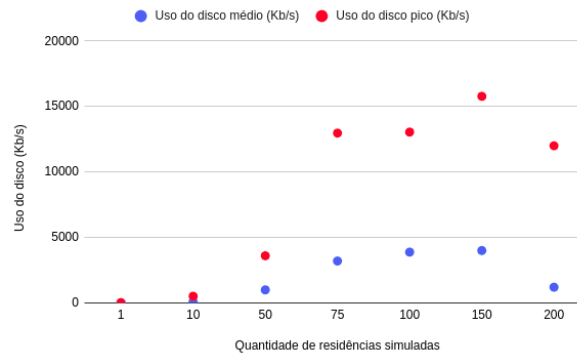


Figura 9: Uso disco rígido em KB/s x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 1,5s

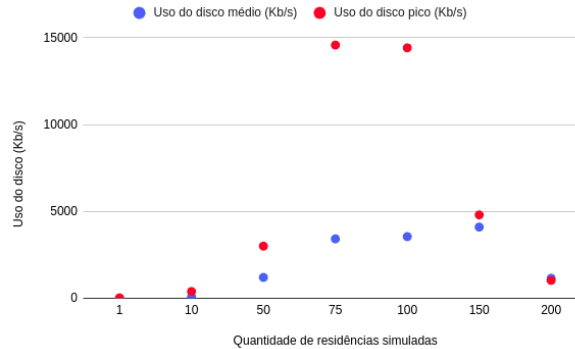


Figura 10: Uso disco rígido em KB/s x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 2,5s

Analisando a tabela 4 em relação a taxa de atualização foi percebido uma leve queda no uso da CPU tanto da média quanto do pico de uso, concluindo o pensamento criado na discussão sobre o tempo de entrega de uma mensagem, já que taxas de atualização maiores tiveram uma média e um pico de uso da CPU menores se comparado as taxas de atualização menores.

Podemos concluir que esse recurso foi o recurso limitante na escala dessa simulação, pois acima de 200 residências simuladas a máquina apresentou lentidão constante e altíssimo uso da CPU, sendo este também o principal recurso que modifica o tempo de entrega de uma mensagem e que também atua na variação da taxa de escrita e leitura no disco.

Por fim, é importante comentar que no caso de 200 residências simuladas o uso da CPU foi extremamente elevado em alguns momentos, causando lentidão na máquina e impossibilitando todas as residências simuladas terminarem de enviar todas as mensagens que deveriam enviar. Portanto, no caso de 200 residências e 150 residências o número de mensagens trocadas entre os dispositivos acabou sendo menor, pois o tempo necessário para que o número mensagens trocadas se igualasse com as outras simulações feitas com menos residências ultrapassou 1 hora e nesse experimento não foram realizados testes aguardando mais tempo do que isso para todas as mensagens serem trocadas. Isso explica a diminuição que aparece no uso de disco em alguns casos, com número de residências de 150 ou 200, já que se menos mensagens forem trocadas, então menos interações com o disco são necessárias pela plataforma da Konker.

4.5 Uso da Konker e simulação dos dispositivos

Os dados apresentados até o momento, como descrito anteriormente, são calculados pela soma do uso de cada recurso pela plataforma da Konker mais o uso dos *scripts* que simulam os dispositivos. Nesta seção apresentamos os dados separados para analisar o uso de cada recurso por cada um desses processos.

Tabela 4: Porcentagem de uso da CPU da plataforma da Konker e do script de simulação dos dispositivos para diferentes quantidades de residências simuladas e taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Uso CPU médio (%)	Uso CPU pico (%)
1	0,5	3,1	8,2
1	1,5	1,3	4,0
1	2,5	0,7	4,5
10	0,5	7,3	29,0
10	1,5	5,9	26,1
10	2,5	5,0	18,0
50	0,5	70,1	250,2
50	1,5	63,0	190,8
50	2,5	55,3	200,1
75	0,5	75,3	260,7
75	1,5	72,0	200,0
75	2,5	70,5	200,6
100	0,5	83,0	230,0
100	1,5	80,4	199,1
100	2,5	77,4	202,3
150	0,5	89,8	309,1
150	1,5	89,4	302,3
150	2,5	87,1	299,1
200	0,5	200,5	400,9
200	1,5	195,7	410,2
200	2,5	199,8	380,7

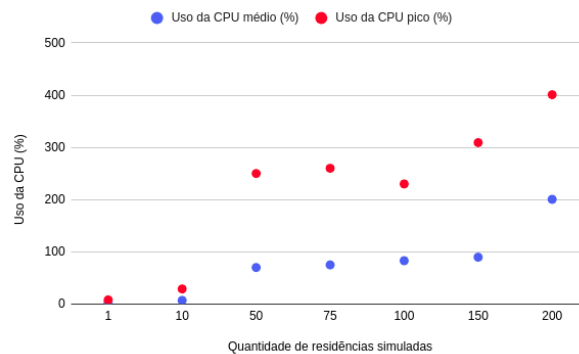


Figura 11: Porcentagem do uso da CPU x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 0,5s

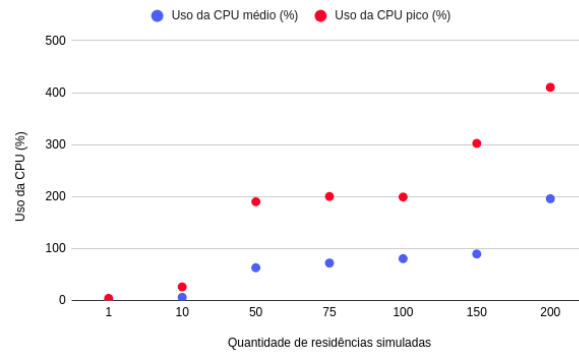


Figura 12: Porcentagem do uso da CPU x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 1,5s

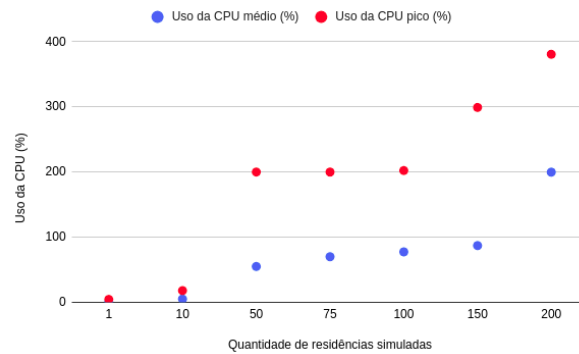


Figura 13: Porcentagem do uso da CPU x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 2,5s

4.5.1 Uso do disco rígido da plataforma da Konker e do script que simula os dispositivos físicos

O uso do disco rígido anotado foi inteiramente o uso da plataforma da Konker já que o script de simulação dos dispositivos não teve taxas consideráveis de uso do disco.

4.5.2 Uso de memória da plataforma da Konker e do script que simula os dispositivos físicos

Como discutido anteriormente a memória não apresentou grandes variações e não foi um recurso que limitou a escala das simulações. Olhando para o uso de memória de cada processo separadamente nos gráficos 14, 15, 16 percebe-se que o uso de memória também não apresenta grandes variações em relação a quantidade de residências simuladas, entretanto nota-se que o processo da plataforma da Konker utilizou mais memória do que o script de simulação dos dispositivos.

Por fim nas tabelas 5, 6 fica claro que não houve variação para diferentes taxas de atualização de dispositivos, logo para nenhum dos processos a memória em nenhum momento durante as simulação foi um recurso crítico para a escala das simulações.

Tabela 5: Porcentagem de uso da memória do processo da plataforma da Konker para diferentes quantidades de residências simuladas e diferentes taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Uso mem. media (%)	Uso mem. pico (%)
1	0,5	1,9	1,9
1	1,5	1,8	1,8
1	2,5	1,8	1,8
10	0,5	1,8	1,8
10	1,5	1,7	1,7
10	2,5	1,7	1,7
50	0,5	1,7	1,7
50	1,5	1,8	1,8
50	2,5	1,8	1,8
75	0,5	0,7	0,8
75	1,5	0,7	0,8
75	2,5	0,9	0,9
100	0,5	1	1,1
100	1,5	1	1,1
100	2,5	1,1	1,2
150	0,5	1,2	1,3
150	1,5	0,7	0,7
150	2,5	0,9	0,9
200	0,5	0,9	0,9
200	1,5	0,9	0,9
200	2,5	0,9	1

Tabela 6: Porcentagem de uso da memória do script de simulação dos dispositivos para diferentes quantidades de residências simuladas e diferentes taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Uso mem. media (%)	Uso mem. pico (%)
1	0,5	0,1	0,1
1	1,5	0,1	0,1
1	2,5	0,1	0,1
10	0,5	0,1	0,1
10	1,5	0,1	0,1
10	2,5	0,1	0,1
50	0,5	0,1	0,1
50	1,5	0,1	0,1
50	2,5	0,1	0,1
75	0,5	0,3	0,2
75	1,5	0,3	0,2
75	2,5	0,3	0,3
100	0,5	0,3	0,3
100	1,5	0,3	0,3
100	2,5	0,3	0,3
150	0,5	0,3	0,2
150	1,5	0,3	0,3
150	2,5	0,2	0,2
200	0,5	0,2	0,2
200	1,5	0,2	0,2
200	2,5	0,2	0,2

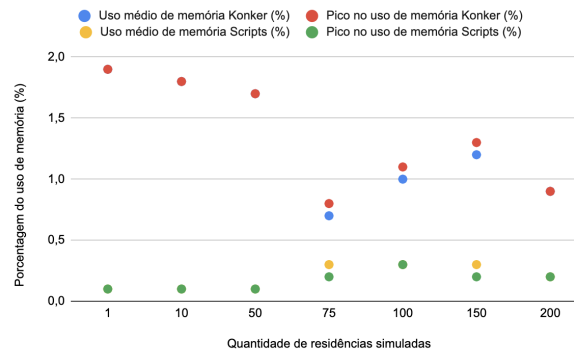


Figura 14: Porcentagem do uso da memória x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 0,5s comparando o uso entre processo da Konker e processo de simulação dos dispositivos

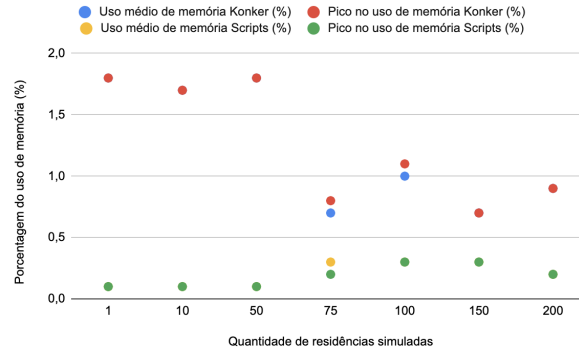


Figura 15: Porcentagem do uso da memória x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 1,5s comparando o uso entre processo da Konker e processo de simulação dos dispositivos

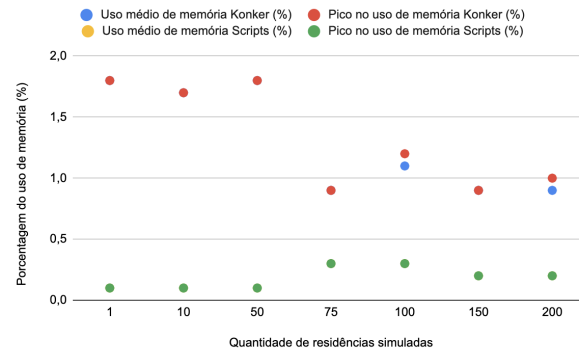


Figura 16: Porcentagem do uso da memória x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 2,5s comparando o uso entre processo da Konker e processo de simulação dos dispositivos

4.5.3 Uso da CPU da plataforma da Konker e do script que simula os dispositivos físicos

O uso da CPU já foi apresentado como o recurso que limitou a escalabilidade da simulação anteriormente com a análise da soma dos recursos de cada processo. Desta vez com a análise para cada processo podemos notar pelos gráficos 17, 18, 19 que a plataforma da Konker utilizou na média mais a CPU do que o script de simulação dos dispositivos, e o pico de uso da CPU de ambos o processo é bastante similar. Com isso podemos perceber que esses dois processos tem características de processamento diferentes. Enquanto o processo do script de simulação dos dispositivos tem um pico de processamento e fica com uma média baixa a plataforma da Konker tem um pico similar ao pico do script porem permanece em uma média maior de uso da CPU. Essas características são esperadas já que o script deve enviar mensagens com uma taxa de atualização definida então no momento de envio das mensagens ocorre um pico de processamento e após o envio o script não tem muitas tarefas para executar e a plataforma da Konker que tem que realizar todo o processamento dessas mensagens.

Além disso analisando as tabelas 7, 8 podemos ver uma diminuição no uso da CPU tanto na média quanto no pico com o aumento da taxa de atualização. Essa queda ocorre de forma mais expressiva no processo do script de simulação dos dispositivos já que esse processo tem como principal tarefa o envio das mensagens na taxa definida.

Por fim podemos concluir que o processo da Konker necessitou de mais processamento já que teve um pico na média do uso da CPU expressiva para o caso de 200 residências simuladas enquanto que o script não teve um aumento tão expressivo.

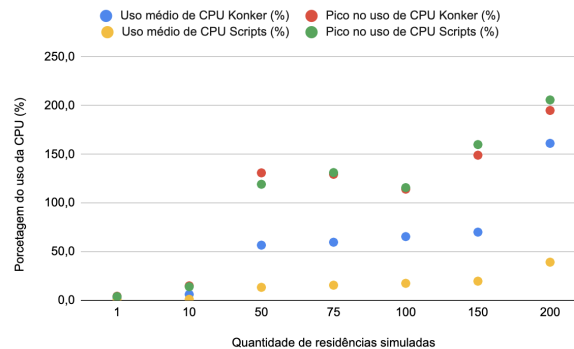


Figura 17: Porcentagem do uso da CPU x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 0,5s comparando o uso entre processo da Konker e processo de simulação dos dispositivos

5 Conclusão

Neste relatório apresentamos a plataforma da Konker e validamos seu uso como uma alternativa viável para simulação de aplicações IoT, desenvolvendo uma aplicação que controla

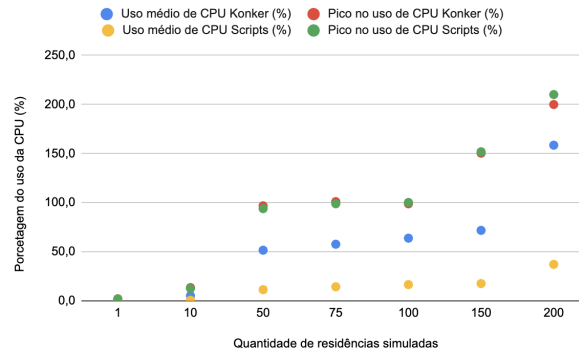


Figura 18: Porcentagem do uso da CPU x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 1,5s comparando o uso entre processo da Konker e processo de simulação dos dispositivos

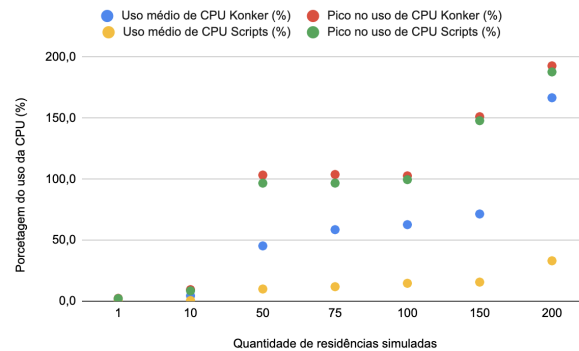


Figura 19: Porcentagem do uso da CPU x Quantidade de residências simuladas com os dispositivos das residências configurados para uma taxa de atualização de 2,5s comparando o uso entre processo da Konker e processo de simulação dos dispositivos

Tabela 7: Porcentagem de uso da CPU do processo da plataforma da Konker para diferentes quantidades de residências simuladas e diferentes taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Uso CPU médio (%)	Uso CPU pico (%)
1	0,5	2,6	4,3
1	1,5	1,1	2,1
1	2,5	0,6	2,4
10	0,5	6,1	14,9
10	1,5	5,2	13,5
10	2,5	4,4	9,5
50	0,5	56,7	131,0
50	1,5	51,6	96,9
50	2,5	45,3	103,4
75	0,5	59,7	129,4
75	1,5	57,7	101,1
75	2,5	58,6	103,8
100	0,5	65,5	114,2
100	1,5	63,8	98,9
100	2,5	62,7	102,7
150	0,5	70,1	149,1
150	1,5	71,8	150,4
150	2,5	71,5	151,2
200	0,5	161,3	195,1
200	1,5	158,6	200,0
200	2,5	166,7	192,8

uma casa inteligente simples com 4 dispositivos e que o controle dos dispositivos se dá por um celular. Todos os dispositivos tiveram uma representação na plataforma da Konker, se comunicando com ela através do protocolo MQTT.

Por fim, concluímos que com essa plataforma também é possível fazer experimentos simulando os dispositivos com o intuito de fazer testes de escalabilidade de uma aplicação, notando que o principal recurso computacional que limitou a escala da simulação foi o uso da CPU. Entretanto durante esse processo alguns pontos poderiam ser melhorados para experimentos mais avançados no futuro, sendo eles:

1. A criação de uma biblioteca que facilite a criação de dispositivos simulados que podem se comunicar com a Konker para o desenvolvimento mais rápido de cenário de simulação;
2. Testes com poder computacional maior, por exemplo utilizando uma nuvem para testes de escala mais próximos um ambiente de uso real;
3. Uma comparação com outros serviços oferecidos para analisar competitivamente o desempenho da plataforma da Konker em relação a outros serviços de simulações de IoT oferecidos.

Tabela 8: Porcentagem de uso da CPU do script que simula os dispositivos para diferentes quantidades de residências simuladas e diferentes taxas de atualizações dos dispositivos

Residências	Taxa de atual. (s)	Uso CPU médio (%)	Uso CPU pico (%)
1	0,5	0,5	3,9
1	1,5	0,2	1,9
1	2,5	0,1	2,1
10	0,5	1,2	14,1
10	1,5	0,7	12,6
10	2,5	0,6	8,5
50	0,5	13,4	119,2
50	1,5	11,4	93,9
50	2,5	10,0	96,7
75	0,5	15,6	131,3
75	1,5	14,3	98,9
75	2,5	11,9	96,8
100	0,5	17,5	115,8
100	1,5	16,6	100,2
100	2,5	14,7	99,6
150	0,5	19,7	160,0
150	1,5	17,6	151,9
150	2,5	15,6	147,9
200	0,5	39,2	205,8
200	1,5	37,1	210,2
200	2,5	33,1	187,9

Referências

- [1] L. Bittencourt, R. Immicha, R. Sakellarioub, N. Fonseca, E. Madeiraa, M. Curadoc, L. Villas, L. DaSilva, C. Lee, O. Ranaf, *The Internet of Things, Fog and Cloud continuum: Integration and challenges*. Internet of Things (2018).
- [2] C. Puliafito, D. M. Gonçalves, M. M. Lopes, L. L. Martins, E. Madeira, E. Mingozi, O. Rana, L. F. Bittencourt, *MobFogSim: Simulation of mobility and migration for fog computing*, Simulation Modelling Practice and Theory (2020).
- [3] X. Zeng, S. K. Garg, P. Strazdins, P. P. Jayaraman, D. Georgakopoulos, R. Ranjan, *IOTSim: A simulator for analysing IoT applications*, Journal of Systems Architecture (2017).