

Optimizing Energy Savings on University Campus Using IoT and Automated Planning

Kassia C. S. Lopes¹, Wellington Umbelino¹, Ana Lívia Lopes¹, Arthur R. da Silva¹, Cecília M. C. Mesquita¹, Marcus V. Nogueira¹, Maria E. S. Marques¹, Pedro N. Damacena¹, William M. Brito¹, Viviane Menezes¹, Michel Sales¹

¹Universidade Federal do Ceará – Campus Quixadá

{kassia.lopes, wellingtonsi, slopeslivia, arthurroberto, cecimesquita}@alu.ufc.br

{marcusnogueira, eduardamarques, pedronobrega, william04}@alu.ufc.br

{vivianemenezes, michelsb}@ufc.br

Abstract. *Population growth and economic development have driven increased energy consumption. In public buildings, HVAC and lighting systems account for a significant portion of this consumption. The adoption of automation technologies, using Internet of Things (IoT)-based devices, can help reduce energy waste in these types of environments. However, solutions based exclusively on IoT tend to disregard the dynamic nature of spaces and user behavior. To overcome this limitation, Artificial Intelligence-based approaches, such as Automated Planning, have been used to manage these devices and achieve goals such as minimizing energy consumption while maintaining user comfort and safety. In this context, this work proposes the integration of a physical layer composed of sensors and actuators into an automated planning system, enabling devices to self-adapt to specific environmental conditions and needs in real time. A preliminary version of the proposed system was deployed on the Federal University of Ceará Campus in Quixadá (UFC-Quixadá), enabling intelligent monitoring and control of devices in real environments.*

Resumo. *O crescimento populacional e o desenvolvimento econômico têm impulsionado o aumento do consumo de energia. Nos prédios públicos, os sistemas de climatização e iluminação correspondem a uma parcela expressiva desse consumo. A adoção de tecnologias de automação, com o uso de dispositivos baseados em Internet das Coisas (IoT), pode contribuir para a redução do desperdício de energia nesse tipo de ambiente. No entanto, soluções baseadas exclusivamente em IoT tendem a desconsiderar a dinamicidade dos espaços e o comportamento dos usuários. Para superar essa limitação, abordagens baseadas em Inteligência Artificial, como Planejamento Automatizado, têm sido utilizadas para gerenciar esses dispositivos e alcançar metas como a minimização do consumo de energia mantendo o conforto e a segurança dos usuários. Neste contexto, este trabalho propõe a integração de uma camada física composta por sensores e atuadores a um sistema de planejamento automatizado, possibilitando a auto-adaptação dos dispositivos às condições e necessidades específicas do ambiente em tempo real. Uma versão preliminar do sistema proposto foi implementada no Campus da Universidade Federal do Ceará em Quixadá (UFC-Quixadá), permitindo o monitoramento e o controle inteligente dos dispositivos em ambientes reais.*

1. Introdução

O crescimento populacional e o desenvolvimento socioeconômico das últimas décadas têm intensificado a demanda global por energia, resultando em impactos ambientais significativos. No Brasil, as edificações urbanas — incluindo residências, comércios e indústrias — respondem por cerca de 79% do consumo total de eletricidade, segundo dados do Anuário Estatístico de Energia Elétrica [ENGIE 2024]. Diante desse cenário, a busca por práticas de eficiência energética tornou-se essencial para reduzir custos e mitigar os efeitos ambientais decorrentes do uso excessivo de recursos energéticos.

Edifícios públicos, em particular, são grandes consumidores de energia, com destaque para os sistemas de iluminação e climatização, que respondem por parcela significativa do consumo. Aproximadamente 20% do consumo total de qualquer edificação provém do setor de AVAC (aquecimento, ventilação e ar-condicionado). Nesse contexto, soluções digitais baseadas na Internet das Coisas (IoT) têm sido amplamente exploradas para promover automação e monitoramento inteligente desses ambientes, visando maior eficiência e sustentabilidade [Rio et al. 2018].

Universidades federais brasileiras também enfrentam desafios relacionados ao consumo energético elevado. Em 2022, a Universidade Federal do Ceará (UFC) gastou mais de R\$14,8 milhões anuais [Universidade Federal do Ceará 2022]. Iniciativas como a campanha UFC Sustentável demonstram o compromisso institucional com a redução do desperdício energético. Paralelamente, projetos como o *Smart Campus* [Amurim et al. 2021], desenvolvido na UFC - Quixadá, vêm explorando o uso de dispositivos IoT para controle automatizado de equipamentos, como lâmpadas e ar-condicionado, integrando sensores para monitoramento de temperatura e umidade.

No entanto, estudos recentes indicam que soluções baseadas exclusivamente em IoT podem atuar de forma limitada no problema, sem considerar variáveis como ocupação dos ambientes, conforto humano e condições naturais, como iluminação solar [Georgievski et al. 2023]. Para superar essas limitações, abordagens de Inteligência Artificial (IA), como Aprendizado de Máquina e Planejamento Automatizado tem sido utilizadas para economia de energia em prédios inteligentes. Planejamento Automatizado é uma subárea da IA que se concentra na criação de sequências de ações para que um agente inteligente alcance suas metas [Ghallab et al. 2004]. Diferentemente de aplicações baseadas em dados [Rojek et al. 2025], o uso de Planejamento Automatizado independe da necessidade de grandes volumes de dados. Em [Georgievski et al. 2023], os autores definem ações para que um planejador otimize o uso de recursos energéticos, priorizando economia de energia sem comprometer o conforto dos usuários. Nesse contexto, este trabalho apresenta uma proposta de integração de planejamento automatizado para gerenciamento de dispositivos IoT no Campus da UFC - Quixadá, com foco na redução do consumo energético em ambientes universitários.

O restante do trabalho está organizado da seguinte forma. A Seção 2 apresenta os conceitos teóricos para entendimento do trabalho tais como Planejamento Automatizado e Internet das Coisas. Na Seção 3 são apresentados os trabalhos relacionados. A Seção 4 discute a proposta e, por fim, na Seção 5 são apresentadas a conclusão e trabalhos futuros.

2. Fundamentação Teórica

2.1. Planejamento Automatizado

Planejamento Automatizado é a subárea da IA que estuda o processo de escolha e organização de ações para que um agente inteligente alcance suas metas. O domínio de planejamento é a descrição formal das características do agente e do ambiente. Ele é composto por um conjunto de predicados, que descrevem as características do agente e do ambiente, e por um conjunto de ações, que descrevem as habilidades do agente no ambiente [Ghallab et al. 2004]. Para modelar ambientes reais, o domínio de planejamento terá um grande número de estados, o que impossibilita a representação explícita por meio de um grafo de transição de estados. Por essa razão, a comunidade de planejamento automatizado utiliza uma representação implícita dos domínios, por meio de linguagens de descrição de ações tais como PDDL (*Planning Domain Description Language*) [Haslum et al. 2019].

Em um domínio de planejamento, as ações são definidas por meio de pré-condições e efeitos. As pré-condições representam os requisitos necessários para que uma ação possa ser executada em um dado estado. Os efeitos, por sua vez, descrevem as mudanças provocadas pela execução da ação, modificando o estado atual do mundo. A Figura 1 apresenta um trecho do domínio de planejamento chamado *elevator* [Ghallab et al. 2004, Haslum et al. 2019] que modela o comportamento de elevadores em um prédio com múltiplos andares e passageiros que precisam ser transportados de um andar de origem para um andar de destino. Na linha 1 temos a definição do nome do domínio (:define). Na linha temos o tipo de requerimentos que há no domínio (:requirements). Neste caso, é um domínio com variáveis tipadas (:typing). Os predicados (:predicates) desse domínio são (linhas 5 a 8): passenger-at, indica se o passageiro (?person) está em um andar específico (?floor). boarded, se o passageiro (?person) embarcou em um andar (?lift). lift-at, indica que o elevador (?lift) está em um dado andar (floor) e next, que representa a ordem dos andares. A ação mostrada na figura é a move-up que modela o ato de um elevador subir de um andar para outro. Os parâmetros dessa ação são (linha 10): ?lift do tipo elevator e ?cur e ?nxt do tipo num. As pré-condições são (linhas 11 e 12) que o elevador tem que está no andar ?cur e que ele precisa chegar em um andar nxt. Os efeitos da ação (linhas 13 e 14) são que o elevador não estará mais no andar ?cur (and (not (lift-at ?lift ?cur))) e estará em um andar ?nxt (lift-at ?lift ?nxt).

Um problema de planejamento é definido em um domínio por meio de um estado inicial e a descrição de uma meta a ser alcançada pelo agente. A Figura 2 ilustra um exemplo de problema para o domínio apresentado na Figura 1. Nas linhas 1 e 2 temos, respectivamente, o nome do problema e o domínio ao qual este problema pertence. Nas linhas 3 a 6 temos a descrição dos objetos do problema, no qual temos: 5 andares, representados pelos objetos n1, n2, n3, n4 e n5 (linha 4); 3 passageiros, representados pelos objetos p1, p2, p3 (linha 5) e 2 elevadores, representados pelos objetos e1 e e2 (linha 6). Na situação inicial (linhas 7 a 10) é descrito como andares estão conectados (next n1 n2,), que o elevador e1 está no andar n1 (lift-at e1 n1), que o elevador e2 está no andar n5 (lift-at e2 n5), que os passageiros p1 e p2 estão no andar n2 (passenger-at p1 n2, passenger-at p2 n2) e que o passageiro p3 está no andar n4 (passenger-at p3 n4). A meta é que todos os passageiros (p1, p2 e p3)

```

1  (define (domain: elevators))
2    (:requirements :typing)
3    (:types elevator passenger num - object)
4    (:predicates
5      (passenger-at ?person - passanger ?floor - num)
6      (boarded ?person - passenger ?lift - elevator)
7      (lift-at ?lift - elevator ?floor - num)
8      (next ?n1 - num ?n2 - num))
9    (:action move-up
10      :parameters (?lift - elevator ?cur ?nxt - num)
11      :precondition (and (lift-at ?lift ?cur) (next ?cur ?nxt))
12      :effect (and (not (lift-at ?lift ?cur)) (lift-at ?lift ?nxt
13        ))))
13      ...

```

Figure 1. Exemplo de um trecho do domínio *elevator* [Haslum et al. 2019].

estejam no andar n1.

```

1  (define (problem elevators-problem)
2    (:domain elevators)
3    (:objects
4      n1 n2 n3 n4 n5 - num
5      p1 p2 p3 - passenger
6      e1 e2 - elevator)
7    (:init
8      (next n1 n2) (next n2 n3) (next n3 n4) (next n4 n5)
9      (lift-at e1 n1) (lift-at e2 n5)
10     (passenger-at p1 n2) (passenger-at p2 n2) (passenger-at p3 n4))
11    (:goal (and (passenger-at p1 n1) (passenger-at p2 n1) (passenger-at
12      p3 n1))))
```

Figure 2. Exemplo de um problema de planejamento no domínio *elevator* [Haslum et al. 2019].

A solução é um *plano*, definido como uma sequência de ações que leva o agente do estado inicial para um estado que satisfaz a meta [Ghallab et al. 2004]. Um plano pode possuir métricas (*metric*), descritas no problema, que especificam a base sobre o qual um plano será avaliado [?]. Por fim, o planejador é um algoritmo que, dado um domínio e um problema de planejamento, retorna um plano que possibilite o agente sair do estado inicial e alcançar suas metas ou uma falha, caso não exista solução [Russell and Norvig 2004].

2.1.1. Expressões Numéricas e Ações Durativas

Em ambientes reais há a necessidade de raciocinar com processos que dependem diretamente do tempo para serem executados. Problemas como gerenciamento de energia envolvem grandezas sujeitas a esses processos [Coles et al. 2009]. A versão 2.1 da linguagem PDDL é capaz de expressar propriedades temporais e numéricas dos domínios de planejamento com a adição de fluentes numéricos e ações durativas [Fox and Long 2003].

As expressões numéricas são definidas por meio de operadores aritméticos, a partir de expressões numéricas primitivas, que são valores associados a tuplas de objeto por meio de funções de domínio. A sintaxe possibilita a expressão de atualizações a atribuições numéricas, podendo estas serem diretas ou relativas, como aumentar (*increase*) e diminuir (*decrease*). Por meio dos fluentes numéricos é possível a representação de atributos como quantidade de recursos, utilidade acumulada ou contadores [Fox and Long 2003].

As ações durativas são aquelas que ocorrem ao longo de um intervalo de tempo. Para definição das ações durativas é necessário especificar se o efeito será imediato ou no fim de uma ação, além de informar se a condição deve ser válida no inicio, fim ou durante toda a ação. Para isso não usadas as notações (*at start*), no início, (*end start*), no fim, (*over all*), ao longo do intervalo de tempo. A Figura 3 ilustra uma versão durativa (:durative-action) da ação move-up apresentada na Figura 1, representando de forma mais realista a passagem do elevador de um andar para outro, que ocorre ao longo de um intervalo de tempo. A duração da ação leva em consideração a distância entre o andar atual (?cur) e o destino (?nxt), calculada pela função *floor_distance* e a velocidade do elevador, indicada pela função *elevator_speed* (linhas 3 e 4). As condições da ação (linhas 9 e 10) são que no ínicio o elevador deve estar no andar de origem e que a relação ?next deve ser válida durante toda a ação. Os efeitos da ação (linhas 11 e 12) são que o elevador deixa de estar no andar atual ((*at start (not (lift-at ?lift ?cur))*) e que chega no próximo andar ao fim da ação (*at end (lift-at ?lift ?nxt))*).

```

1 (:functions
2 (person_speed ?person - passenger)
3 (elevator_speed ?lift - elevator)
4 (floor_distance ?f1 ?f2 - num))
5 (:durative-action move-up
6   :parameters (?lift - elevator ?cur ?nxt - num)
7   :duration (= ?duration (/ (floor_distance ?cur ?nxt)
8     (elevator_speed ?lift)))
9   :condition (and (at start (lift-at ?lift ?cur))
10      (over all (next ?cur ?nxt)))
11   :effect (and (at start (not (lift-at ?lift ?cur)))
12      (at end (lift-at ?lift ?nxt))))
```

Figure 3. Exemplo de uma ação durativa do domínio *elevator* [Haslum et al. 2019]

2.2. Internet das Coisas

A tecnologia Internet das Coisas (IoT) vem ganhando grande destaque nas últimas décadas. A interconectividade entre dispositivos, juntamente com a integração fluida das tecnologias IoT no cotidiano, vem transformando o panorama da tecnologia [Mesquita et al. 2024]. As redes de sensores sem fio vêm sendo utilizadas cada vez mais em diversas áreas, como saúde, agricultura, monitoramento de ambiente, medição inteligente, dentre tantas outras. Além disso, diversos tipos de sistemas de comunicação sem fio vêm sendo empregados na comunicação desses sensores, como Bluetooth, satélites, radiofrequência, entre outros [Dias et al. 2019].

2.2.1. Protocolo MQTT

O MQTT (Message Queuing Telemetry Transport) é um protocolo leve e eficiente, projetado para a transmissão de mensagens em dispositivos com recursos limitados de hardware e restrições de largura de banda. Ele opera sobre o protocolo TCP/IP e utiliza o modelo de comunicação *Publish/Subscribe*, que permite a comunicação bilateral entre dispositivos conectados através de um componente central denominado *broker* [da Conceição and de Resende Costa 2023, Aleesha and Laseena 2022].

No modelo *Publish/Subscribe*, o *broker* é responsável por gerenciar e encaminhar as mensagens publicadas em tópicos para os dispositivos assinantes (*subscribers*). Um *tópico* é um canal ou categoria de mensagens, ao qual os dispositivos interessados se inscrevem para receber informações específicas. Os dispositivos que enviam mensagens a um tópico são chamados de *publishers*, enquanto aqueles que recebem as mensagens são denominados *subscribers* [da Conceição and de Resende Costa 2023].

Além disso, o MQTT oferece comunicação assíncrona e é especialmente indicado para ambientes IoT onde a largura de banda é limitada e a conectividade pode ser intermitente. O protocolo também permite configurar diferentes níveis de Qualidade de Serviço (QoS), garantindo a entrega das mensagens de acordo com as necessidades da aplicação, desde uma entrega "no máximo uma vez" até uma entrega "exatamente uma vez" [Aleesha and Laseena 2022].

Por essas características, o MQTT é amplamente adotado em aplicações IoT que exigem eficiência energética e redução do tráfego de rede, fundamentais para a sustentabilidade dos dispositivos conectados [da Conceição and de Resende Costa 2023, Aleesha and Laseena 2022].

3. Trabalhos Relacionados

O trabalho de [Georgievski et al. 2023] propõe o uso de planejamento automatizado para gerenciar dispositivos IoT em edifícios inteligentes, com o intuito de reduzir os custos com energia elétrica. Os autores avaliaram a eficácia do sistema através de simulações e constataram redução de 43% no custo da energia com uso do planejador. O presente estudo busca implementar essa proposta instalando sensores no campus da UFC -Quixadá, adequando as métricas para o cenário do país e realizar avaliações a partir de dados reais.

Já o trabalho de [Amurim et al. 2021] implementa uma solução IoT baseada no *middleware* FIWARE para controle do consumo de energia e automação de serviços em um campus universitário, utilizando sensores de temperatura, corrente, câmeras com reconhecimento facial e atuadores. Em nosso trabalho, seguimos uma linha semelhante de arquitetura e objetivos, porém avançamos ao integrar técnicas de IA para gerenciamento automático dos dispositivos, ampliando a capacidade de adaptação e eficiência do sistema.

E, por fim, [Ma et al. 2023a] desenvolveu o DEMSA, um *middleware* habilitado por *Digital Twin* (DT) para espaços inteligentes autoadaptativos. A solução integra sensores IoT, simulações preditivas e um planejador para equilibrar conforto térmico e consumo energético em edifícios, com foco especial em sistemas HVAC. O sistema realiza simulações do ambiente físico, coleta *feedback* dos ocupantes e utiliza técnicas de planejamento automatizado para ajustar dinamicamente as estratégias de controle. Em nosso

trabalho, seguimos uma abordagem semelhante, porém avançamos ao implantar os sensores em um ambiente físico, de modo que o planejador possa trabalhar com dados reais.

4. Integrando IoT e Planejamento Automatizado para Economia de Energia em um Campus Universitário

Para controlar o ambiente de forma auto-adaptativa e otimizar o consumo de energia, este trabalho propõe o desenvolvimento de uma camada física de dispositivos IoT e a integração da mesma a um sistema de planejamento automatizado para controlar os sistemas de climatização, iluminação e outros recursos energéticos no campus da UFC - Quixadá. Essa integração permitirá que as ações executadas pelos dispositivos sejam decididas pelo planejador, com base nas condições reais do ambiente, diferentemente de outros trabalhos como [Georgievski et al. 2023] e [Ma et al. 2023b], que analisaram o uso do planejamento automatizado apenas em cenários simulados.

Um ambiente simulado, embora leve em consideração o funcionamento do edifício, não consegue representar de maneira precisa o dinamismo e variações que ocorrem no cenário real. Dessa forma, o comportamento reproduzido nem sempre corresponde às condições reais do ambiente. A inserção de uma camada física com dispositivos reais pode contribuir com essa limitação, permitindo que o sistema seja alimentado com dados em tempo real e possa se autoadaptar às mudanças do ambiente. Por exemplo, situações como em um determinado dia o professor e alunos de uma turma saem da sala em que a aula estaria programada para assistir uma palestra em um auditório.

A arquitetura da solução, conforme apresentado na Figura 4, composta por quatro camadas, (i) Camada Física, (ii) *Middleware*, (iii) Camada de Gerência e (iv) Camada de Planejamento, detalhadas nas seções a seguir.

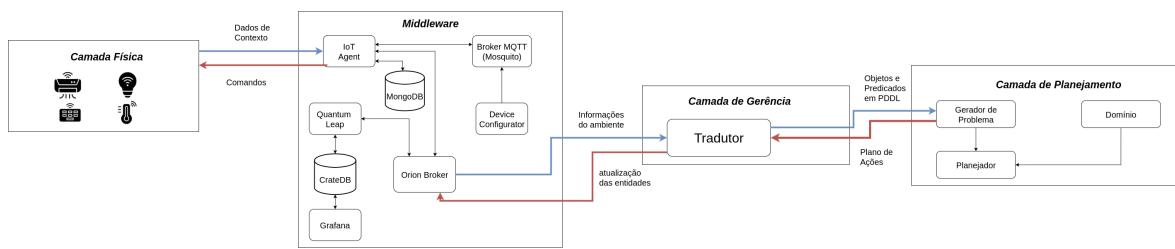


Figure 4. Arquitetura da solução proposta e fluxo de dados.

4.1. Camada Física

Para gerenciamento de equipamentos e sistemas de iluminação e climatização do ambiente real, será implantada uma rede de sensores e atuadores, que serão responsáveis, respectivamente, pela coleta de dados do ambiente e pela realização de ações no mesmo.

Na Figura 5(a) é ilustrado o dispositivo responsável pela medição da temperatura e umidade. Ele utiliza o sensor DHT11 e será responsável por monitorar a temperatura do ambiente, auxiliando no controle do ar-condicionado para mantê-lo sempre em uma temperatura ideal. Para controlar os aparelhos de ar-condicionado e projetores será utilizado o dispositivo da Figura 5(b). Esse controle utiliza um LED Infravermelho (IR), que emite a mesma frequência usada pelos controles remotos convencionais, permitindo ligar/desligar

os aparelhos, assim como ajustar a temperatura do ar-condicionado. A Figura 5(c) apresenta o dispositivo composto por um módulo relé, que atua como uma chave, recebendo comandos para ligar/desligar as lâmpadas. O dispositivo da Figura 5(d) realizará o monitoramento do consumo energético dos equipamentos. Ele possui um sensor de corrente não intrusivo SCT-013-000 que coleta amostras do sinal de corrente. Todos os dispositivos utilizam como microcontrolador a placa ESP-01 e foram desenvolvidos conforme apresentado em [Amurim et al. 2021]. Além disso, foi implementado um dispositivo para contagem de pessoas no ambiente, utilizando dois sensores de distância VL53L0X. Para a transmissão dos dados coletados para o *middleware* os dispositivos utilizam o protocolo MQTT.

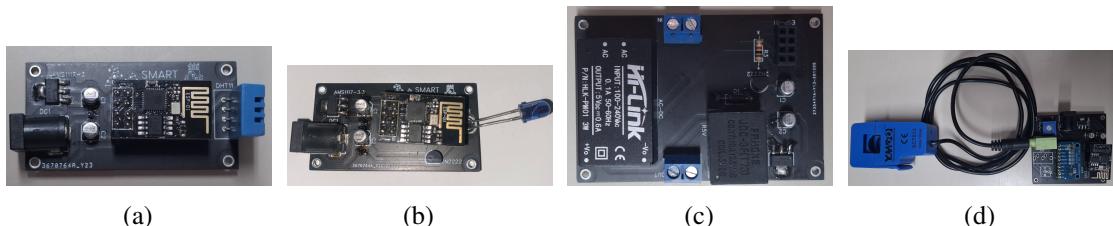


Figure 5. Dispositivos IoT utilizados em uma sala da UFC-Quixadá.

4.2. Middleware

O FIWARE é uma plataforma de código aberto que se destaca pela sua compatibilidade com os principais protocolos utilizados em ecossistemas de IoT [Amurim et al. 2021]. Apresenta uma arquitetura extensível, composta por diversos elementos denominados *Generic Enablers* (GE) que usam o padrão *Next Generation Service Interface* (NGSI-09/NGSI-10) para se comunicarem.

Na arquitetura proposta, o *FIWARE* atua como um intermediário. A Camada Física se conecta ao mesmo por meio do *broker* MQTT, o Mosquito. Após a publicação dos dados de contexto no *broker*, eles são enviados ao **IoT Agent**, que interpreta as mensagens MQTT e as converte em atualizações de contexto, que são armazenadas como entidades no **Orion Broker**. Esse, gerencia todas as informações, fornecendo também requisições HTTP para que esses dados possam ser atualizados e requisitados por outras aplicações.

4.3. Camada de Gerência

A Camada de Gerência irá consumir as informações do ambiente por meio do FIWARE, requisitando os dados ao Orion Broker por meio de requisições HTTP. Nela, um tradutor converterá os mesmos em objetos e predicados no formato PDDL, permitindo a construção do problema de planejamento pela Camada de Planejamento.

4.4. Camada de Planejamento

A Camada de Planejamento é composta por três componentes, conforme ilustrado na Figura 4: (i) Gerador de Problema; (ii) Planejador e (iii) Arquivo de Domínio. O Gerador de Problema é responsável por criar nossas instâncias de problema, que representam uma situação específica do ambiente. A partir dos objetos e predicados em PDDL, recebidos da camada de gerência, um problema é instanciado. Esse problema, juntamente com o

arquivo de domínio, são utilizados como entrada no planejador, o qual é responsável por elaborar um plano de ações.

A Figura 6 apresenta um trecho do domínio modelado, `smart_campus` e a versão completa¹ está disponível para acesso. Para atender as nossas necessidades definimos os predicados, funções e ações. Os predicados são os elementos do domínio que capturam os estados dos dispositivos IoT e condições dos equipamentos físicos (projetores, ar-condicionado e lâmpadas). As funções correspondem as métricas ou quantitativos do ambiente, representados por valores numéricos que são atualizados pelos dados dos sensores instalados ou pelo efeito das ações. Já ações representam as operações que um agente pode executar para alterar o estado do ambiente. Dentro do nosso contexto, correspondem aos comandos que os dispositivos podem executar.

As linhas 4 a 7 descrevem os predicados empregados, entre eles: `device-on`, que indica se um determinado dispositivo (ar-condicionado, lâmpada, projetor) está ligado ou não. A ação durativa nas linhas 11 a 15, denominada `turn_off_light`, possui dois parâmetros: a sala onde o equipamento está instalado (`?r`) e o `light` (lâmpada) que deve ser desligado (`?l`). Para que essa ação seja executada, é necessário que o ambiente esteja desocupado – ou seja, que o número de ocupantes seja igual a zero. Essa condição é verificada por meio da função `occupancy ?r - room` (linha 9). Como efeito da ação durativa, o estado da lâmpada é alterado para desligado.

```

1 (define (domain smart_campus)
2   (:requirements :typing :duractive-actions :numeric-fluents :
3     fluents)
4   (:types device room air-conditioner light projector - device)
5   (:predicates
6     (device-in-room ?r - room ?d - device)
7     (device-on ?r - room ?d - device)
8   )
9   (:functions
10    (occupancy ?r - room)
11    ...))
12   (:durative-action turn_off_light
13    :parameters (?r - room ?l - light)
14    :duration (= ?duration 0.08333)
15    :condition (and (at start (device-in-room ?r ?l)) (at start (
16      device-on ?r ?l)) (at start (= (occupancy ?r) 0)))
17      :effect (at start (not (device-on ?r ?l))))
18    ...)
```

Figure 6. Trecho do domínio `smartcampus` em PDDL.

No sistema proposto, o problema é gerado automaticamente, a partir dos dados coletados dos sensores. Para fins de explicação, modelamos um exemplo do problema de planejamento (Figura 7), correspondente ao ambiente da sala do PET TI (Tecnologia da Informação), localizada no térreo do Bloco 1 no Campus da UFC em Quixadá. A versão completa² está disponível para acesso. Nas linha 3 temos a descrição dos objetos do

¹<https://bit.ly/smartcampusdomain>

²https://bit.ly/smartcampusproblem_b11rpert

problema. Cada objeto definido no problema representa uma entidade física presente na sala, que é vinculado a um dispositivo IoT para ser responsável por seu controle. São eles: na qual temos: 1 ar-condicionado, b11-rpet-ac1, 2 interruptores, representados pelos objetos b11-rpet-11 e b11-rpet-12 e 1 projetor, b11-rpet-p1. Na situação inicial (linhas 5 a 8), temos descrito o estado dos dispositivos, no qual todos estão ligados e alocados na sala, e que a quantidade de pessoas na sala é igual a 1 ($= (\text{ooccupancy } ?\text{b11-rpet}) \ 1$). A seção :goal define o estado desejado e os objetivos a serem alcançados. A meta é que o planejador gere um plano que resulte no valor mínimo para a função total-energy-consumed.

```

1 (define (problem b11_rpet)
2   (:domain smart_campus)
3     (:objects b11-rpet - room b11-rpet-ac1 - air-conditioner
4       b11-rpet-11 b11-rpet-12 - light b11-rpet-p1 - projector ...)
5     (:init
6       (= (ooccupancy b11-rpet) 1)
7       (device-in-room b11-rpet b11-rpet-ac1)
8       ...
9       (= (total-energy-consumed) 0)
10      )
11      (:goal and (not (device-on ?b11-rpet-ac1)) ...))
12    )

```

Figure 7. Exemplo de problema para o domínio *smart_campus*.

4.5. Funcionamento da Arquitetura Proposta

O processo de montagem do problema ocorre desde a camada física. Os dispositivos físicos coletam informações e executam ações. Primeiramente, os sensores coletam informações e as enviam para o *Broker MQTT*, enviando as leituras para seu tópico correspondente. Uma vez publicados, o IoT Agente recebe as mensagens, as converte para o formato NGSI e envia para o Orion Broker. Com isso, os atributos da entidade correspondente são atualizados. Assim que os dados atuais são armazenados no Orion Broker, a Camada de Gerência (CG) requisita as informações por meio de requisições HTTP GET. Após o recebimento desses dados, a CG realiza o processo de traduzi-los em PDDL, transformando-os em predicados e objetos. Cada entidade do FIWARE será mapeada para um objeto em PDDL e cada atributo será convertido para um predicado. Após a tradução dos dados, eles são recebidos pela Camada de Planejamento, por meio do Gerador de Problema. A partir dos objetos e predicados recebidos, que descrevem o estado atual do ambiente, preenche um arquivo em PDDL com as seções :objects, :init, :goal. O arquivo de problema gerado, em conjunto com o arquivo do domínio smart_campus, já definido previamente, são entradas para o Planejador. Esse, explora possíveis ações até encontrar a melhor sequência que satisfaça nosso objetivo.

A saída é um plano de ações gerada pelo planejador escolhido, o *Temporal Fast Downward*, que permite lidar com ações durativas e valores numéricos. A saída obtida são as ações que devem ser executadas no ambiente real para que a meta de minimização do consumo de energia seja alcançada. Para que as ações sugeridas sejam efetuadas no ambiente real, elas precisam chegar ao dispositivo de uma forma que ele compreenda. Após a

elaboração do plano, o mesmo que enviado para a camada de gerência, as traduz cada ação em requisições do tipo PATCH, atualizando os atributos das entidades correspondentes. O *Orion* então notifica o *IoT Agent* da atualização, publicando a mensagem no tópico do dispositivo. Por fim, o mesmo interpreta a mensagem e executa o comando correspondente a mesma. É por meio desse fluxo entre as camadas que todas as ações previstas no plano são efetivamente executadas no ambiente físico. As ações definidas no nível de planejamento vão sendo sucessivamente traduzidas em representações como JSON e mensagens MQTT, até atingirem o formato comprehensível pelos dispositivos reais, que então executam os comandos finais.

5. Conclusão e Trabalhos Futuros

O presente trabalho apresentou a proposta preliminar de integração de dispositivos IoT com planejamento automatizado para o gerenciamento dos recursos energéticos do Campus da UFC em Quixadá, tais como lâmpadas, aparelhos de ar-condicionado e projetores. As principais contribuições deste trabalho foram: (i) proposta de arquitetura que integra a camada física ao sistema de planejamento automatizado; (ii) modelagem do domínio de planejamento *smart_campus*, com predicados e ações durativas para controle dos dispositivos IoT; (iii) modelagem do problema de planejamento referente a uma das salas do Campus; (iv) escolha e execução do planejador temporal capaz de obter planos para domínios com ações durativas(v) instalação física dos sensores e atuadores para testar a viabilidade da arquitetura proposta.

Como trabalhos futuros pretendemos: modelar problemas de planejamento referentes às demais salas do campus (tais como sala de aula, auditório, laboratórios, salas administrativas); instalar sensores e atuadores nas demais salas do campus e avaliar o funcionamento da arquitetura em relação ao consumo total de energia.

Agradecimentos

Agradecemos ao Programa de Educação Tutorial em Tecnologia da Informação da UFC-Quixadá pelo apoio financeiro a este projeto. Agradecemos também ao CNPq, processo 440098/2024-6 pelo apoio financeiro.

References

- Aleesha, M. and Laseena, C. A. (2022). Mqtt protocol for resource constrained iot applications: A review. In *Proceedings of the International Conference on Systems, Energy and Environment 2022 (ICSEE 2022)*. Disponível em: <https://ssrn.com/abstract=4299372> ou <http://dx.doi.org/10.2139/ssrn.4299372>. Acesso em: 21 mai. 2025.
- Amurim, A. D., da Silva, J. I., Ortiz, M. D., Rego, P. A., and de Souza, J. N. (2021). Uma solução de iot baseada no fiware para gerenciamento de recursos energéticos e serviços acadêmicos em um campus universitário. In *Workshop de Computação Urbana (CoUrb)*, pages 265–278. SBC.
- Coles, A., Coles, A., Fox, M., and Long, D. (2009). Temporal planning in domains with linear processes. pages 1671–1676.

- da Conceição, W. N. E. and de Resende Costa, R. M. (2023). Análise do protocolo mqtt para comunicação iot através de um cenário de comunicação. In *Anais da Escola Regional de Redes de Computadores (ERRC)*, Juiz de Fora, MG, Brasil. SBC - Sociedade Brasileira de Computação. ISSN 2595-5911.
- Dias, U., Hell, M., Medeiros, Á., Silveira, D., and de Aguiar, E. (2019). Self-organizing fuzzy rule-based approach for dealing with the classification of indoor environments for iot applications. In *Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)*, pages 1044–1055. SBC.
- ENGIE (2024). Anuário estatístico de energia elétrica. Disponível em: <https://www.engie.com.br/>. Acesso em: 19 mai. 2025.
- Fox, M. and Long, D. (2003). Pddl2.1: An extension to pddl for expressing temporal planning domains. *J. Artif. Intell. Res. (JAIR)*, 20:61–124.
- Georgievski, I., Schlatow, J., and Aiello, M. (2023). Ai temporal planning for energy smart buildings. *Energy Informatics*, 6(1):1–17.
- Ghallab, M., Nau, D., and Traverso, P. (2004). *Automated Planning: theory and practice*. Elsevier.
- Haslum, P., Lipovetzky, N., Magazzeni, D., Muise, C., Brachman, R., Rossi, F., and Stone, P. (2019). *An introduction to the planning domain definition language*, volume 13. Springer.
- Ma, J., Bouloukakis, G., Kattepur, A., Yus, R., and Conan, D. (2023a). Demsa: a dt-enabled middleware for self-adaptive smart spaces. In *Proceedings of the 1st International Workshop on Middleware for Digital Twin*, pages 1–6.
- Ma, J., Bouloukakis, G., Kattepur, A., Yus, R., and Conan, D. (2023b). DEMSA: a DT-enabled middleware for self-adaptive smart spaces. In *Midd4DT '23: Proceedings of the 1st International Workshop on Middleware for Digital Twin*, pages 1–6, Bologna, Italy. ACM/IFIP.
- Mesquita, R. P., Teixeira, E. H., Ribeiro, A. J., Gandolpho, B. D., and Pontes, F. L. F. (2024). Computer vision applied to smart markets: a case study for empty shelf detection. In *Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)*, pages 460–471. SBC.
- Rio, L. S. D., Costa, M. R., and Silva, J. A. (2018). Proposta de ambientes inteligentes iot sob a ótica da eficiência energética. In *Encontro Anual de Tecnologia da Informação*, pages 122–130. SBC.
- Rojek, I., Mikołajewski, D., Mroziński, A., Macko, M., Bednarek, T., and Tyburek, K. (2025). Internet of things applications for energy management in buildings using artificial intelligence—a case study. *Energies*, 18(7):1706.
- Russell, S. J. and Norvig, P. (2004). *Inteligência artificial*. Elsevier.
- Universidade Federal do Ceará (2022). Superintendência de infraestrutura e gestão ambiental - Água e energia. Disponível em: <https://ufcinfra.ufc.br>. Acesso em: 19 mai. 2025.