

Plataforma Open-hardware de baixo custo para teste de dispositivos IoT voltados para Smart Cities

Diógenes V. M. Ferreira^{1,2,3}, Vitor N. Coelho^{3,4}, and Sidelmo M. Silva⁵

¹ Programa de Pós-Graduação em Engenharia Elétrica - Universidade Federal de Minas Gerais - Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil

² Laboratório de Automação Predial - Departamento de Controle e Automação - Escola de Minas - Universidade Federal de Ouro Preto, Ouro Preto, MG, Brasil

³ Grupo da Causa Humana, Ouro Preto, Brazil

⁴ Instituto de Computação, Universidade Federal Fluminense, Niterói, Brasil

⁵ Departamento de Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil

viegas.diogenes@gmail.com

Resumo Em todo o mundo, empresas de diversas áreas de tecnologia têm como um dos focos de suas pesquisas a Internet das Coisas. Muitos pesquisadores acreditam que ela irá trazer ao mundo contemporâneo uma nova revolução da informação, estabelecendo novos protocolos e paradigmas de comunicação. A ideia principal deste paradigma é que todos os objetos estarão conectados à internet possibilitando monitoramento e controle, além da interação entre eles por meio do recebimento e envio de informação. Dentro deste conceito abrangente, situa-se outro que também vem recebendo considerável visibilidade, as *Smart Cities*. Uma “Cidade Inteligente” é aquela em que objetos, lugares, equipamentos e, até mesmo, seres vivos estão interconectados de alguma forma, interagindo de forma distribuída e autônoma, monitorando, armazenando, tratando e otimizando dados de forma a deixar espaços compartilhados mais confortáveis, seguros e sustentáveis. Toda essa “inteligência” embarcada e todo esse tráfego de dados necessitam de um processamento de uma quantidade enorme de informação e consomem muita energia para funcionar. Com isso, torna-se extremamente necessário que o desenvolvimento de novas tecnologias de *hardware* acompanhe essa demanda de dispositivos de baixo consumo. Além disso, *hardwares* especializados, como o proposto nesse trabalho, são de grande importância para atingir esses propósitos requisitados. Este trabalho faz um aprofundamento no conceito da Internet das Coisas, focando no desenvolvimento de *Smart Cities* e em seguida, apresenta uma proposta de uma plataforma de *hardware* que possibilita o desenvolvimento de dispositivos para *Smart Cities*. O arcabouço apresentado não se atém a apenas um protocolo de comunicação, visto que um protocolo universal ainda não foi definido.

Keywords: Smart Cities, IoT, Sistemas Embarcados, Hardware

1 Introdução

Com o desenvolvimento da computação móvel, das comunicações sem fio e o aumento da complexidade dos equipamentos eletrônicos, vem ganhando cada vez mais espaço no campo da ciência da computação um novo paradigma chamado Internet das Coisas (*IoT – Internet of Things*) [1,2,3]. Esse conceito vem também sendo denominado como Internet de Tudo, do inglês *Internet of Everything* [4,5,6]. De forma comum, essas tecnologias promovem uma forma de distribuir, ao nosso redor, coisas e objetos diversos conectados entre si ou até mesmo na internet, com a capacidade de interagir e cooperar com vizinhos. Os fins de cada agente/dispositivo é distinto, alguns são egoístas, otimizando seus próprios interesses [7]; enquanto outros atuam com o fim de alcançar um objetivo comum. O mar de informações oriundo de sensores auto-configuráveis [8,9] poderá ser rotulado, processado e transmitido. Essas informações poderão melhorar a condição de nossas vidas e facilmente resolver problemas do cotidiano [10], principalmente, com o auxílio de técnicas da Inteligência Artificial (AI) e tomadas de decisão sábias, inspiradas em estratégias da Pesquisa Operacional (PO). Do mesmo modo, todas as “coisas” conectadas à rede se tornarão participantes ativos do ambiente onde forem instalados, possuindo as seguintes habilidades:

- Reagir de forma autônoma segundo dados de sensoriamento e Inteligência Artificial embarcada;
- Analisar dados oriundos de outros dispositivos ligados à rede e utilizar-se deles para interferir no meio, modificar parâmetros internos, comparar com dados internos e oferecer uma resposta;
- Analisar comportamento e padrões tornando seu funcionamento cada vez mais eficiente e inteligente;
- Receber intervenções humanas por meio de “centrais de comando” a fim de coordenar suas funções com necessidades dinâmicas;

Dessa forma, mais do que estarem conectados à internet, os dispositivos estão começando a conversar entre si e tomar decisões sem necessariamente exigir a interferência humana. Neste cenário surge a possibilidade de comunicações “*Machine to Machine*” (M2M), “*Human to Machine*” (H2M) e “*Machine in Humans*” (MiH) [11,12]. A figura 1 apresenta uma demonstração visual desses tipos de comunicação.

Segundo Chourabi et al., o desenvolvimento das *Smart Cities* (SC) será necessário para se mitigar os problemas causados pelo grande aumento da população urbana, que vem acontecendo não só no nosso país, mas no mundo inteiro [14]. O relatório de prospecção de urbanização da ONU mostra que este processo de urbanização tende a continuar acelerado pelos próximos anos[15]. Devido a esse crescimento acelerado, cidades antes planejadas para atender um número de cidadãos terão agora que acomodar um contingente ainda maior de pessoas.

Apesar de todo avanço existente na área de computação e sistemas embarcados, a Internet das Coisas e os consequentes sonhos idealizados para as SC

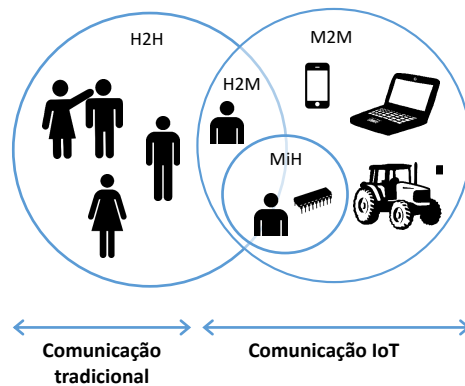


Figura 1. Tipos de comunicação possíveis dentro da IoT. - Adaptado [13].

enfrentam um problema aparentemente simples. Essa barreira, que limita alguns sistemas de serem integrados à rede, é a falta de padronização de um protocolo de comunicação de dispositivos para *IoT*. Nesse contexto, essa carência de consenso atrasa o desenvolvimento de dispositivos que possam atender às demandas de cidades do futuro. Visando contornar este problema, neste trabalho, propõe-se uma plataforma de *hardware* que funciona em diversos tipos de protocolos de comunicação sem fio. Em especial, busca-se conceber um arcabouço capaz de testar e desenvolver dispositivos para internet das coisas, independentemente do protocolo de comunicação desejado. Além disso, contribui-se para a literatura apresentando, em detalhes, a placa de circuito impresso (PCI) desenhada e implementada, a qual possibilita a conexão de diferentes tipos de sensores para que diversas soluções possam ser testadas. A plataforma desenvolvida, é de baixo custo e possui o potencial de substituir os kits de testes de fabricantes (que muitas vezes são caros, principalmente no Brasil). Ressaltam-se as seguintes contribuições deste presente trabalho:

- Discussão de novos paradigmas de comunicação nas cidades inteligentes, por meio de dispositivos embutidos com *IoT*;
- Concepção de uma nova placa para testes em ambientes interconectados:
 - Plataforma híbrida capaz de suportar distintos protocolos de comunicação do estado-da-arte;
 - Possível arcabouço flexível, robusto e de baixo custo para ser utilizado em cidades e redes inteligentes;

O restante do trabalho foi dividido como mostrado a seguir. Na Seção 2, apresenta-se a interligação entre *IoT* e o conceito de *Smart Cities*. É feita uma breve introdução dos conceitos e o estado-da-arte do cidades inteligentes e *IoT*. A Seção 3 detalha um exemplo que motiva o avanço de dispositivos unificados. Para tal, exaltamos uma placada que poderia ser utilizada para testes de redes de

comunicação alinhadas com os conceitos requisitados pelas SC. Tal dispositivo poderia ser uma base sólida para embarcar técnicas de inteligência computacional e o paradigma dos sistemas multi-agentes. Finalmente, a Seção 4 apresenta as conclusões obtidas na pesquisa e no desenvolvimento deste trabalho, bem como, aponta diretrizes futuras.

2 Smart Cities e a IoT

A aplicação do paradigma *IoT* no contexto urbano para solucionar problemas de gerenciamento de serviços urbanos é que nasce um dos conceitos de *Smart City* [16]. Mesmo que não exista ainda uma definição exata de “*Smart City*”, é sabido que tal conceito foi inicialmente difundido nas áreas tecnológicas. Devido a grande acatamento social, distintas áreas vêm discutindo o escopo deste amplo tema, que envolve, sem dúvidas, inteligência de dispositivos e melhor planejamento urbano e social. Um dos objetivos principais é se fazer um melhor uso dos recursos públicos a fim de se melhorar os serviços oferecidos aos cidadãos, enquanto custos operacionais por parte da administração pública e impactos ambientais são reduzidos [17]. Assim sendo, a tecnologia aplicada a sistemas *IoT* pode ser capaz de oferecer ferramentas aos planejadores urbanos com o objetivo de apoiar a melhoria contínua dos espaços urbanos.

Uma das maiores contribuições para o crescimento do Mercado de *IoT* é a crescente demanda por cidades inteligentes em uma escala global. Existem atualmente mais de 200 projetos de cidades inteligentes ao redor do mundo, que produzem oportunidades gigantescas para vendedores de *IoT* e companhias de consultoria da área.

De fato, a *IoT* abre um leque de opções gigantesco para aplicações em *Smart Cities* [3]. São aplicações desde o monitoramento em tempo real da qualidade da água dos rios e fontes que abastecem uma determinada região à sistemas de controle de iluminação pública. Escolar e colaboradores propuseram um sistema de controle adaptativo em fontes de luz e conseguiram uma economia de energia de 35% [18].

Kevin Ashton, co-fundador e diretor executivo do Centro de Auto-ID no MIT, cunhou o termo Internet das Coisas [19,20] e o mencionou pela primeira vez em uma apresentação que ele fez para a *Procter & Gamble* em 1999. Segundo ele, praticamente todos os dados disponíveis na internet até aquele momento teriam sido capturados e criados por seres humanos de forma manual, e que por a raça humana não ser boa em captar dados do mundo real por limitação de tempo, atenção e precisão, o ideal seria que se tivesse computadores que aprendessem tudo que existe para saber sobre as coisas. Nesse sentido, os equipamentos capturariam dados de forma autônoma e poderiam dar informações sobre todos os dispositivos na rede, assim, seria possível se identificar quando eles precisariam de substituição, reparação ou *recall* e se eles já teriam “dado o melhor de si” [21].

Desta forma, nota-se que a ideia de *IoT* se iniciou há um bom tempo. Nos anos 90, tecnologias como RFID, sensores e algumas inovações *wireless* leva-

ram a algumas aplicações conectando dispositivos e “coisas”. A maioria das aplicações de RFID, naquela época, envolviam logística, depósitos e as redes de fornecedores em geral. Apesar das dificuldades na época, o uso de RFID (e junto com ele, outras tecnologias sem fio, como NFC ou “*Near Field Communication*”), tornaram-se populares em áreas além de logística e gerenciamento de rede de fornecedores, desde: transporte público; identificação (de animais de estimação para as pessoas); controle e pagamento automático de pedágios; controle de acesso e autenticação; monitoramento de tráfego; varejo; até, naquela época, formas inovadoras de publicidade *outdoor*. Este crescimento da utilização da tecnologia foi, entre outros, devido à redução do custo das *tags* e da padronização de protocolos. Com o advento da tecnologia e o surgimento de microprocessadores cada vez mais velozes, consumindo cada vez menos energia [22] e custando cada vez menos, a *IoT* começou a se tornar algo mais complexo. Sistemas com elevadíssima capacidade de processamento de dados começaram a surgir, trazendo consigo um novo universo de possibilidades, aparentemente, sem limites para a *IoT*. Dessa forma, mais do que estarem conectados à internet, os dispositivos estão começando a conversar entre si e tomar decisões sem necessariamente exigir a interferência humana [12].

2.1 Unificando as cidades com protocolo e padronizações

Com tantas empresas trabalhando em diferentes produtos, tecnologias e plataformas, fazer com que esses dispositivos comuniquem entre si não será uma tarefa fácil e a compatibilidade mundial muito provavelmente não ocorrerá. Diversos grupos estão trabalhando para criar um protocolo aberto que possa permitir a interoperabilidade entre diversos produtos. Entre eles estão *AllSeen Alliance* que possui *Qualcomm*, *LG*, *Microsoft*, *Panasonic* e *Sony* como membros e o *Open Interconnect Consortium (OIC)*, que possui apoio da *Intel*, *Cisco*, *GE*, *Samsung* e *HP*. Outros protocolos já difundidos também se categorizam dentro da *IoT* como *Wi-Fi*, *LE Bluetooth*, *NFC*, *RFID* que são largamente utilizados em produtos de consumo e outros como *ZigBee*, *Z-Wave* e *6LoWPAN* que vem sendo utilizados em aplicações industriais e prediais. Todavia, ainda não está claro quem irá vencer essa “batalha” de protocolos, mas muitos acreditam que vamos acabar com três ou quatro protocolos diferentes ao invés de um único como já ocorre em diversos outros setores de tecnologia. Como o caso das placas *embedded linux* que estão sendo fabricadas (*Raspberry Pi*, *BeagleBoard*, *Intel Galileo*, entre outras) e embutidas com distintos protocolos. Soluções de equipamentos que se comuniquem por mais de um protocolo podem vir a surgir, mas nesse meio tempo o consumidor terá de escolher um fabricante e se limitar aos seus produtos e outros que se comuniquem com eles. Vermesan e Friess, discursam sobre diversos padrões e protocolos que estão sendo desenvolvidos, dentre eles, CEN/ISO e CENELEC/IEC, ETSI, IEEE, IETF, ITU-T, OASIS, OGC, oneM2M e o GSI [23].

Neste cenário diversificado em que cada desenvolvedor e seus apoiadores defendem a supremacia de suas propostas de padronização da comunicação de dispositivos, muitos irão se destacar seja por funcionalidade, custo ou eficiência.

Por outro lado, como muitos ainda estão em processo de desenvolvimento, ainda não se pode afirmar qual será o protocolo definido para os dispositivos inteligentes.

2.2 Um mar de informações e big-data

Um dos problemas enfrentados por grande parte dos desenvolvedores de protocolos de comunicação em *IoT* é estabelecer a interoperabilidade entre os diferentes dispositivos presentes na rede ao mesmo passo que possibilita seu acesso através de um *IPv6* na nuvem. Visando ultrapassar esta barreira, parcerias vêm sendo feitas entre diferentes protocolos a fim de se chegar a um consenso, no que se diz respeito a uma linguagem de comunicação padrão universal para *IoT*.

Tal ponto afeta diretamente as SC, visto que uma grande gama de dispositivos estão se comunicando e atuando. Para concretizar aplicações reais e que operem de forma segura e efetiva faz-se necessário melhores padronizações e protocolos.

Uma grande parceria foi feita em 2015 entre a *ZigBee Alliance* [24], um dos padrões que domina hoje o mercado de *IoT* e a *Thread Group*, um padrão considerado promissor entre os especialistas que roda sobre o protocolo de rede *IPv6* e um padrão de rede aberto (*Open standard networking protocol*) para redes 802.15.4 de baixo consumo que pode se conectar a centenas de dispositivos ou diretamente à nuvem de forma simples e segura [25]. Esta parceria que foi definida como *ZigBee sobre Thread (ZigBee over Thread)* visava solucionar os problemas de interoperabilidade entre dispositivos ao mesmo passo que facilitava a comunicação deste tipo de rede com a nuvem onde sistemas poderiam se comunicar através da linguagem *ZigBee* em uma rede *Thread* [26].

Outro grande problema é a definição de um padrão de perfis de dispositivos, ou seja, padronizar de forma universal as características e funções de diferentes dispositivos. A *ZigBee Alliance* criou o que chamaram de *ZigBee Cluster Library*, que nada mais é que uma biblioteca de perfil de dispositivos que usam o padrão *ZigBee*.

Ashton, apesar de mostrar em seu artigo que ainda não existe um padrão de comunicação acordado, afirma que a convergência da *IoT* está próxima [27]. No dia 3 de Janeiro deste ano, na maior mostra e conferência internacional de produtos eletrônicos focados ao consumidor (CES 2017), a *Zigbee Alliance* apresentou um novo padrão que clama ser a linguagem universal para *IoT* [28]. Levando o nome de *Dotdot*, essa nova linguagem para *IoT* que trabalha em conjunto com a rede IP da *Thread* promete aliar todas os requisitos necessários para atender às demandas atuais e futuras da *IoT* [29]. É muito provável que teremos em um futuro próximo, soluções e trabalhos científicos em *IoT* que se baseiem nestas tecnologias.

Limites para aplicações de *IoT* para *Smart Cities* parecem não existir. Nos próximos anos são esperadas infinitas soluções que prometem aumentar a segurança, melhorar a qualidade de vida e promover o consumo de energia mais consciente em nossas cidades.

3 Exemplificando o potencial das pequenas aplicações

Os conceitos apresentados e enfatizados neste trabalho exaltam um dos problemas atualmente enfrentados por parte dos desenvolvedores de sistemas. Devido a dificuldade para chegar-se a um consenso de um protocolo de comunicação universal, esta seção motiva estudos, avanço e desenvolvimento de uma plataforma de *hardware*. O arcabouço apresentado, apesar de não ser muito complexo, apresenta-se como uma solução dinâmica no que diz respeito ao protocolo de comunicação e sensores/atuadores a serem utilizados em:

- Novos dispositivos para as cidades inteligentes:
 - Iluminação pública; Controle de Tráfego; Controle de Qualidade do Ar; entre outros.
- Novos dispositivos para as redes inteligentes:
 - Comunicação entre fontes de energia renovável ou entre consumidores, em sistemas *microgrid*; Mercado de venda e troca de energia; entre outros.

Com relação ao protocolo de comunicação, os autores se basearam na plataforma de módulos de comunicação *thru-hole* da empresa DIGI chamada de *XBee* [30]. O motivo de tal escolha se deve ao fato de que esta empresa mantém uma mesma plataforma de *hardware* para todos os seus módulos de comunicação sem fio, e com isso, o *hardware* proposto suporta qualquer módulo deste fabricante, podendo assim ser usado para testar diversas redes apresentadas na Figura 2 e também uma rede *Bluetooth*, utilizando módulo de outros fabricantes que se baseiam na plataforma da *DIGI*.

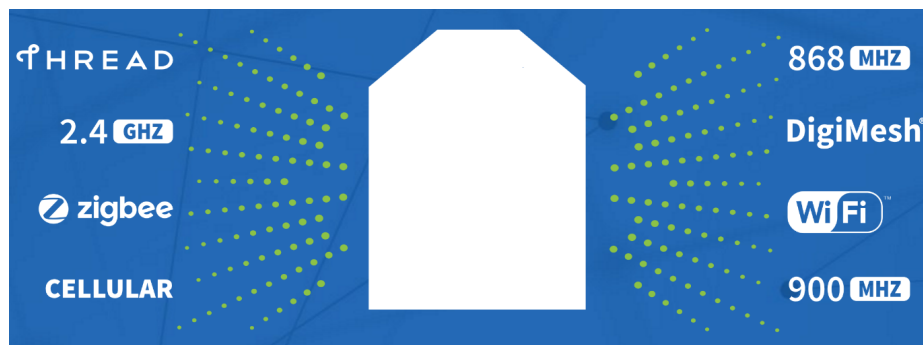


Figura 2. Módulo *Xbee* - Adaptado DIGI

A figura 3 apresenta um diagrama mostrando os módulos do sistema proposto e a forma em que eles estão interligados. Nela pode ser observado que é possível a conexão de diferentes sensores a serem utilizados em inúmeras aplicações.

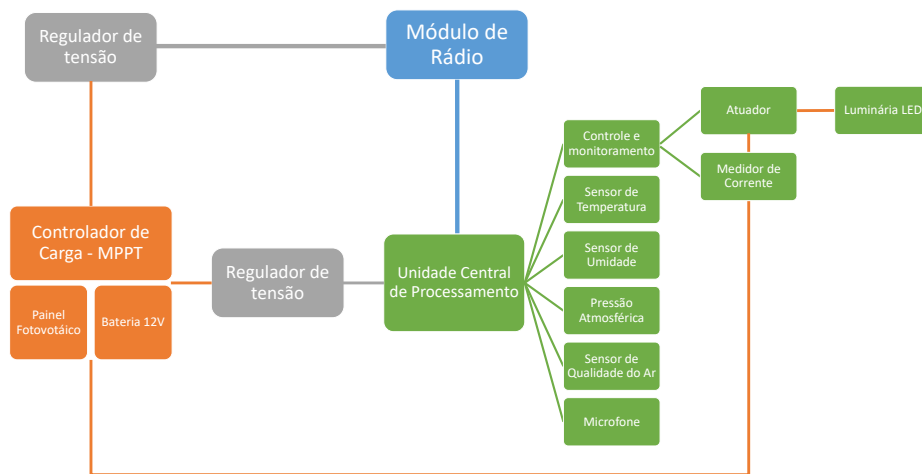


Figura 3. Estrutura do sistema proposto

A Figura 4 mostra um modelo em 3D do aspecto final da placa desenvolvida, já conectada em um módulo da empresa DIGI e um diagrama apresentando os diferentes protocolos que podem ser testados nela. Já a Figura 5 mostra o circuito final desenvolvido no Laboratório de Automação Predial (LAP), da Universidade Federal de Ouro Preto.

A PCI proposta ⁶ possui dois barramentos de extensão de 6 pinos cada, separados em 22mm que possibilitam a ligação de uma outra PCI contendo sensores, atuadores ou até mesmo microcontroladores a serem testados antes de se desenvolver o protótipo final de algum dispositivo a ser produzido. Estes barramentos de extensão fornecem alimentação de 5V/3.3V; conexão com entradas e saídas digitais do módulo Xbee e portas de comunicação serial. Com isso, a plataforma proposta é dinâmica e flexível tanto no sentido de protocolo de comunicação como na sua função/aplicação em *IoT*.

A PCI é composta por diversos componentes eletrônicos que fazem a interface entre o módulo a ela ligado, a fonte de alimentação e a placa de extensão conectada nos barramentos. A interface com a fonte de alimentação é feita através de 2 reguladores de tensão que regulam uma tensão de entrada em corrente contínua entre 7,5-35V para 3,3V e 5V. Também estão colocados na placa alguns LED's, botões e potenciômetros que possibilitam o teste de portas I/O (entrada e saída) analógicas e digitais do módulo. Já os barramentos de 6 pinos, estão conectados diretamente aos módulos, possibilitando assim a conexão de uma infinidade de sensores, atuadores e microcontroladores ao módulo tendo a sua funcionalidade modificável. O CI'S U1 e U2 são reguladores de tensão da *Texas Instruments*,

⁶ Para acesso ao modelo realístico 3D e o diagrama eletrônico completo acesse https://www.dropbox.com/sh/vaio0t1so40tn1e/AABPKvPLVSVjdMm_IG0rhqL0a?dl=0

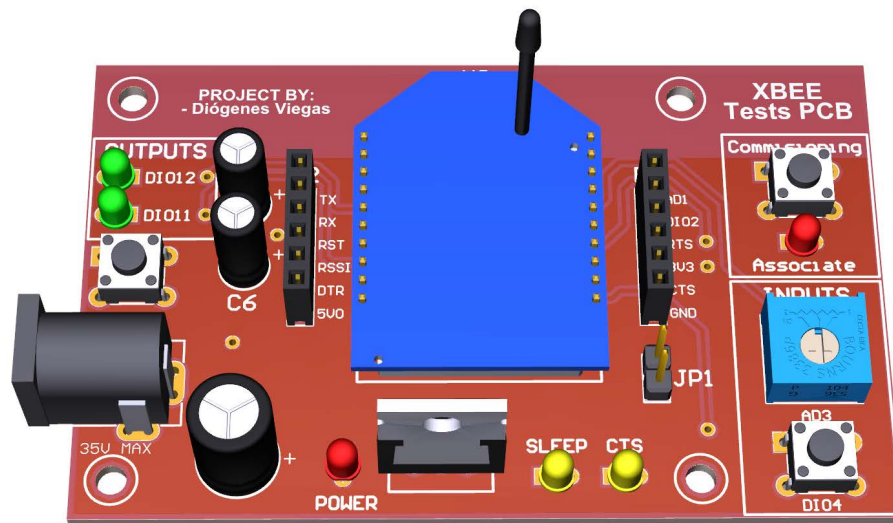


Figura 4. Modelo 3D realístico da PCI

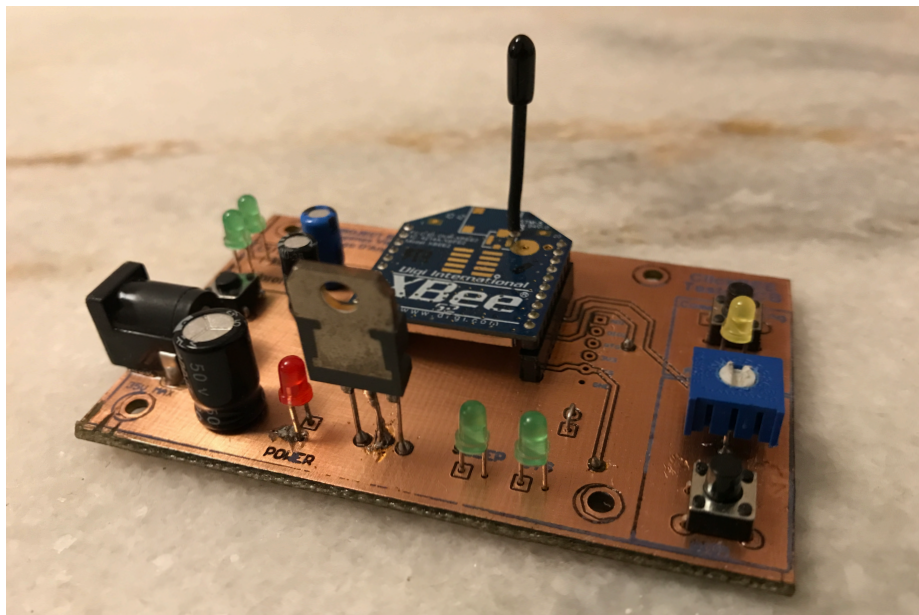


Figura 5. Sistema montado

sendo o primeiro o LM7805 que possui tensão de saída de $5 \pm 0.1V$ e o segundo o LM2937-3.3 que possui tensão de saída de $3.3 \pm 0.1V$, estas tensões são sufici-

entes para alimentar tanto o módulo conectado à PCI, quanto à placa extensora contento os outros dispositivos.

Por ser um sistema flexível, ou seja, ter a capacidade de se conectar à diversos módulos de comunicação sem fio e também possibilitar a conexão de placas de expansão, a plataforma proposta oferece a possibilidade de uso de sistemas para aplicações mais robustas e de alto desempenho. Com isso, podem ser ligados a ela microprocessadores de alta performance ou até mesmo GPU's para um possível processamento de dados *in locu* e em tempo real para diversas funções.

Apesar de não ter um nível de complexidade muito elevado, a plataforma proposta foi desenvolvida para servir de ferramenta dinâmica para testes de diferentes dispositivos e aplicações para sistemas baseados em *IoT* para serem utilizados no ambiente de *Smart Cities*. O fabricante já oferece uma plataforma similar, mas o custo elevado principalmente no Brasil, motivou o grupo a desenvolver o próprio sistema de baixo custo.

4 Considerações finais e possíveis extensões

Neste trabalho, foi abordado o conceito de Internet das coisas e sua relação com Cidades Inteligentes. Essa nova tendência, a de interconectar as “coisas”, e fazê-las trocarem informações, se torna essencial para a existência de um sistema inteligente em que máquinas fazem todo o trabalho do homem com o objetivo de tornar o ambiente em que ele vive, mais seguro, mais agradável e muito mais sustentável. Novas tecnologias de sistemas embarcados com funcionalidades em *IoT* vêm sendo desenvolvidas dentro dos padrões em que os fabricantes fazem parte, e com isso a intercomunicabilidade entre dispositivos se torna impossível. Visando obter uma plataforma de *hardware* que possibilite o teste de dispositivos *IoT* sem se ater a um determinado tipo de protocolo, um simples estudo de caso exaltou um *hardware* que pode ser testado em diferentes tipos de comunicação. Tal dispositivo motiva a unificações do protocolos, em especial, no contexto das cidades inteligentes.

Com mais iniciativas que seguissem o âmbito desse trabalho poderia-se evitar uma rápida inutilização dos sistemas que vêm sendo propostos, visto que um protocolo universal (ou mais) serão definidos. Mesmo que tal fato não ocorra, a compatibilidade deve ser um dos objetivos centrais para o desenvolvimentos de dispositivos que sejam compatíveis com as nossas cidades do futuro.

De uma forma geral, o sistema proposto mostra-se como uma interessante solução para testes de rede e sensores diversos para aplicações em um ambiente de cidade inteligente. Com o objetivo de se entender melhor a dinâmica de uma cidade e atuar de forma a otimizar serviços e propor novas soluções, um sistema deste tipo serve como base para evolução no desenvolvimento de um sistema comercial *IoT* para *Smart Cities*.

5 Agradecimentos

Os autores gostariam de agradecer à Universidade Federal de Ouro Preto (UFOP), e ao seu Laboratório de Automação Predial (LAP) do Departamento de Controle e Automação e Técnicas Fundamentais (DECAT) , pela infraestrutura indispensável para realização deste trabalho. Também gostariam de agradecer à Fundação Gorceix, pelo apoio financeiro.

Referências

1. L. Tan, N. Wang, Future internet: The internet of things, in: *Advanced Computer Theory and Engineering (ICACTE)*, 2010 3rd International Conference on, Vol. 5, IEEE, 2010, pp. V5–376.
2. J. M. Hernández-Muñoz, J. B. Vercher, L. Muñoz, J. A. Galache, M. Presser, L. A. H. Gómez, J. Pettersson, Smart cities at the forefront of the future internet, in: *The Future Internet Assembly*, Springer, 2011, pp. 447–462.
3. A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, Internet of things for smart cities, *IEEE Internet of Things journal* 1 (1) (2014) 22–32. doi:10.1109/JIOT.2014.2306328.
4. R. Y. Clarke, Smart cities and the internet of everything: The foundation for delivering next-generation citizen services, Alexandria, VA, Tech. Rep.
5. S. Mitchell, N. Villa, M. Stewart-Weeks, A. Lange, The internet of everything for cities, *Connecting People, Process, Data, and Things to Improve the ‘Livability’ of Cities and Communities*, Cisco.
6. A. Majeed, Developing countries and internet-of-everything (ioe), in: *Computing and Communication Workshop and Conference (CCWC)*, 2017 IEEE 7th Annual, IEEE, 2017, pp. 1–4.
7. D. Weyns, A. Omicini, J. Odell, Environment as a first class abstraction in multi-agent systems, *Autonomous Agents and Multi-Agent Systems* 14 (1) (2007) 5–30. doi:10.1007/s10458-006-0012-0.
8. M. Batty, Big data, smart cities and city planning, *Dialogues in Human Geography* 3 (3) (2013) 274–279.
9. R. Kitchin, The real-time city? big data and smart urbanism, *GeoJournal* 79 (1) (2014) 1–14.
10. O. Vermesan, European research cluster on the internet of things-outlook of iot activities in europe.
11. J. Poncela, J. Moreno, M. Aamir, Analysis of m2m capabilities in 4g, in: *Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE)*, 2014 4th International Conference on, IEEE, 2014, pp. 1–5.
12. R. Minerva, A. Biru, D. Rotondi, Towards a definition of the internet of things (iot), *IEEE Internet Initiative* (1).
13. D. Minoli, *Building the internet of things with IPv6 and MIPv6: The evolving world of M2M communications*, John Wiley & Sons, 2013.
14. H. Chourabi, T. Nam, S. Walker, J. R. Gil-Garcia, S. Mellouli, K. Nahon, T. A. Pardo, H. J. Scholl, Understanding smart cities: An integrative framework, in: *System Science (HICSS)*, 2012 45th Hawaii International Conference on, IEEE, 2012, pp. 2289–2297. doi:10.1109/HICSS.2012.615.

15. ONU, World urbanization prospects: The 2014 revision, Department of Economic and Social Affairs, UN.
16. H. Schaffers, N. Komninos, M. Pallot, B. Trousse, M. Nilsson, A. Oliveira, Smart cities and the future internet: Towards cooperation frameworks for open innovation, in: The Future Internet Assembly, Springer, 2011, pp. 431–446. doi:10.1007/978-3-642-20898-0_31.
17. M. Peris-Ortiz, D. R. Bennett, D. P.-B. Yábar, Sustainable smart cities doi:10.1007/978-3-319-40895-8.
18. S. Escolar, J. Carretero, M.-C. Marinescu, S. Chessa, Estimating energy savings in smart street lighting by using an adaptive control system, International Journal of Distributed Sensor Networks doi:10.1155/2014/971587.
19. K. Ashton, That ‘internet of things’ thing, RfID Journal 22 (7) (2009) 97–114.
20. S. Madakam, R. Ramaswamy, S. Tripathi, Internet of things (iot): A literature review, Journal of Computer and Communications 3 (05) (2015) 164.
21. T. Guardian, The internet of things is revolutionising our lives, but standards are a must — media network — the guardian, <https://www.theguardian.com/media-network/2015/mar/31/the-internet-of-things-is-revolutionising-our-lives-but-standards-are-a-must>, (Acessado em 29/01/2017) (Março 2015).
22. D. B. Kirk, W. H. Wen-mei, Programming massively parallel processors: a hands-on approach, 2nd Edition, Morgan Kaufmann, 2012.
23. O. Vermesan, P. Friess, Internet of things - from research and innovation to market deployment, River Publishers Aalborg, 2014.
24. Zigbee, zigbee alliance, <http://www.zigbee.org/>, (Acessado em 20/03/2017).
25. T. Group, Zigbee alliance and thread group successfully demonstrate products running zigbee’s universal language for smart devices on thread networks ; thread group, <https://threadgroup.org/news-events/press-releases/ID/140/ZigBee-Alliance-and-Thread-Group-Successfully-Demonstrate-Products-Running-ZigBees-Universal-Language-for-Smart-Devices-on-Thread-Networks>, (Acessado em 20/02/2017) (Dezembro 2016).
26. J. Jacobson, Zigbee teams with thread group on ipv6 home automation protocol - ce pro, http://www.cepro.com/article/zigbee_teams_with_thread_group_on_ipv6_home_automation_protocol, (Acessado em 20/02/2017) (Abril 2015).
27. S. Ashton, Iot convergence is coming — EE times, http://www.eetimes.com/author.asp?section_id=36&doc_id=1331309, (Acessado em 20/02/2017) (Fevereiro 2017).
28. ZigBee-Alliance, The zigbee alliance to unveil universal language for the iot from ces 2017, <http://www.zigbee.org/the-zigbee-alliance-to-unveil-universal-language-for-the-iot-from-ces-2017-making-it-possible-for-smart-objects-to-work-together-on-any-network/>, (Acessado em 20/02/2017) (Janeiro 2017).
29. P. Gutierrez, Zigbee alliance launches new iot language - news - iot hub, <https://www.iothub.com.au/news/zigbee-alliance-launches-new-iot-device-language-446800>, (Acessado em 20/02/2017) (Janeiro 2017).
30. DIGI, M2m communications, remote monitoring & management - digi international, <https://www.digi.com/>, (Acessado em 20/03/2017).

Pesquisa Atual e Futura no Laboratório de Sistemas de Computação da UFSM: Soluções para Cidades Inteligentes

Marcia Pasin¹, Liza Lunardi Lemos², Rafael Fao de Moura¹,
Ricardo Silveira Rodrigues¹, Emmanuel Katende Dinanga³,
Eric Thomas Zancanaro¹, Marina Laisa Mota da Silva¹, Felipe Silvano Perini⁴,
Cleandro Flores De Gasperi¹, Leonardo de Abreu Schmidt¹,
Ezequiel Rodrigues Ribeiro¹

¹ Universidade Federal de Santa Maria

Av. Roraima, 1.000 - Cidade Universitária, Santa Maria - RS, 97105-900

² Pós-Graduação em Ciência da Computação

Universidade Federal do Rio Grande do Sul

Avenida Bento Gonçalves, 9500 - Agronomia, Porto Alegre - RS, 91509-900

³ Pós-Graduação em Engenharia de Automação e Sistemas

Universidade Federal de Santa Catarina

Bairro Trindade, Caixa Postal 476, Florianópolis - SC, 88040-900

⁴ Pós-Graduação em Engenharia de Software

Instituto de Gestão e Tecnologia da Informação

Complexo Paisagem - Rua Roma, 26 - Santa Lucia, Belo Horizonte - MG, 30360-680

Resumo Este texto relata trabalhos desenvolvidos e em desenvolvimento no Laboratório de Sistemas de Computação (LSC) da Universidade Federal de Santa Maria (UFSM) para a implementação de soluções para o contexto de Cidades Inteligentes. Os projetos tem sido desenvolvidos desde 2010 pelos alunos dos cursos de bacharelado em Ciência da Computação e Sistemas de Informação, e de mestrado do Programa de Pós-Graduação em Informática da UFSM. Mais precisamente, neste texto são enfocados projetos nos eixos de sistemas de transportes inteligentes, infraestruturas para cidades inteligentes e automação residencial. Os eixos são explorados de duas formas: simulação computacional e prototipação. Os trabalhos que envolvem prototipação foram desenvolvidos na plataforma de *software* e *hardware* Arduino, e em plataformas de programação abertas. O ambiente de simulação usado foi o SUMO. A principal contribuição deste texto é mostrar que uma coleção de trabalhos pontuais pode colaborar no avanço do estado da arte.

Palavras-chave: simulação, prototipação, sistemas de transporte inteligentes.

1 Introdução

Este texto relata trabalhos desenvolvidos e em desenvolvimento por pesquisadores no Laboratório de Sistemas de Computação (LSC) da Universidade Federal

de Santa Maria (UFSM) para a implementação de soluções para o contexto de Cidades Inteligentes. Os projetos tem sido desenvolvidos desde 2010 pelos alunos dos cursos de bacharelado em Ciência da Computação e Sistemas de Informação, e de mestrado do Programa de Pós-Graduação em Informática da UFSM.

Mais precisamente, neste texto são enfocados projetos nos eixos de sistemas de transportes inteligentes, infraestruturas para cidades inteligentes e automação residencial. Os eixos são explorados de duas formas: simulação computacional e prototipação. Para a simulação de sistemas de transporte, foi usado o simulador de código aberto SUMO [1]. Os trabalhos que envolvem prototipação estão em em duas frentes: trabalhos desenvolvidos na plataforma de *software* e *hardware* Arduino⁵ e trabalhos desenvolvidos em plataformas de programação abertas. Plataformas de programação abertas foram usadas para simular serviços de base gerenciais para infraestruturas computacionais, como serviços de gerenciamento autônomicos e um serviço para atualização de *software*. Um objetivo claro e que está presente em todos os trabalhos é o uso do suporte de plataformas de baixo custo e de sistemas de implementação aberta.

O conceito de *smart cities*, ou cidades inteligentes, emprega tecnologias de informação e comunicação (TICs) para melhorar a infraestrutura urbana, buscando tornar os centros urbanos mais eficientes e melhores de se viver. De forma análoga, o conceito de sistemas de transportes inteligentes também faz uso das TICs para melhorar a eficiência e a segurança de redes de transporte.

Sabe-se também que o conceito cidades inteligentes e de sistemas de transportes inteligentes dificilmente serão completamente implementados. Entretanto, estes conceitos direcionam e impulsionam as aplicações executam em plataformas inteligentes, o que abre uma infinidade de oportunidades para pesquisa e desenvolvimento de novas soluções.

2 Simulação no controle de interseções de trânsito

No futuro, com a implantação das tecnologias de redes veiculares, o trânsito irá se beneficiar da comunicação intra-veicular para o controle semafórico. Semáforos serão virtualizados, e não haverá mais a necessidade do uso de sinaleiras para o controle do fluxo de veículos e pedestres. Até lá, novos algoritmos devem ser projetados e avaliados.

2.1 Distribuição do fluxo do trânsito

Em Dinanga e Pasin [2], foi apresentada uma implementação de uma simulação no contexto de redes de transporte. Diferentes políticas para o gerenciamento de interseções foram avaliadas. A simulação conta com duas vias concorrentes: uma com sentido sul ao norte (*s*), e outra com sentido oeste ao leste (*w*), as duas se encontrando uma interseção, conforme mostrado na Figura 1. Os veículos que estão dentro do alcance de transmissão podem colaborar para o algoritmo de controle semafórico.

⁵ <https://www.arduino.cc>

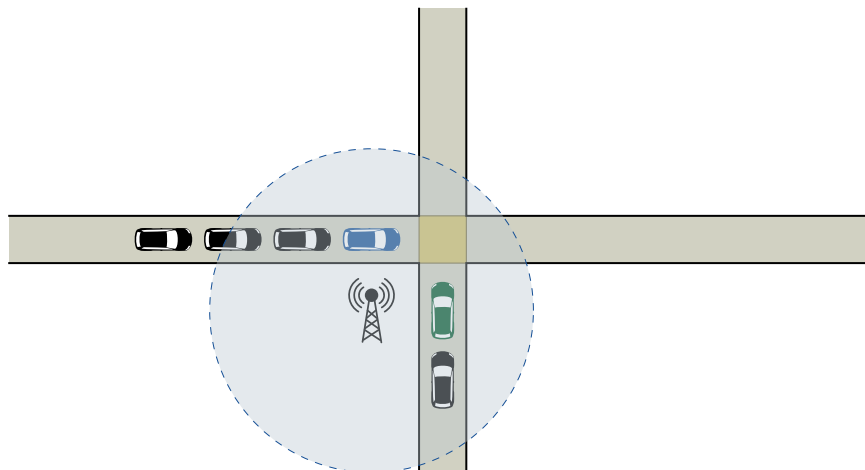


Figura 1: Cenário alvo com uma interseção e duas vias.

Foram criados três estados de fluxos do trânsito: E1, E2, e E3, baseados na preferência da s sobre a w na passagem dos veículos por uma interseção, no período do acesso dos veículos nas duas vias, e na velocidade máxima das duas vias. Depois foram implementadas cinco políticas para controlar a passagem dos veículos de s e w pela interseção. Por fim, foram gerados 15 cenários decorrentes da aplicação das políticas implementadas aos estados de fluxos do trânsito definidos, classificados em três grupos de cenários de acordo com cada estado do trânsito.

Experimentos com estes três cenários foram executados para o controle da interseção com duas vias concorrentes, para avaliar a distribuição do fluxo. Os resultados de mostraram que quando o fluxo é idêntico nas duas vias (E1), a política da maior fila primeiro (LQF) é a mais adequada. Quando a frequência do fluxo da via preferencial é superior à via concorrente (E2), a política temporizada do semáforo mostrou-se a mais adequada. Por fim, quando a frequência do fluxo da via preferencial é inferior à via concorrente (E3), o algoritmo *zipper merge* (ZM) foi a mais adequado. O algoritmo ZM alterna entre os fluxos, permitindo a passagem de x veículos (isto é, um comboio) a cada *round*. Sempre que ocorre a troca do fluxo, é adicionada uma penalidade pois uma fila precisa parar e outra precisa iniciar o escoamento. Se o comboio é formado por apenas um carro, isto é, se $x = 1$, haverá sempre uma alternância de passagem de fluxo cada vez que um único carro passa a interseção.

Com os experimentos, concluiu-se que aplicar apenas uma política não adaptativa para controlar o trânsito urbano não é a solução mais adequada. Como o trânsito é dinâmico, políticas adaptáveis ao dado contexto podem ser mais interessantes.

2.2 Eficiência no escoamento do trânsito

Na sequência, no trabalho de Fao de Moura *et al.* [3] foi comparada a eficiência no escoamento do trânsito (vazão) de algoritmos para o controle semafórico virtuais em redes veiculares. A eficiência implica na redução dos tempos de viagens e, possivelmente, auxilia na redução de congestionamentos. Os algoritmos foram implementados no simulador SUMO. Resultados demonstram que soluções adaptativas oferecem ganhos significativos.

Foram implementados três algoritmos que possibilitam operação em semáforos virtuais: ZM, fila mais velha primeiro (OQF) e a LQF. A métrica usada para avaliar os diferentes algoritmos foi a vazão.

O algoritmo OQF avalia o tempo de geração do veículo no topo das filas concorrentes w e s , uma para cada fluxo. Se o veículo no topo da fila w é saído antes da garagem que o veículo no topo da fila s , a fila de w passa primeiro. De outra forma, a fila s passa primeiro. Finalmente, o algoritmo LQF avalia o tamanho das filas em w e s . Se o tamanho da fila w é maior que o tamanho da fila em s , a fila em w passa primeiro. De outra forma, a fila s passa primeiro.

Os três algoritmos foram avaliados levando em conta diferentes fluxos de trânsito. O algoritmo ZM foi avaliado usando diferentes tamanhos de comboios: $x = 1, 10$ e 20 veículos. Este algoritmo foi contrastado com os demais. O que se pode observar, é que algoritmos baseados em comboios, como ZM para $x = 20$ e LQF, apresentam a melhor vazão mesmo com diferentes volumes de tráfego. Quando $x < 10$, ocorrem muitas trocas nas filas, e o algoritmo ZM perde a eficiência em termos de escoamento. A vazão, portanto, é baixa. O Algoritmo OQF, voltado para a justiça, dado que a fila mais velha nunca espera, também apresentou resultados interessantes e similares ao LQF. Portanto, conforme o esperado, algoritmos com comboios grandes e, portanto, com menos trocas, são as melhores soluções para escoamento eficiente de veículos.

Finalmente, uma exploração mais ampla de algoritmos para o controle de uma interseção foi realizada em Pasin *et al.* [4]. Nesse trabalho, foram avaliados três algoritmos usando duas métricas opostas: justiça e vazão. Os algoritmos que avaliam cada demanda de cada veículo individualmente, considerando aspectos do fluxo, são tipicamente mais adequados para a justiça. Algoritmos que priorizam comboios são mais adequados para a vazão. Um novo algoritmo foi proposto, eficiente em termos de justiça e vazão.

2.3 Eficiência em termos de baixa emissão de gases poluentes

Com o aumento de veículos nas vias, há também outros problemas associados, como o efeito estufa causado pela emissão de gases poluentes provenientes dos veículos e a alta demanda por combustível. Esses gases tóxicos são prejudiciais para a saúde e o meio ambiente. Dessa forma, a otimização de tráfego se torna importante, não somente para a redução do tempo de viagem dos veículos, como também para minimizar a emissão dos gases poluentes.

Em Liza e Pasin [5] são utilizados diferentes algoritmos para controlar as interseções em uma rede de transporte. Os autores, testam o comportamento

dos algoritmos: ZM, LQF e *First In, First Out* (FIFO). O objetivo do trabalho é investigar qual das abordagens minimiza a emissão de gases poluentes (CO_2), o consumo de combustível e a duração média da viagem, e aumenta a vazão de veículos nas interseções.

Para avaliação dos algoritmos, experimentos foram conduzidos no simulador SUMO utilizando uma rede de transporte com quatro interseções. A rede de transporte em questão possui uma via arterial e quatro vias coletoras, as quais cruzam a via arterial. Todas as interseções possuem um semáforo virtual, que adota uma política (ZM, LQF ou FIFO). Em cada execução, todos os semáforos virtuais executam a mesma política. Além disso, três níveis de fluxo de tráfego foram testados (alto, médio e baixo). Para cada nível foram alteradas as variáveis velocidade das vias e o período com que os veículos são inseridos na simulação. Os resultados obtidos pelos algoritmos foram comparados entre eles.

Para as métricas emissão de CO_2 , consumo de combustível e tempo médio de viagem, quando menor o valor, melhor. Para a métrica vazão, quanto maior o valor, melhor é o desempenho do algoritmo. A Figura 2 apresenta os gráficos dos resultados obtidos pelos algoritmos para cada uma das métricas. O algoritmo orientado a comboios LQF é o que obtém os melhores resultados se comparado com os algoritmos ZM e FIFO, em todos os níveis de fluxo de tráfego. Portanto, entre as políticas testadas, o algoritmo LQF é o que emite menos poluentes e aumenta a vazão na rede de transporte avaliada.

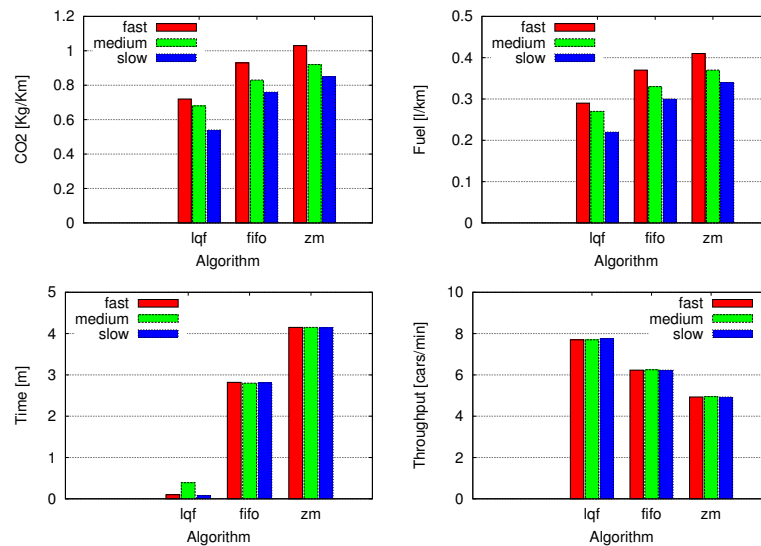


Figura 2: Gráficos dos resultados obtidos pelos algoritmos para diferentes métricas e fluxos de tráfego.

3 Prototipação: infraestruturas de suporte para cidades inteligentes

Uma vez implantado o conceito de cidades inteligentes e de sistemas de transportes inteligentes, sabe-se que estas aplicações executarão sobre infraestruturas também inteligentes. As infraestruturas terão demandas elásticas, onde o sistema está sujeito a picos de carga na chamada hora do *rush*, quando recebe muitas requisições de usuários, e momentos de ociosidade, por exemplo, durante a madrugada. Um sistema de informação ao usuário de transporte é uma aplicação típica de demanda elástica. Sabe-se também que este sistema precisa estar continuamente disponível e que desativá-lo para reparar falhas ou realizar atualização, é algo inaceitável. Os trabalhos que seguem contribuem nesta direção.

3.1 Infraestruturas para aplicações com demandas elásticas

Em Perini *et al.* [6] [7] foi proposto um sistema de auto-gerenciamento para uma infraestrutura computacional que suporta aplicações Web com demandas elásticas. O esquema desta arquitetura é apresentado na Figura 3.

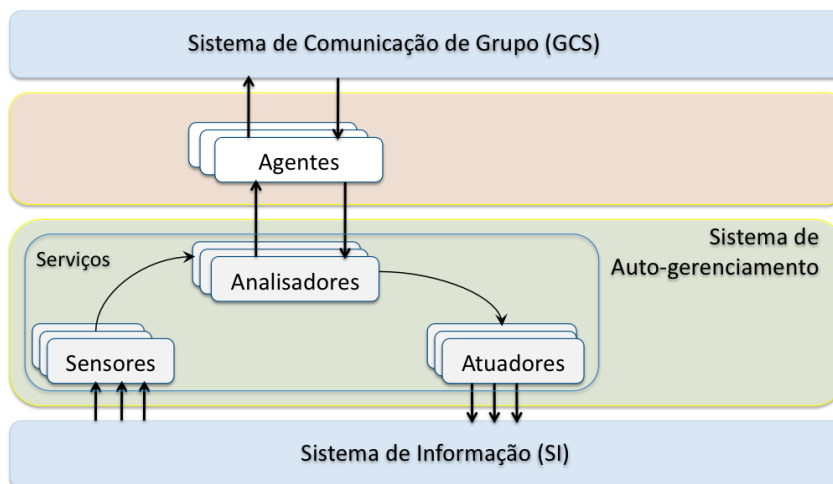


Figura 3: Arquitetura em múltiplas camadas para um sistema com auto-gerenciamento que suporta uma aplicação provedora de informação ao usuário de transportes.

O sistema de auto-gerenciamento é implementado como uma coleção de serviços descentralizados. Baseado na informação coletada por sensores embutidos na aplicação (Sistema de Informação ou SI) que executa em uma infraestrutura

computacional, os serviços de auto-gerenciamento desencadeiam adaptação dinâmica sempre que necessário. Por exemplo, o serviço de reconfiguração adapta a quantidade de servidores disponíveis na infraestrutura com a finalidade de manter a qualidade do serviço.

Esta proposta destaca as seguintes características: (i) suporte ao tratamento de decisões conflitantes, (ii) economia de recursos: em períodos de ociosidade, servidores excedentes são logicamente removidos da infraestrutura e podem ser alocados a outras aplicações ou podem operar em *standby*, e (iii) suporte estratégico para reconfiguração de servidores devido à demanda elástica e manutenção de configuração (número de servidores) em virtude de picos de carga momentâneos.

O suporte à tomada de decisão é realizado através de um sistema multiagente. As decisões não são somente tomadas com base em métricas atuais, mas com base em dados históricos. O suporte à reconfiguração é oferecido através de primitivas implementadas por um sistema de comunicação de grupo (SCG)⁶.

3.2 Atualização dinâmica em infraestruturas computacionais

Apesar de sistemas que oferecem informação ao usuário não possuírem demanda de tempo real crítica, eles precisam estar continuamente disponíveis. Uma questão importante é que os sistemas providos por terceiros (bancos de dados, servidores de aplicação) estão em contínua atualização. Novas versões são lançadas no mercado a cada ano e é importante mantê-las atualizadas pois a cada nova versão, novas melhorias são adicionadas. O sistema de gerenciamento de banco de dados (SGBD) é parte fundamental do *software* corporativo e sua atualização dinâmica não é simples devido à complexidade inerente. Frequentemente, atualização de *software* é uma tarefa executada manualmente e está associada à indisponibilidade de serviço e uso de *hardware* adicional.

Em De Gasperi [8] foi proposto um passo na direção da atualização dinâmica de *software* (ADS) para SGBDs. Nesta solução, o modelo de componentes de *software* é usado como um alternativa de suporte para a ADS. O modelo de componentes apresenta um nível de abstração que favorece a troca de componentes pois oculta detalhes de implementação. Em uma infraestrutura de *software*, cada SGBD, e cada servidor de aplicação pode ser encapsulado em um componente e uma versão defasada pode ser substituída por uma nova versão daquele componente, se houver compatibilidade. A atualização ocorre de forma incremental, sem interromper o serviço. Para validação da proposta foi desenvolvido um protótipo utilizando o modelo de componentes Fractal⁷. Experimentos em ambiente controlado confirmam a viabilidade da solução se as versões forem compatíveis. Porém, destaca-se que o próprio modelo de componentes ainda é um gargalo.

⁶ <http://jgroups.org>

⁷ <http://fractal.ow2.org/tutorials/adl>

4 Prototipação em sistemas de transporte: localização de veículos

A plataforma Arduino oferece em uma excelente alternativa para a prototipação de pequenos projetos de computação embarcada. O LSC adquiriu em 2015 quatro carros-robôs Arduino, e diferentes tecnologias de comunicação, que são usados em experimentos no contexto de redes veiculares. Esta plataforma está sendo usada como base para a implementação de projetos no contexto de redes veiculares no LSC.

Silveira Rodrigues *et al.* [9] apresentaram um serviço de auto-localização para veículos aproveitando a infraestrutura de redes veiculares. Em um esquema baseado em trilateração que comunica sensores e veículo Arduino, um algoritmo de auto-localização possibilita que o veículo se localize no ambiente.

A solução proposta possui dois tipos de componentes principais: sensores e unidade central de processamento. Os sensores captam informações sobre a localização de um veículo v e são implementados por SONARes (*Sound Navigation and Ranging*). A informação coletada pelo sonar consiste na distância entre o sonar e um veículo v . Através desta distância, a unidade central de processamento computa as coordenadas (x, y) de um dado veículo. As coordenadas são válidas apenas para o ambiente e que são conhecidas através pontos de referências, podendo ser um mapa do ambiente ou posições preestabelecidas.

Um sonar é um instrumento que detecta a presença de um objeto medindo o tempo de refração de uma onda sonora. Neste trabalho, um conjunto de sonares é apresentado como uma tecnologia para localização veículos. Esta solução, além de oferecer precisão significativa, representa uma opção de baixo custo em comparação ao GPS.

O sonar implementado neste projeto é composto por um sensor ultrassônico HC-SR04 e um servo motor 9g SG90, que são conectados à uma placa Arduino UNO. O sensor ultrassônico detecta e calcula a distância de obstáculos na sua frente. Ele emite uma onda sonora que é refletida quando colide com um obstáculo. A distância é calculada através da medição do intervalo de tempo que a onda sonora é emitida até o momento que ela retorna refletida.

Para que o sistema de localização calcule a posição do veículo, o sensor é acoplado ao servo motor. O servo motor rotaciona o sensor em seu próprio eixo, fazendo com que aumente a área de detecção e permitindo calcular a posição do veículo. Através da plataforma Arduino, é possível controlar precisamente qual o grau de rotação que o eixo do servo motor está rotacionado e, conseqüentemente, a direção para qual o sensor está apontando.

Além de controlar o sensor e o motor, o Arduino é responsável por juntar essas informações com a posição do sonar no ambiente, e enviá-las para a computador, que opera como central de controle. A cada rotação do servo motor, é enviada para a central uma mensagem contendo essas informações. De posse das informações coletadas por três sonares, a central infere a posição de um veículo por trilateração, estando ele em movimento ou parado. Experimentos foram realizados em um ambiente controlado e apresentaram erro de $\approx 8\text{cm}$, entre a

posição obtida pelos sonares e aposição indicada na pista, para uma pista circular com 5,5m de comprimento. Dado que o carro tem tamanho de 20cm², este erro é aceitável. A Figura 4 mostra o esquema do cenário onde os experimentos foram realizados e foto real do cenário com a pista circular.

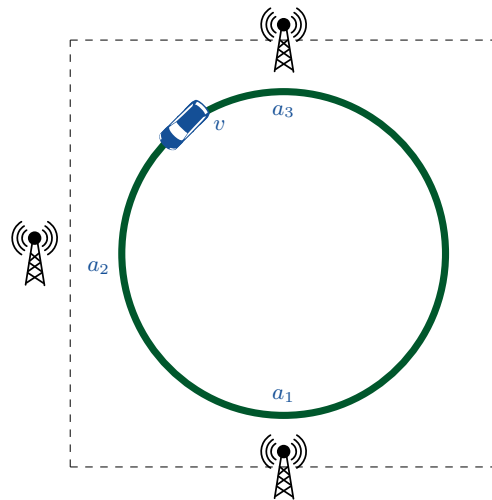


Figura 4: Esquema do cenário dos experimentos e foto real do cenário com a pista circular.

5 Prototipação em automação residencial

Tecnologias para automatização de ambientes e residências buscam prover controle automático e segurança enquanto diminuem o consumo de energia. Uma *smart house* consiste em uma casa com sensores inteligentes e atuadores operando em diferentes plataformas com objetivos diferentes. Em Mota da Silva [10] foi desenvolvida uma arquitetura de serviços para provimento de informações e gerenciamento de uma *smart house*.

Uma maquete foi construída para atuar como sistema protótipo. A maquete possui uma placa Arduino UNO, que é responsável pelo controle de sensores e atuadores. O processamento dos dados é realizado em um computador convencional. A ênfase deste projeto é no sistema de iluminação e no sistema de alarme, que conta com sensores de presença, controlados através de uma interface Web. A interface Web foi desenvolvida com o uso do *framework* Django⁸. Através da

⁸ <https://www.djangoproject.com>

interface, pode-se visualizar os dados coletados, atualizados frequentemente com o uso da tecnologia AJAX⁹, em forma de gráficos e relatórios por período informado, informando quanto tempo as luzes ficaram ligadas, desligadas, quantas vezes foram acesas e se o alarme foi ativado ou não. O usuário também pode ligar ou desligar as luzes manualmente. Também foram desenvolvidos um módulo para controlar o portão da garagem e outro módulo de alarme contra intrusão.

Em duas peças da casa foram instaladas lâmpadas LED e sensores de presença. Foram necessários o uso de resistores para permitir o acionamento das lâmpadas de LED na casa. A aparência da maquete com a placa Arduino é apresentada na Figura 5.

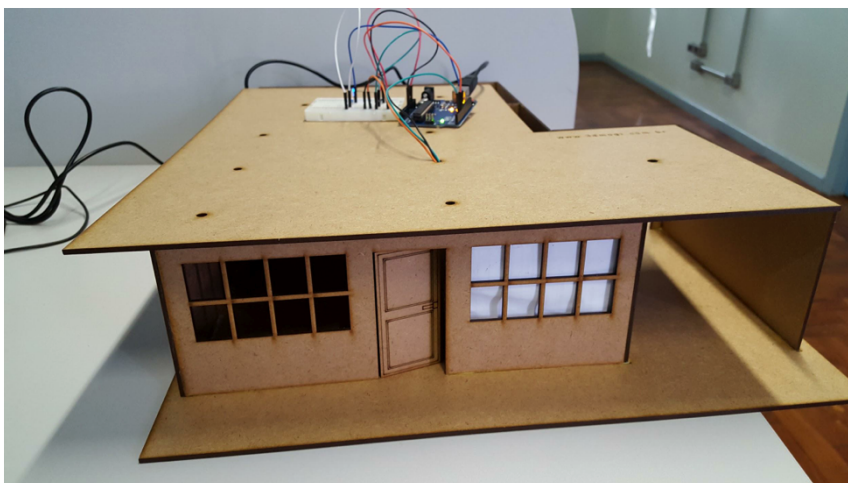


Figura 5: Maquete usada no projeto.

Os próximos passos deste trabalho incluem *(i)* a troca da placa Arduino UNO por uma placa mais robusta, para permitir a adição do módulo Wi-Fi para estabelecer a comunicação entre a casa e um computador remoto, *(ii)* melhoramento da interface gráfica, com a adição de novas funções e *(iii)* melhoramento do módulo de relatórios.

6 Trabalhos em desenvolvimento e pesquisa futura

Finalmente, esta seção apresenta trabalhos que estão sendo desenvolvidos no LSC. Destaca-se a avaliação do serviço intra-campus da UFSM e a avaliação da comunicação veicular utilizando diferentes tecnologias.

⁹ <http://glm-ajax.sourceforge.net>

6.1 Avaliação do serviço intra-campus da UFSM

A UFSM conta com um vasto *campus* e um serviço gratuito de transporte interno para facilitar a circulação de alunos, professores e funcionários. Basicamente, neste serviço, existe um ônibus circular que parte a cada 30 minutos do terminal Centro Comercial do Campus, percorrendo um percurso pré-estabelecido. O serviço opera das 7h30min até as 22 horas, de segunda a sexta.

A tabela detalhada com os horários (de partida no ponto inicial e de chegada no ponto final), está disponível no *site* da UFSM. A informação sobre os horários oferecida no *site* é estática e não reflete o fato de que o trânsito está sujeito a transtornos. Por exemplo, em horários de pico, atrasos são recorrentes.

Outro problema é a ausência de informação sobre paradas deste ônibus na UFSM. Com exceção da parada inicial, não se tem informações sobre a localização das demais paradas e sobre os horários que o ônibus serve tais paradas. Também não se sabe qual horário o ônibus chega em qual parada.

O trabalho em andamento, desenvolvido pelo aluno Ezequiel Rodrigues Ribeiro propõe uma avaliação deste serviço. A avaliação é composta de duas partes: (i) coleta de dados através de um aplicativo, e (ii) descoberta de conhecimento pela associação dos dados coletados.

Como resultado principal do desenvolvimento deste trabalho, espera-se avaliar o serviço e detectar onde o serviço pode ser melhorado. Perguntas que devem ser respondidas: qual é o horário mais sobrecarregado? Qual é o horário no qual o ônibus está mais vazio? Quais são as paradas com maior demanda? Como a demanda está distribuída pelos dias da semana? Onde estão as paradas? Espera-se, de forma geral, colaborar eventualmente para a qualidade do serviço do transporte intra-campus.

6.2 Avaliação da comunicação veicular utilizando diferentes tecnologias

Outro projeto de pesquisa em andamento busca avaliar a possibilidade de implementar a comunicação veículo-infraestrutura usando diferentes tecnologias. A plataforma Arduino é uma plataforma bem restrita, em termos computacionais. Entretanto, possui um conjunto amplo de módulos de comunicação, com diferentes tecnologias (Bluetooth, Bluetooth LE, Wi-Fi, LoRA¹⁰). Estas tecnologias já foram adquiridas pelo LSC e estão sendo avaliadas pelo aluno Leonardo de Abreu Schmidt.

7 Conclusões

Este texto apresentou os projetos desenvolvidos e em desenvolvimento pela equipe do LSC da UFSM no contexto de cidade inteligentes, ao longo dos últimos 8 anos. Destacam-se dois eixos: transportes e automação residencial, e duas abordagens: prototipação e simulação. Os projetos contam ou contaram com

¹⁰ <https://www.lora-alliance.org>

colaborações externas, onde destaca-se a Universidade Federal do Rio Grande do Sul (UFRGS, Porto Alegre), Institut Polytechnique des Sciences Avancées (IPSA, Ivry-sur-Seine, França) e Universidade Humboldt (Berlim, Alemanha). É importante ressaltar que as soluções apresentadas pelo grupo de pesquisa focam sempre no uso de plataformas de baixo custo e de especificação aberta.

Referências

1. M. Behrisch, L. Bieker, J. Erdmann, D. Krajzewicz. SUMO – Simulation of Urban MObility: an overview, In: Proceedings of the 3rd SIMUL: The Third International Conference on Advances in System Simulation - ThinkMind (SIMUL 2011). pp. 55-60.
2. E. K. Dinanga, M. Pasin. Toward equitable vehicle-based intersection control in transportation networks. In: Proceedings of the 8th International Workshop on Agents in Traffic and Transportation (ATT-2014), Vizzari, Kluegl, Vokrinek (eds.), joint with the 13th Int. Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014). Paris, França. pp. 48-55.
3. R. Fao de Moura, E. T. Zancanaro, L. Lunardi-Lemos, M. Pasin. Avaliação da eficiência de algoritmos para semáforos virtuais em redes veiculares. In: ERAD 2015, 2015, Gramado - RS. Anais da XV ERAD, 2015. pp. 185-188.
4. M. Pasin, B. Scheuermann, R. Fao de Moura. VANET-based intersection control with a throughput/fairness tradeoff. In: Proceedings of the 8th IFIP Wireless and Mobile Networking Conference (WMNC 2015), 5-7 Outubro 2015, Munique, Alemanha. pp. 208-215.
5. L. Lunardi Lemos, M. Pasin. Intersection control in transportation networks: opportunities to minimize air pollution emissions. In: IEEE 19th International Conference on Intelligent Transportation Systems (ITSC 2016), Novembro 2016, Rio de Janeiro, Brasil. pp. 1616-1621.
6. M. Pasin, F. S. Perini, Ana L. C. Bazzan. Self-management as support to an advanced traveler information system. In: Proceedings of the SBSI 2012 - São Paulo - SP: Sociedade Brasileira de Computação, 2012. pp. 138-143.
7. M. Pasin, Ana L. C. Bazzan, F. S. Perini. Toward a self-managed distributed infrastructure to an advanced traveler information system. In: Proceedings of the Autosoft 2011, em conjunto com CBSOft 2011, São Paulo - SP. pp. 33-39.
8. C. F. De Gasperi, M. Pasin. Proposta para atualização de SGBDs para aplicações com demanda de contínua disponibilidade usando suporte do modelo de componentes de software. In: Proceedings of the Workshop de Testes e Tolerância a Falhas 2013 - WTF 2013, em conjunto com Simpósio Brasileiro de Redes de Computadores 2013 (SBRC 2013), Brasília - DF, Brazil. Porto Alegre: Sociedade Brasileira de Computação, 2013. v. 1. pp. 3-16.
9. R. Silveira Rodrigues, M. Pasin, R. Machado. Indoor position tracking: an application using the Arduino mobile platform. In: Proceedings of the 10th IFIP Wireless and Mobile Networking Conference (WMNC 2017), 25-27 Setembro 2017, Valencia, Espanha.
10. M. L. Mota da Silva. Serviço para gerenciar uma Smart House: uma implementação usando a plataforma Arduino. Trabalho de Graduação, Curso de Sistemas de Informação, Universidade Federal de Santa Maria, Santa Maria - RS, Julho de 2016. 33p.

Desenvolvimento de um protótipo com utilização de LoRaWan como Solução de Comunicação de Baixo Custo

Fernando Vinícios Manchini de Souza e Roberto dos Santos Rabello

Universidade de Passo Fundo, Passo Fundo, Brasil,
168730@upf.br
rabello@upf.br

Resumo Com a aglomeração de pessoas recursos são cada vez mais mal utilizados, as cidades inteligentes apoiadas fortemente na Internet das Coisas surgem como um contraponto, para a otimização de recursos e bem-estar social, mas o custo da comunicação utilizando rede celular é um dos grandes limitadores para aplicações em larga escala, ou em cidades pequenas com poucos recursos. Esse trabalho apresenta o desenvolvimento de protótipo de baixo custo utilizando a tecnologia *LoRaWan* como meio de comunicação viável para a implementação de futuras aplicações, tanto no aspecto financeiro e baixa complexidade, sendo possível cobrir a área de uma cidade de médio porte com apenas um *gateway*.

Keywords: LoRaWan, lpwan, Cidades Inteligentes, Fazendas Inteligentes

1 Introdução

Com a aglomeração de pessoas nas cidades recursos e bens são muitas vezes mal utilizados[1], faltando para uns enquanto são desperdiçados por outros. No cenário de cidades inteligentes, a comunicação e o acesso facilitado a informação são pontos chave para um menor desperdício de tempo e recursos, por exemplo a notificação antecipada que o trânsito estará bloqueado e a sugestão do desvio, o agendamento de consultas médicas pela internet, a distribuição de serviços essenciais como bancos pela cidade, otimização dos meios de transporte através do compartilhamento, produção e distribuição de alimentos em fazendas urbanas. Esses casos representam a economia compartilhada *blockchain* e cidades inteligentes, onde é promovido o empoderamento do cidadão, interagindo de forma mais ativa na comunidade, sendo como prestador de serviço ou usuário.

Todas essas iniciativas necessitam de sistemas de comunicação, estes são fortemente dependentes de poucas empresas de telecomunicação que proveem altas velocidades, mas com um alto custo por dispositivo, que devido a necessidades regulatórias restringem o mercado, inibindo, até certo ponto, o investimento e a concorrência, tornando inviável algumas aplicações.

Como contraponto as redes *LPWAN* apresentam novas possibilidades com longas distâncias, baixo consumo energético e custo, no caso da implementação *LoRaWan*, por utilizar faixa de frequência restritas, que não necessitam de cadastramento junto ao órgão regulatório, reduz a burocracia e custos para sua utilização, sendo ainda possível instalar um *gateway* e mantê-lo em operação, sem necessidade de intermediários, que ocorre em outras redes como a *SIGFOX*, que a cobertura do sinal é exclusivo de uma empresa em cada país [2].

O desenvolvimento de sistemas inteligentes customizados, focados em problemas específicos de cada cidade, tende a ser mais efetivo e gerando mais oportunidades de negócios para empresas e pesquisadores locais. O objetivo deste artigo é desenvolver um protótipo de baixo custo para verificar a viabilidade para implementações de aplicações utilizando a tecnologia *LoRaWan* no contexto da coleta e comunicação nas cidades e fazendas inteligentes, tendo como ambiente do experimento a cidade de Ijuí, no noroeste do estado do Rio Grande do Sul.

Considerando os trabalhos relacionados já desenvolvidos em outros países utilizando a tecnologia *LoRaWan*, buscando extrair seus acertos para o estudo de viabilidade da aplicação desta tecnologia em pequenas e médias cidades, onde os recursos financeiros são mais escassos para esse tipo de investimento.

2 Fundamentação Teórica

A maior parte da população vive nas cidades, cerca de 3,42 bilhões de pessoas superando 3,41 bilhões das áreas rurais e a estimativa que em 2050 sejam 5,2 bilhões representando um percentual de 84 % vivendo nas cidades [1]. Essa questão aliada com o fato que as cidades crescem sem um planejamento central, mas sim com as decisões individuais de cada cidadão criando um ambiente anárquico, geram vários problemas para o desenvolvimento e crescimento das cidades, no entanto acaba criando oportunidades em vários aspectos.

Com o advento da quarta revolução industrial criaram-se novos paradigmas, apoiados fortemente na tecnologia e na integração entre o mundo físico, virtual e biológico permitido pela *Internet of Things*, IoT e Inteligência Artificial, IA, um exemplo de paradigma é a produção distribuída local, para atender demandas específicas com produtos customizados, tendo como exemplo a impressão em 3D [3]. O conceito de *Internet of Things*, IoT, emergiu no fim dos anos 90 por ASHTON em 1999, em sua palestra para a P&G sobre RFID [4], desde essa primeira explanação o conceito evolui muito e hoje pode ser definido como uma infraestrutura global para interligação de objetos físicos e virtuais, através dos sistemas de comunicação para captura de informações e atuação no ambiente [5] [6]. A popularização de soluções IoT vêm crescendo devido à vários fatores como a evolução na miniaturização, redução de custo de sensores, velocidade de conexão, capacidade de identidade única de cada dispositivo, poder de processamento e armazenamento das informações na nuvem a baixo custo e o desenvolvimento da computação cognitiva e *machine learning* [7].

A convergência destes aspectos juntamente com as preocupações ambientais e de bem-estar social, quando balanceadas para atender as necessidades e desejos

da população com uma capacidade funcional sustentável, utilizando a tecnologia para transformar a infraestrutura e otimizar os recursos, criando assim um ambiente com o conceito de cidade inteligente[8]. Para fazer a leitura do mundo real são utilizados um grande número de dispositivos e sensores, distribuídos em longas distâncias e lugares muitas vezes de difícil acesso, ainda é possível que estejam em uma área que não tenham disponibilidade de fornecimento de energia elétrica, sendo necessário uma longa autonomia de bateria.

2.1 LoRaWan

O nome *LoRa* é a abreviatura de *Long Range*, sendo uma *Low Power Wide Area Network*, *LPWAN* que consistem em uma arquitetura de rede voltada a baixo consumo de energia e longo alcance, pontos chave para IoT, provendo comunicação segura bidirecional, mobilidade e serviço de localização. Esta rede geralmente utiliza a topologia estrela, com um *gateway* como ponte entre os dispositivos e o servidor central da rede, esse que possui uma conexão TCP/IP convencional. Outro ponto no padrão *LoRa* é que ele é aberto, permitindo qualquer empresa produza módulos, entretanto devem ser homologados pela *LoRa Alliance*[9].

A comunicação ocorre entre o dispositivo e o *gateway* utilizando faixas de frequência e taxa de dados específicas, as taxas definem o tempo que o dispositivo vai utilizar o canal de comunicação, assim como o tempo que a mensagem vai ficar no ar, quanto menor a taxa de transferência maior será o tempo no ar, as taxas de transferência variam de 0,3 kbps a 50 kbps. Isso na prática permite que na mesma área de abrangência de um *gateway* tenha diferentes taxa de dados, pois os dispositivos localizados mais perto do *gateway* podem ficar menos tempo no ar, assim economizando energia.

Um dos pontos críticos da infraestrutura para IoT é a segurança, assim para aumentá-la os dados durante a comunicação são criptografados e utilizando também identificadores únicos:

- *Unique Network key* (EUI64): Identificador único na rede com 64 bits.
- *Unique Application key* (EUI64): Identificador único da aplicação com 64 bits.
- *Device specific key* (EUI128) Identificador único do dispositivo com 128 bits.

Os dispositivos *LoRa* são categorizados em classes para utilização adequada em cada projeto, sempre voltado para o menor consumo de energia:

Classe A: dispositivos bidirecionais, mas tendo a possibilidade de receber mensagens restritas logo após o envio, essa forma de operação reduz muito o consumo de energia, pois durante o resto do tempo o módulo estará desligado.

Classe B: dispositivos bidirecionais, mas tem possibilidade de receber mensagem com tarefas sincronizadas de tempo em tempo. Ao enviar uma mensagem pode ser criado uma tarefa para sincronizar o próximo horário que o dispositivo estará ouvindo.

Classe C: dispositivo bidirecional mas com a possibilidade de recebimento de mensagens a quase qualquer momento.

A utilização de faixas de frequência de rádio é limitada pelos governos locais dos países para definir regras, organização e evitar problemas de interferências entre os equipamentos de uso crítico, assim há um acordo internacional que define algumas frequências sem necessidade de regulamentação para incentivar o desenvolvimento industrial, científico e médico, esse acordo é denominado ISM band, *industrial, scientific and medical*. No Brasil o órgão regulador de telecomunicação é a Agência Nacional de Telecomunicação, ANATEL, entre as suas responsabilidades está as definições das faixas de frequência que necessitam de cadastramento para operação, logo segundo a resolução 506 da Anatel, no capítulo II, define as faixas de frequência restritas, que não necessitam de cadastramento junto ao órgão, conforme Tabela 1 [10].

Tabela 1. Tabela 3 segundo Resolução 506 Anatel

Radiofrequência
902-907,5 MHz
915-928 MHz
2.400-2.483,5 MHz
5.725-5.875 MHz
24,00-24,25 GHz

A instrução da *LoRa Alliance* para o continente americano é utilizar a faixa de frequência 902 a 928 MHz, Como pode ser observado nas Tabela 1 a faixa entre 907,5 a 915 MHz não pode ser utilizadas, assim ao utilizar dispositivos do padrão americano deve se tomar o cuidado de inutilizar esses canais via software na aplicação. No entanto essa limitação já ocorre em outros países que padronizaram a distribuição das faixas de frequência de tal modo que não sejam inutilizado canais, como é o caso do padrão australiano, entretanto ainda não há uma padrão brasileiro definido [11].

A plataforma para prototipação rápida, coleta e comunicação, *The Things Network*, TTN, é desenvolvida pela comunidade buscando a implantação de *gateways* de acesso público para a difusão da tecnologia *LoRaWan*, sendo que atualmente são cerca de 1389 ativos ao redor do mundo. Aliada com a plataforma Arduino que é comum a utilização no início de projetos, principalmente pelo seu baixo custo e tempo de aprendizado, iniciado em 2005 para ajudar estudantes no aprendizado de eletrônica e programação, foi o primeiro projeto de código aberto em hardware [12].

2.2 Trabalhos Relacionados

Inicialmente foi realizado busca em base de dados de artigos, CAPES, Scholar - Google acadêmico, *IEEE - xplore*, ACM e *Springer*, com a palavras chave *loraWan*, *lora* e *lpwan*, com objetivo de encontrar pesquisa que abordem as especificação da tecnologia, limitações e aplicações. Por ser uma tecnologia emergente existem muitas lacunas, entretanto há alguns trabalhos que podem ser citados,

Understanding the Limits of LoRaWAN de Adelantado, Ferran et al [13] realiza um estudo imparcial e visão geral da tecnologia abordando as suas capacidades e limitações, como taxa de transmissão, distância de transmissão, capacidade máxima de dispositivos, probabilidade de sucesso de transmissão. Dados esses itens, relata contextos em que pode ser aplicado a tecnologia *LoRaWan* e suas limitações nesses cenários, como cidades inteligentes, transporte, logística e vigilância em vídeo.

Evaluation of LoRa LPWAN Technology for Indoor Remote Health and Well-being Monitoring de Juha Petäjälä [14] relata os experimentos utilizando dispositivos comerciais da tecnologia *LoRaWan* no campus da Universidade de Oulu na Finlândia, utilizando diferentes *spreading factors*, larguras de banda e potência de transmissão. No caso de estudo utilizando uma potência de transmissão de 14 dBm cobriu toda a área do campus, 570 x 320 metros, com uma média de 96,7 % de sucesso de transmissão, já utilizando 2 dBm obtiveram uma perda de 13,1% de pacotes. São relatados testes tanto no exterior quanto no interior de construções, utilizando *spreading factors* de 7 a 10, ao final é realizado um relato breve do consumo de energia com essa variação de *spreading factors*, concluindo que é possível a transmissão, mesmo *indoor* em 300 metros.

The design and implementation of a smart-parking system for Helsinki Area de Charles Wasswa Sewagudde [14] relata a validação de um sistema de estacionamento inteligente, com o foco na construção do dispositivo, o trabalho utiliza os conceitos de IoT para abordar o problema de estacionamento tendo uma região crítica da cidade, os dispositivos foram criados utilizam a tecnologia *LoRaWan*, com um módulo comercial, porém montado em uma placa de circuito impresso desenvolvida pelo próprio pesquisador. A aplicação tinha como principal objetivo verificar se há algum carro utilizando a vaga, auxiliando motoristas que buscam estacionamento e também a função de parquímetro. Ao final são relatados as dificuldades na implementação da solução tais como, custo dos dispositivos, a limitação do uso das bandas ISM, a necessidade de conhecimento diversificado em eletrônica, rádio, sistemas embarcados, banco de dados e tecnologias web, demandam bastante mão de obra aumentando o custo das aplicações. No entanto o pesquisador deixa algumas provocações e desafios futuros como automatização da identificação dos carros, pagamentos automatizados e soluções que não necessitem de *gateways*.

3 Metodologia

A implementação foi iniciada definindo um esboço do fluxo das informações através da arquitetura representada pela Figura 1, dividido em blocos e subdividindo em partes para simplificar o desenvolvimento. A parte central é o *gateway* que irá receber as informações do dispositivo via *LoRaWan*, posteriormente encaminhar para os serviços na internet pelo protocolo TCP/IP.

O desafio inicial foi encontrar fornecedores de equipamentos *LoRaWan* com preços acessíveis no varejo, pois o mercado até início de 2017 possuía apenas *gateways* profissionais com custos superior a 300 dólares [15], entretanto com a

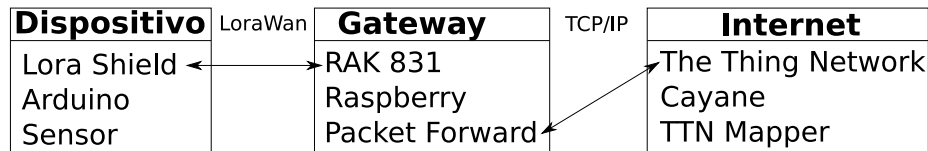


Figura 1. Arquitetura do Protótipo

difusão da tecnologia surgiram novas soluções [16], assim como soluções voltadas para a prototipação e plataformas de desenvolvimento que apesar de possuírem alguns recursos limitados [17], tem a metade do custo, sendo possível a montagem de um *gateway* para instalação *indoor* com aproximadamente 165 dólares e cada dispositivo com valores abaixo de 40 dólares.

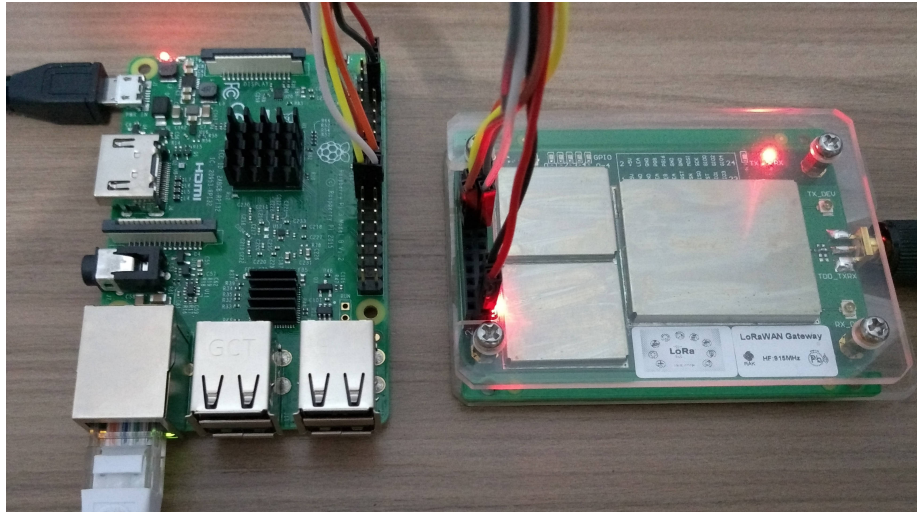


Figura 2. *Raspberry Pi 3 Modelo B* e Placa RAK 831 conectadas diretamente pelas portas GPIO

Na construção do gateway, Figura 2, foi utilizado a placa *Raspberry Pi 3 Modelo B* que possui todas as conexões e poder de processamento necessários para a operação da placa RAK 831, assim como tem seu custo extremamente baixo. No caso do módulo RAK 831, este é baseado no *Semtech SX1301*, assim como outros concentradores mais caros do mercado, podendo operar na frequência de 433, 868 e 915 MHz, com sensibilidade de recepção de -138dBm, suporte a 8 canais de recepção e 1 para envio de informações.

Na *Raspberry* foi instalado o sistema operacional padrão, *raspbian* [18], em seguida habilitado a comunicação via *Serial Peripheral Interface*, SPI. A interligação física entre o módulo RAK foi realizada utilizando as portas GPIO da

Raspberry tendo como base a documentação oficial [19] e membros da comunidade [20]. O *gateway* construído, Figura 2, conta também com uma antena omnidirecional de 5 dBi. O próximo passo foi o cadastro do *gateway* na plataforma TTN que necessita de informações básicas como identificador único, localização geográfica e faixa de frequência que será utilizada, no caso 915 Mhz.

Com esses dados também é configurado o *packet forwarder*, este que é responsável de encaminhar as mensagens dos dispositivos para a nuvem na TTN [21]. Pelo fato do módulo RAK 831 ter sido lançado a pouco tempo, não há suporte oficial pela TTN, entretanto foi possível de realizar a comunicação utilizando o protocolo SPI de outro módulo, *IMST ic880a*, que também utiliza o mesmo *semtech SX1301*.

O objetivo para o dispositivo era ser algo simples, para explorar o ambiente de desenvolvimento, noções básicas de eletrônica e principalmente as capacidades de transmissão de um simples dispositivo. No desenvolvimento do dispositivo o maior desafio foi utilizar as bibliotecas adequadamente para o módulo *LoRaWan* e frequência utilizada, no caso as principais foram desenvolvidas na Europa, operando em outras faixas de frequência, assim deve se ter o cuidado de realizar a adequação para as faixas corretas para o padrão americano. A biblioteca *LMiC* foi desenvolvida inicialmente pela IBM para módulos *LoRa* Semtech SX1272/SX1276 e HopeRF RFM92/95, mas atualmente foi modificada pela comunidade em repositórios *fork* do original sendo possível de forma relativamente simples integração com o Arduino [22]. Isso aliado com os módulos *LoRa* para Arduino já serem encapsulados na forma de *shields* agilizando a prototipação para um dispositivo simples, no caso do dispositivo construído para os testes constitui-se de um *Arduino Uno*, *Shield Lora Dragino* [17], sensor de temperatura e umidade modelo *DHT 11*, antena de 2 dBi e bateria.

Neste ponto é possível realizar a comunicação do dispositivo para o gateway, e do gateway para a nuvem, permitindo a captura de informações do ambiente real e transpondo para o virtual, entretanto foi utilizado um protocolo das mensagens voltado para rápida prototipação, seguindo o padrão *Cayenne Low Power Payload*, LLP,[23] este que é baseado no padrão *IPSO Alliance*, que busca prover interoperabilidade entre dispositivos inteligentes de vários fabricantes, como Google, Bosh, Intel, Texas Instrument, entre outros [24]. Utilizando mensagens neste protocolo é possível integrar e capturar as mensagens diretamente da rede TTN para o aplicativo *cayenne* para *smartphone* em uma alto nível, com poucos passos, sendo de certa forma didático.

A Figura 3 exemplifica como as informações são exibidas no aplicativo *cayenne*, no caso, utilizando o protocolo corretamente, já especificadas as grandezas, essas são configuradas automaticamente no aplicativo.

O TTN Mapper é um projeto derivado da *The Things Network* que consiste em um sistema que visa mapear e divulgar a cobertura dos *gateways*, assim incentivar outros desenvolvedores a utilizarem estes. Resumidamente, através do aplicativo para Android disponível no site, o usuário deve vincular um dispositivo cadastrado na rede TTN, em seguida acessar o modo de mapeamento e se mover pela cidade. Todos os pacotes recebidos serão analisados o *received signal strength*

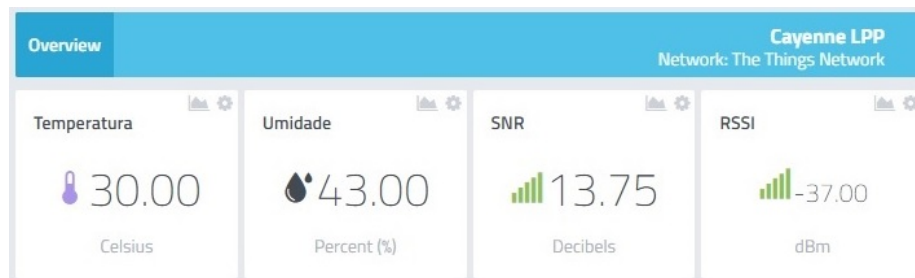


Figura 3. Informações sendo exibidas n aplicativo *cayenne*

indicator, RSSI e com essas informações será construído um mapa de calor com a cobertura do *gateway* na plataforma [25].

3.1 Análise e discussão dos resultados

Para validação foi realizado um experimento na cidade de Ijuí, localizada no noroeste do estado do Rio Grande de Sul, buscando mensurar o alcance dos dispositivos em um ambiente real de uma cidade com obstáculos, como árvores, prédios, elevações e interferências de outros possíveis aparelhos que utilizam esta mesma faixa de frequência. O *gateway* estava localizado *indoor* no 5º andar em um edifício na área central da cidade, aproximadamente a 321 metros de altura em relação ao nível do mar, assim foram realizados medições principalmente em uma direção.



Figura 4. Cobertura de Sinal do Protótipo no TTN Mapper [26]

Conforme pode ser observado na Figura 4 a cobertura do protótipo obtida pelo TTN Mapper, as áreas em tom de vermelho indicam maior potência do sinal, amarelo representam mediana e azul baixa. Neste teste inicial a maior

distância obtida foi de 2,35 Km, devido a geografia da cidade de Ijuí [27] possuindo ondulações moderadas, lugares mais distantes, porém mais altos, tem melhor cobertura, que lugares mais próximos que são mais baixos, pelo fato que durante o deslocamento ocasionalmente obtinha-se linha de visada para o *gateway*.

Pelo fato do *gateway* estar *indoor*, este tem seu alcance prejudicado na direção perpendicular ou oposta a janela, tendo baixas medições com apenas 2 quarteirões de distância. Assim constata-se que a instalação em ambiente aberto seria essencial para uma aplicação real.

Algumas medições acabaram sendo realizadas com o veículo em movimento, que não foi efetivo, entretanto as medições mais distantes foram realizadas com o veículo estacionado, obtendo melhores resultados.

O investimento financeiro total em equipamentos de 165 dólares para montar está estrutura, que permite realizar a comunicação sem fio através da cidade, aliado com o fato que para o dispositivo foi necessário apenas 40 dólares, e pode ser reduzido esse custo otimizando a construção, é muito atrativo e expande muitos horizontes para novas aplicações em vários segmentos. No entanto a compra de equipamentos importados no Brasil tem um grande entrave que são os impostos de importação de 60%, isso acaba desencorajando, até certo ponto, pesquisadores e empreendedores apostarem em tecnologias emergentes, pois o risco acaba sendo maior.

O mercado de cidades e fazendas inteligentes é muito vasto, não será efetivo uma organização estar presente em todas as aplicações e setores, cada nicho terá que se movimentar individualmente gerando demanda para os que o cercam, no caso da comunicação gerará grandes volumes de dados contínuos, o setor de análise de dados terá sua demanda elevada, que por sua vez irá apontar novos dados que precisam ser obtidos. Da mesma maneira que cada cidade irá implementar soluções para seus devidos problemas, que muitas vezes serão parecidos mas não iguais das cidades vizinhas, que criarão suas próprias soluções, gerando demandas locais que serão resolvidas pelos seus cidadãos.

4 Considerações

Este trabalho pode demonstrar que a tecnologia *LoRaWan* cumpre as expectativas de ser uma solução de baixo custo, pois não tem necessidade da contratação de planos de dados por dispositivo, mas apenas para o *gateway*, devido ao fato de utilizar frequência restritas, respeitando a mesma regras de rede *Wi-Fi*, 2.400 MHz, não tendo a necessidade de regulamentação. Os testes iniciais, mesmo com o equipamento de baixo custo, demonstram a cobertura de um raio de 2,35 Km, já permitindo a criação de aplicações abrangendo quase toda a cidade de Ijuí com apenas 1 *gateway*.

Como trabalhos futuros pretende-se investigar a cobertura do *gateway* sendo instalado em um local mais alto para buscar reduzir a limitação da topografia, as limitações com um número maior de dispositivos simultâneos para averiguar qual seria a interferência entre eles e as deficiências no tráfego, mas principalmente

é prover a criações de aplicações para solução de problemas reais utilizando a massa de dados que a tecnologia permite coletar.

Referências

1. NATIONS, U. *Urban and Rural Areas 2009*. Acessado: 14 Agosto. 2017. Disponível em: <<http://www.un.org/en/development/desa/population/publications/urbanization/urban-rural.shtml>>. Acesso em: 14 Agosto. 2017.
2. SIGFOX. *SIGFOX*. Acessado: 16 jun. 2017. Disponível em: <<https://www.sigfox.com>>. Acesso em: 16 jun. 2017.
3. SCHWAB, K. *A Quarta Revolução Industrial*. [S.l.]: Edipro, 2016.
4. ASHTON, K. That 'internet of things' thing. *RFiD Journal*, v. 22, n. 7, p. 97–114, 2009.
5. VERMESAN, O.; FRIESS, P. *Internet of things-from research and innovation to market deployment*. [S.l.]: River Publishers Aalborg, 2014.
6. ITU. *Overview of the Internet of thing - International Telecommunication Union*. Acessado: 14 jun. 2017. Disponível em: <<https://www.itu.int/rec/T-REC-Y.2060-201206-I>>. Acesso em: 14 mai. 2017.
7. DIAS, R. R. de F. *Internet das Coisas Sem Mistérios: Uma nova inteligência para os negócios*. São Paulo: Netpress Books, 2016. 120 p.
8. FARIAS, J. et al. Cidades inteligentes e comunicações. v. 1, p. 28–32, 10 2011.
9. ALLIANCE, L. *Lorawan™ specification*. *LoRa Alliance*, 2016.
10. RESOLUÇÃO n 506, de 1º de julho de 2008. Acessado: 14 mai. 2017. Disponível em: <<http://www.anatel.gov.br/legislacao/resolucoes/23-2008/104-resolucao-506>>. Acesso em: 14 mai. 2017.
11. ALLIANCE, L. *Lorawan™ regional parameters*. *LoRa Alliance*, 2016.
12. ARDUINO - About Us. Acessado: 1 jun. 2017. Disponível em: <<https://www.arduino.cc/en/Main/AboutUs>>. Acesso em: 01 jun. 2017.
13. ADELANTADO, F. et al. Understanding the limits of lorawan. *arXiv preprint arXiv:1607.08011*, 2016.
14. PETÄJÄJÄRVI, J. et al. Evaluation of lora lpwan technology for indoor remote health and wellbeing monitoring. *International Journal of Wireless Information Networks*, Springer, v. 24, n. 2, p. 153–165, 2017.
15. LORIOT. *Lora Gateways and concentrators*. Acessado: 28 ago. 2017. Disponível em: <<https://www.loriot.io/lora-gateways.html>>. Acesso em: 28 ago. 2017.
16. RAKWIRELESS. *RAK831 - LoRaWan Gateway Module*. Acessado: 28 ago. 2017. Disponível em: <<http://www.rakwireless.com/en/WisKeyOSH/RAK831>>. Acesso em: 28 ago. 2017.
17. DRAGINO. *Dragino - Lora Shield*. Acessado: 28 ago. 2017. Acesso em: 28 ago. 2017.
18. FOUNDATION, R. P. *RASPBIAN*. Acessado: 02 ago. 2017. Disponível em: <<https://www.raspberrypi.org/downloads/raspbian/>>. Acesso em: 02 ago. 2017.
19. FOUNDATION, R. P. *GPIO: RASPBERRY PI MODELS A AND B*. Acessado: 20 ago. 2017. Disponível em: <<https://www.raspberrypi.org/documentation/usage/gpio/>>. Acesso em: 20 ago. 2017.
20. KRISH, N. *Getting started with the RAK 831 Lora Gateway and RPi3*. Acessado: 20 ago. 2017. Disponível em: <<https://www.hackster.io/naresh-krish/getting-started-with-the-rak-831-lora-gateway-and-rpi3-e3351d>>. Acesso em: 20 ago. 2017.

21. THETHINGSNETWORK. *TTN Packet Forwarder*. Acessado: 28 ago. 2017. Disponível em: <https://github.com/TheThingsNetwork/packet_forwarder>. Acesso em: 28 ago. 2017.
22. IBM. *IBM LMIC*. Acessado: 28 ago. 2017. Disponível em: <<https://github.com/matthijskooijman/arduino-lmic>>. Acesso em: 28 ago. 2017.
23. MYDEVICES cayenne. *cayenne MyDevices*. Acessado: 28 ago. 2017. Disponível em: <<https://cayenne.mydevices.com>>. Acesso em: 28 ago. 2017.
24. ALLIANCE, I. *IPSO Alliance*. Acessado: 28 ago. 2017. Disponível em: <<https://www.ipso-alliance.org>>. Acesso em: 28 ago. 2017.
25. THETHINGSNETWORK. *TTN Mapper*. Acessado: 28 ago. 2017. Disponível em: <<https://ttnmapper.org/>>. Acesso em: 28 ago. 2017.
26. MAPPER, T. *TTN Mapper - gateway=manchini*. Acessado: 07 set. 2017. Disponível em: <<http://ttnmapper.org/?gateway=manchini-gateway&type=radar>>. Acesso em: 07 set. 2017.
27. IJUÍ, M. D. *Geografia de Ijuí*. Acessado: 20 ago. 2017. Disponível em: <<http://www.ijui.rs.gov.br/paginas/geografia>>. Acesso em: 20 ago. 2017.

Coleta Seletiva e o Problema de Roteamento de Veículos Green – Estudo de Caso

Gregory Tonin Santos and Luiza Amalia Pinto Cantão

Univ. Estadual Paulista Júlio de Mesquita Filho, UNESP
gregorytonin@hotmail.com, luiza@sorocaba.unesp.br

Resumo Neste trabalho apresentamos um modelo de Problema de Roteamento de Veículos Green voltado à redução de consumo de combustível e subsequente redução nas emissões de gases de efeito estufa. Um estudo de caso deste modelo é construído baseado nos dados fornecidos pela cooperativa de coleta seletiva CORESO, da cidade de Sorocaba, SP. A solução do modelo é obtida através da metaheurística *Ant Colony System* associada a Busca Tabu. Os resultados assim obtidos são comparados com os resultados exatos via um software comercial (CPLEX).

Palavras Chaves: Problema de Roteamento de Veículos Green; Taxa de Consumo de Combustível; Otimização por Colônia de Formigas.

1 Introdução

Os constantes processos de industrialização em conjunto com o crescimento econômico e populacional geraram, no decorrer dos anos, um aumento na produção de resíduos sólidos com uma sensível modificação em sua composição, indo de um predomínio de fração orgânica para outra com degradação mais lenta. Neste cenário a reciclagem de materiais torna-se uma prática indispensável para mitigação dos impactos negativos relacionados a este aumento [1].

A reciclagem consiste em procedimentos capazes de reutilizar materiais descartados, recolocando-os no ciclo produtivo. Esta alternativa é vantajosa do ponto de vista ambiental e social, pois reduz a necessidade de utilização de recursos naturais e gera empregos. Esta técnica se inicia na separação dos materiais nas residências seguido de sua destinação para cooperativas, catadores ou empresas de recicladores para aplicação dos processos de reciclagem para cada tipo de material coletado e reinserção no ciclo produtivo.

Desta forma, cooperativas de reciclagem tornam-se agentes importantes deste processo à medida em que permitem o aumento da vida útil de materiais através da coleta seletiva, separação e destinação para indústrias especializadas, auxiliando na logística reversa de produtos e recuperando os materiais utilizados. Infelizmente, a participação destas cooperativas na coleta seletiva vem diminuindo sistematicamente [2].

Esta queda de participação das cooperativas deve-se, possivelmente, a falta de investimentos financeiros e a ineficiência da coleta seletiva, visto que em muitos casos possuem poucos veículos para transporte dos materiais coletados e há

a necessidade de passar em cada residência para coleta [3]. Assim, ferramentas que visam melhorar eficiência das rotas executadas por caminhões são importantes para gestão destas cooperativas, reduzindo tempo de coleta ou consumo de combustível, economizando recursos neste processo.

Além da questão puramente econômica, temos um efeito ambiental indireto, porém benéfico, da redução no consumo de combustíveis: uma proporcional redução nas emissões de Gases de Efeito Estufa (GEE), como o dióxido de carbono (CO_2), metano e o óxido nitroso. Os GEE potencializam o efeito estufa [4], aumentando a temperatura atmosférica média e podendo ocasionar mudanças climáticas de impacto planetário.

Assim, um fator que pode ser estudado de forma a minimizar os custos relacionados ao transporte é o consumo de combustível. Segundo [5], para um caso real de uma transportadora em Shanghai, China, o custo com combustível representava, aproximadamente, 67,5% do custo total do transporte; desta forma apenas a redução do consumo com combustível acarretou em considerável minimização de gastos totais.

Uma das principais formas sistemáticas de se abordar este questão de minimização de consumo de combustíveis e emissão de GEE é através dos chamados Problemas de Roteamento de Veículos (PRV), (*Vehicle Routing Problems (VRP)*, no original em Inglês). Estes consistem de um problema de programação inteira que modela numa frota de veículos que deve satisfazer as demandas de uma série de clientes pré-estabelecidos, atendendo a todos os clientes de acordo com restrições específicas [6]. Na literatura, os PRV com enfoque ambiental são denominados Problemas de Roteamento de Veículos Green, (*Green Vehicle Routing Problems (GVRP)*, do original em Inglês).

A metaheurística implementada para resolução do GVRP estudado, e com vasta aplicação em problemas da literatura, é a Otimização por Colônia de Formigas, em Inglês, *Ant Colony System (ACS)*, desenvolvido por Dorigo e Gambardella (1997), base para o presente trabalho.

2 Formulação Matemática

2.1 Problemas de Roteamento de Veículos com Objetivo de Minimização do Consumo de Combustível

O modelo matemático utilizado é baseado em [5], por ser um modelo bem detalhado e inovador [7]. Nele a velocidade é considerada constante. O principal fator determinante para estimativa do consumo de combustível é o peso total transportado. Adaptamos este modelo para o estudo de caso da cooperativa de materiais recicláveis CORESO, localizado na cidade de Sorocaba – SP, utilizando a metaheurística ACS. O cenário de estudo referente ao teste do GVRP é baseado em [8] e a resolução do modelo matemático proposto por [5].

Usaremos aqui a notação de [9], que estabelece o Problema de Roteamento de Veículos clássico retratado por um grafo orientado completo $G = (N, E)$, onde $N = C \cup \{0, n + 1\}$, $C = \{1, \dots, n\}$ é o conjunto de nós (associado aos clientes), e

0 e $n + 1$ são os nós que representam o depósito. O conjunto $E = \{(i, j) : i, j \in N, i \neq j, i \neq n + 1, j \neq 0\}$ são os arcos que ligam os nós, sendo que nenhum nó termina em 0 e começa no nó $n + 1$, necessariamente todos começam em 0 e terminam em $n + 1$ (depósito). Os custos estão associados com os arcos $(i, j) \in E$ e cada cliente i está associado a uma demanda D_i .

O primeiro passo para descrever o modelo são os pontos de coleta, sua localização em coordenadas geográficas, suas respectivas demandas e as distâncias entre eles. O próximo passo para o entendimento do modelo é o desenvolvimento do FCR (*Fuel Consumption Rate*), valor determinado a partir da função inversa da autonomia dos veículos, ou seja, quantos litros são necessários para o veículo percorrer um quilômetro, dependendo do peso da carga. Utilizamos uma função linear para descrever o FCR do peso coletado de materiais recicláveis em cada ponto, assim como proposto por [5] e apresentado na Equação (1).

$$\rho_{ij} = \rho(Q_{ij}) = \rho_0 + \alpha Q_{ij}, \quad \alpha = \frac{\rho^* - \rho_0}{Q}. \quad (1)$$

Note que ρ_0 e ρ^* são os parâmetros do FCR do caminhão vazio e carregado, respectivamente. A variável Q_{ij} corresponde à quantidade carregada pelo caminhão do nó i para o nó j , $i, j \in N$. Q é a capacidade total de carga do caminhão.

De forma geral as equações do problema de roteamento de veículos estão restritas a alguns fatores, como (1) a carga transportada não pode ultrapassar a capacidade total do veículo; (2) cada cliente é atendido apenas uma vez e (3) as rotas devem iniciar e terminar no nó 0 (depósito).

Definindo c_0 como preço do combustível, d_{ij} a distância do nó i ao nó j , D_i é a demanda do nó i , F o custo fixo para transporte, y_{ij} o peso que um caminhão transporta de i para j e x_{ij} a variável binária de decisão (se $x_{ij} = 1$ o veículo percorre os nós i até j ; caso contrário $x_{ij} = 0$), a formulação do GVRP baseado em [5] com objetivo de minimização de consumo de combustível (Problema (2)), fica:

$$\min \sum_{j=1}^n Fx_{0j} + \sum_{i=0}^n \sum_{j=0}^n c_0 d_{ij} (\rho_{ij} x_{ij} + \alpha y_{ij}) \quad (2a)$$

$$\text{s.t.} \quad \sum_{j=0}^n x_{ij} = 1 \quad \forall j = 1, \dots, n \quad (2b)$$

$$\sum_{j=0}^n x_{ij} - \sum_{j=0}^n x_{ji} = 0 \quad \forall j = 1, \dots, n \quad (2c)$$

$$\sum_{\substack{j=0 \\ j \neq i}}^n y_{ij} - \sum_{\substack{j=0 \\ j \neq i}}^n y_{ji} = D_i \quad \forall j = 1, \dots, n \quad (2d)$$

$$y_{ij} \leq Qx_{ij} \quad \forall i, j = 1, \dots, n \quad (2e)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j = 1, \dots, n \quad (2f)$$

Cada restrição de (2) tem uma finalidade: (2a) é a função objetivo (minimização do custo de combustível); (2b) indica que cada cliente só pode ser visitado por um único veículo; (2c) estabelece que o veículo deve visitar um cliente (nó) e sair; (2d) representa o aumento da carga após o veículo visitar um nó, sendo que o valor é igual à demanda do nó visitado; (2e) determina que as cargas não podem ultrapassar a capacidade do veículo e (2f) é a variável binária de decisão.

Nosso modelo apresenta algumas diferenças em relação a [5]:

- Na Equação (2d), invertemos os somatórios, pois o veículo deixa o depósito vazio e retorna cheio, coletando materiais durante a execução do percurso; para o autor, a formulação é desenvolvida baseada em caminhões de distribuição de água que saem carregados e retornam vazios ao depósito;
- Também simplificamos a função objetivo, Equação (2a), pois desconsideramos o seu custo fixo e focamos somente no custo associado ao consumo de combustível, sendo ρ_{ij} o FCR em cada trecho da rota de acordo com a Equação (1).

$$\sum_{i=1}^n \sum_{j=1}^n c_0 \rho_{ij} d_{ij} x_{ij} \quad (3)$$

A diferença entre a Equação (2a) e a Equação (3) é de que a primeira calcula o custo para uma frota de veículos homogêneos, enquanto a outra realiza o cálculo do custo de um único veículo.

2.2 Ant Colony System (ACS)

O ACS pode ser aplicado em problemas que incluem grafos, onde o nó inicial (depósito) é o ponto de partida para o desenvolvimento de um circuito fechado, percorrendo todos os nós e retornando ao inicial, respeitando certas restrições. As formulações do ACS estão expostas a seguir, como descritas por [10] e [11].

A Equação (4) representa a probabilidade de escolha (p_{ij}^k) da formiga k selecionar um novo nó j para incluir na solução

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [v_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}(t)]^\alpha \cdot [v_{il}]^\beta}, & \text{se } j \in N_i^k \\ 0 & \text{caso contrário} \end{cases} \quad (4)$$

onde $\tau_{ij}(t)$ é a quantidade de feromônio presente na trilha (i, j) no tempo t , $v_{ij} = 1/d_{ij}$ é a visibilidade do nó j que está relacionado ao percurso do nó i , representando a atratividade da formiga em percorrer esse trecho, α e β são parâmetros para determinar a influência do feromônio e da informação heurística, k , o índice de cada formiga e N_i^k , a vizinhança viável da formiga k , sendo este o conjunto de nós ainda não visitados pela formiga k .

O feromônio $\tau_{ij}(t)$ associado com a aresta (i, j) atua de duas formas:

1. Na evaporação, evitando o acúmulo infinito e favorecendo o esquecimento de decisões ruins realizadas;

2. No depósito de feromônio de todas as formigas que passaram sobre a aresta (i, j) .

Após a realização das rotas pelas formigas há a atualização do feromônio. A primeira etapa é a estimativa da quantidade de feromônio depositado sobre as arestas (i, j) , representado na Equação (5)

$$\Delta\tau_{ij}^k(t) = \begin{cases} Q/L^k(t), & \text{se } (i, j) \in S^k(t) \\ 0 & \text{caso contrário} \end{cases}, \quad (5)$$

sendo $\Delta\tau_{ij}^k$ a quantidade de feromônio depositado pela formiga k sobre a aresta (i, j) , Q a quantidade de feromônio secretado por uma formiga em cada interação, L_k a distância percorrida pela formiga k e S_k o conjunto de nós que são percorridos pela formiga k para construção da solução.

Por fim, a Equação (6) representa a atualização do feromônio τ_{ij} associado à aresta (i, j) .

$$\tau_{ij}(t+1) \leftarrow (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}^k(t) \quad (6)$$

onde a primeira parcela representa a evaporação de feromônio, com $\rho \in [0, 1]$ (taxa de evaporação) e a segunda parcela representa o seu depósito. O critério de parada da metaheurística é um número pré-definido de iterações ou a estagnação da solução.

Também implementamos a metaheurística Busca Tabu (BT), fornecendo uma memória à metodologia, que a impede de retornar a uma solução já desenvolvida [12]. Como apresentado por [13], soluções baseadas na metaheurística ACS associadas com a BT apresentam resultados competitivos em comparação com algoritmos renomados da literatura.

A metaheurística ACS desenvolvida neste trabalho baseou-se em [14].

3 Metodologia

3.1 Materiais

Os softwares utilizados para aplicação do modelo foram o Matlab versão R2010a para implementação dos algoritmos, Planilha Eletrônica Excel® versão 1610 para tabulação dos dados da cooperativa, ArcGis 10.3 para o georreferenciamento dos pontos de coleta e Google Earth Pro para obtenção das coordenadas geográficas dos pontos de coleta

O algoritmo, os modelos teste e estudo de caso foram implementados e compilados em um computador com processador Intel i7, 2ª geração, 3.4 GHz, com 8 GB de memória RAM e sistema operacional Microsoft Windows 7®.

3.2 Métodos

Primeiramente formulamos o GVRP com o objetivo de redução do consumo de combustível, de acordo com o modelo proposto por [5] e incorporando as

alterações já discutidas. Sua solução deu-se através da metaheurística ACS implementada no software Matlab.

Desenvolvido o algoritmo, buscou ajustar os parâmetros do ACS de acordo com as instâncias utilizadas por [15] e propostas por [16]. Para ajuste dos parâmetros utilizamos três instâncias de [16]: M-n101-k10, M-n121-k7 e a M-n151-k12, onde n representa a quantidade de nós e k a quantidade inicial de formigas. Para analisar a eficiência e funcionalidade do GVRP implementado neste trabalho com um modelo da literatura, utilizou-se as instâncias e dados de [8].

Visando identificar os parâmetros que melhor se adequam ao modelo desenvolvido comparou-se com os propostos por [15,14], o melhor resultado concluído por [17] e com três parâmetros ajustados empiricamente neste trabalho. Os três parâmetros foram definidos como A, B e C, sendo A ($\alpha = 0,5$; $\beta = 6$; $\rho = 0,3$; $q_0 = 40$), B ($\alpha = 0,3$; $\beta = 6$; $\rho = 0,3$; $q_0 = 60$) e C ($\alpha = 0,3$; $\beta = 6$; $\rho = 0,28$; $q_0 = 60$).

Para todos os cenários e parâmetros, até mesmo para o estudo de caso, definimos 3000 iterações como critério de parada.

O cálculo das distâncias entre os pontos de coleta e o depósito para o modelo teste e para o estudo de caso foi através da implementação, em ambiente Matlab, da distância euclidiana [18], que considera a curvatura do planeta.

3.3 Estudo de Caso

Para o estudo de caso o primeiro passo foi o levantamento de dados operacionais junto à cooperativa de materiais recicláveis CORESO da cidade de Sorocaba – SP, através de reuniões periódicas (uma a cada dois meses), identificando as ruas em que a cooperativa realiza as coletas, o cronograma diário de trabalho dos caminhões para cada semana, o peso total mensal coletado de material e discussão sobre a possibilidade de aplicação dos resultados finais do presente trabalho no seu cronograma, elaborando um novo sequenciamento a ser seguido e executado.

A cooperativa CORESO realiza a coleta de materiais recicláveis, principalmente na zona leste de Sorocaba, atendendo um total de 230 ruas (nós) de segunda à sexta-feira e, aproximadamente, 9000 pontos de coleta, entre casas (coleta de porta em porta) e geradores coletivos. Para a implementação do modelo, levantamos as coordenadas geográficas de todas as ruas em que são realizadas as coletas, com auxílio do software Google Earth Pro.

Implementamos os dados de entrada e os melhores parâmetros estudados do ACS (A e C) no algoritmo desenvolvido para as ruas de cada dia da semana (segunda a sexta-feira).

A cooperativa possui dois caminhões iguais com capacidade de 4.000 quilogramas, assim para o estudo de caso consideramos uma frota homogênea para o desenvolvimento das rotas. Para estimar a emissão de poluentes levantamos as características relacionadas com a emissão de GEE. As quantidades estimadas destes poluentes foram obtidas através de [19]. O veículo utilizado pela cooperativa não aparece na tabela de etiquetagem de veículos do INMETRO, portanto

consideramos aquele com características semelhantes, o Citroën Jumper motor 2.3 16V.

De acordo com informações fornecidas pela cooperativa, os veículos possuem consumo médio de 7,5 km/L, assim adotamos consumo de 9 km/L e 6 km/L para o veículo vazio e cheio, respectivamente; estes valores foram considerados aceitáveis por parte da administração. Com isso, para o cálculo do FCR do modelo proposto, temos $\rho_0 = 1/9$ L/km = 0.1111 L/km e $\rho^* = 1/6$ L/km = 0.1667 L/km. De acordo com a Equação (1), temos que $\alpha \cong 1 \times 10^{-5}$, resultando na equação do FCR: $\rho = 1 \times 10^{-5}Q + 0,111$.

Para cálculo do custo das rotas associados com o combustível, o preço considerado foi de R\$ 2,999 para o Diesel; de acordo com a ANP (Agência Nacional do Petróleo, Gás Natural e Combustível) no dia 01/03/2017 para a Petrobrás Distribuidora S.A. na cidade de Sorocaba – SP [20].

Finalmente, desenvolvemos a representação espacial das rotas, baseada nas soluções calculadas e nos pontos de coleta georreferenciados. Para isso utilizamos as bases cartográficas do município de Sorocaba em escala referência de 1:10.000, do Estado de São Paulo em escala de 1:50.000 e das unidades federativas do Brasil em escala de 1:2.500.000 [21].

Após o desenvolvimento das melhores rotas, ou seja, as que consomem menos combustível (emitindo menos poluente) elaboramos um itinerário para a cooperativa aplicar na execução da coleta de materiais recicláveis, e através de reuniões apresentamos nossos resultados e explanamos sobre o sequenciamento desenvolvido para o motorista e administração.

4 Resultados e Discussão

4.1 Instâncias e Testes

Para estudo dos melhores parâmetros para o modelo, realizamos alguns testes em diferentes cenários, nos baseando em parâmetros da literatura. A Tabela 1 resume as soluções obtidas com os parâmetros estabelecidos para cada instância da literatura. As colunas apresentam, na ordem, os parâmetros que são aplicados na metaheurística do ACS, as instâncias utilizadas, a solução apresentada e o erro percentual em relação à melhor solução obtida para cada instância. Para aplicação do modelo, foram inicializadas 10 formigas para a instância M-n101-k10, para a M-n151-k12 foram 12 formigas, para M-n121-k7 foram 7 formigas e para o caso de [8], 8 formigas.

O pior parâmetro analisado é o proposto por [15], como visto na Tabela 1, provavelmente por ser um dos primeiros trabalhos a abordar o tema. Outro ponto interessante é que o modelo proposto por [14] também trata sobre a aplicação da metaheurística ACS para um GVRP, porém o parâmetro deste autor apresentou a maior diferença em relação ao modelo exato de [8]; esta divergência deve-se possivelmente ao fato do cenário utilizado por este autor ser diferente ao de [8] e por este tratar de um ACS híbrido baseado na metaheurística LNS, que difere do modelo desenvolvido no presente trabalho.

Tabela 1. Solução das instâncias de acordo com cada parâmetro.

Parâmetros	Instâncias	Resultado	Diferença (%)
[15]	M-n101-k10	1119,13	28,46
	M-n151-k12	1513,50	46,53
	M-n121-k7	1512,31	46,26
	[8]	77,16	3,58
[14]	M-n101-k10	1103,43	26,67
	M-n151-k12	1513,50	46,53
	M-n121-k7	1523,81	47,37
	[8]	77,79	4,43
[17]	M-n101-k10	1115,08	28,00
	M-n151-k12	1510,16	46,20
	M-n121-k7	1389,15	34,35
	[8]	77,16	3,58
A	M-n101-k10	937,98	7,67
	M-n151-k12	1257,64	21,76
	M-n121-k7	1371,96	32,68
	[8]	75,52	1,38
B	M-n101-k10	1067,43	22,53
	M-n151-k12	1121,20	8,55
	M-n121-k7	1152,53	11,46
	[8]	76,65	2,90
C	M-n101-k10	985,82	13,16
	M-n151-k12	1164,65	12,75
	M-n121-k7	1062,06	2,71
	[8]	74,90	0,55

A Tabela 2 apresenta as diferenças entre o modelo com resolução exata com o software GAMS 24.3 e com a metaheurística de acordo com o cenário de [8], sendo utilizados os três parâmetros ajustados empiricamente na aplicação do ACS, pois cada parâmetro se ajusta melhor em cada dia. O custo total da rota para o modelo exato é de R\$ 74,49 e o tempo computacional necessário para execução desta solução foi de 7199,82 segundos.

Tabela 2. Diferenças do custo da rota e tempo computacional para as formulações exata e com ACS

	Custo da Rota (R\$)	Tempo Computacional (s)	Diferença (%) Custo da Rota	Diferença (%) tempo Computacional
ACS parâmetro A	75,52	22,83	1,38	-99,68
ACS parâmetro B	76,65	24,09	2,90	-99,67
ACS parâmetro C	74,90	23,47	0,55	-99,67

De acordo com a Tabela 2 é possível identificar o elevado ganho de tempo computacional que a utilização da metaheurística no modelo de GVRP apresenta, sendo necessário 0,32% do tempo para obtenção da solução final com diferença de 1,38% em relação ao modelo com resolução através do software GAMS e utilizando o parâmetro A.

4.2 CORESO

A Tabela 3 apresenta os custos para cada dia da semana desenvolvidos de acordo com o Problema (2) e utilizando dos três parâmetros ajustados no presente

trabalho para a solução encontrada através da metaheurística ACS. Os valores em negrito são os que apresentam os menores custos, sendo as melhores soluções para o estudo de caso proposto.

Tabela 3. Menores custos para cada dia da semana em relação aos três parâmetros ajustados empiricamente

	Parâmetros		
	A	B	C
Segunda	7,46	7,55	7,61
Terça	4,27	4,29	4,28
Quarta	4,11	4,11	4,11
Quinta	4,96	4,73	4,64
Sexta	4,00	4,09	4,09
Total	24,80	24,77	24,73

Para cada dia da semana adotamos a melhor solução (em negrito) em relação aos parâmetros testados. Ou seja, para segunda, terça, quarta e sexta-feira usamos a solução dada pelo parâmetro A e para quinta-feira, o parâmetro C. A Tabela 4 apresenta as rotas obtidas e a Figura 1 sua representação gráfica.

Tabela 4. Rotas para o estudo de caso com a carga coletada e o custo

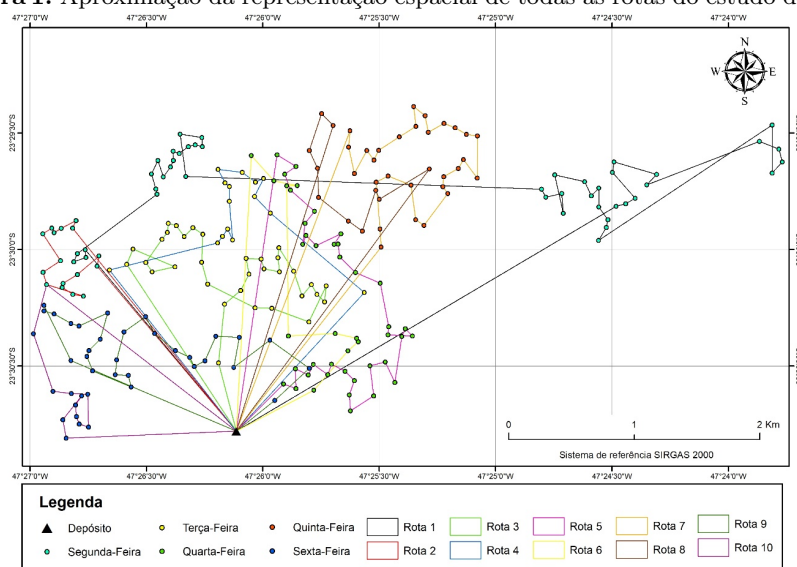
Dias da semana	Rotas	Carga coletada (kg)	Custo (R\$)
Segunda	1 0-15-16-17-14-20-19-18-6-5-4-3-7-26-25-24-13-22-12-11-10-23-9-21-8-33-34-1-27-39-28-38-35-32-29-30-37-36-2-31-54-58-57-56-0	3986,20	5,77
	2 0-53-55-52-51-59-48-49-47-46-50-40-45-41-42-44-43-0	1326,00	1,69
Terça	3 0-29-30-31-41-28-25-26-24-27-23-22-21-7-48-44-46-43-5-2-3-4-6-9-13-14-10-11-15-12-16-42-45-47-0	3995,90	2,31
	4 0-40-20-19-17-18-35-32-33-1-36-37-39-34-38-8-0	1316,30	1,97
Quarta	5 0-43-44-45-37-36-35-34-40-39-28-41-42-29-27-30-26-24-25-23-1-4-22-5-6-8-7-21-20-9-10-11-16-15-14-17-0	3984,20	2,56
	6 0-19-13-18-12-31-32-3-2-33-38-0	1328,10	1,56
Quinta	7 0-33-29-26-28-27-18-11-7-10-9-6-4-3-2-1-12-5-8-15-14-13-21-22-17-25-24-23-38-0	3992,10	2,65
	8 0-16-19-20-37-36-34-35-30-32-31-0	1320,10	1,97
Sexta	9 0-35-25-26-28-30-31-14-33-34-29-40-39-37-36-13-10-4-12-11-1-9-6-8-5-7-3-2-0	3994,10	2,33
	10 0-15-22-19-20-17-21-18-38-23-16-27-24-0	1318,10	1,68

Elaboramos duas rotas para cada dia da semana, executadas por cada um dos dois veículos da cooperativa, atendendo todas as ruas e grandes geradores pré-definidos pela CORESO. O custo total das rotas semanais foi de R\$ 24,50. A Tabela 5 apresenta a emissão de gases do efeito estufa para cada dia da semana baseado nos dados de [19].

Após elaboração das rotas conforme proposto, apresentamos os resultados para a administração da cooperativa CORESO. Neste encontro também expu-

Tabela 5. Emissão de poluentes (em gramas) em cada dia da semana.

Dias da semana	CO ₂	NO _x	CO	NMHC
Segunda-feira	4910,487	5,396	0,878	0,373
Terça-feira	2813,470	3,092	0,503	0,214
Quarta-feira	2706,708	2,974	0,484	0,206
Quinta-feira	3051,932	3,354	0,545	0,232
Sexta-feira	2634,039	2,894	0,471	0,200
Total	16116,636	17,710	2,880	1,226

Figura 1. Aproximação da representação espacial de todas as rotas do estudo de caso.

semos todo o projeto e os resultados para os motoristas, pois a decisão final e a possibilidade de aplicação das rotas são de acordo com seu comprometimento. Eles aceitaram praticar o nosso modelo durante três semanas para avaliar a efetividade das rotas desenvolvidas. Durante este período imprevistos ocorridos com um dos motoristas afetaram as conclusões sobre as novas rotas, porém em alguns dias o nosso sequenciamento foi aplicado.

A cooperativa comunicou que houve aplicação das rotas durante este período de três semanas com algumas alterações, e o motorista informou que houve ligeira melhora em relação à distância percorrida, não indicando redução no consumo diretamente, mas que pode influenciar positivamente. Nas últimas conversas com a administração nos foi informado do interesse de aplicação das rotas em seu itinerário, porém no momento em que houver regularização com o caso do motorista, implementando progressivamente as rotas sugeridas.

5 Conclusão

A utilização da metaheurística ACS apresentou resultados satisfatórios e de grande potencial de aplicação em trabalhos de GVRP, de forma que em determinados cenários a solução com a metaheurística apresentou valores próximos ao ideal, com erro de 0,55%, com necessidade de 0,33% do tempo computacional para sua obtenção em relação ao modelo exato resolvido através do software GAMS. A etapa de ajustes dos parâmetros é fundamental e de extrema importância para obtenção dos resultados finais, sendo esta uma etapa que requer maior tempo de concentração, pois os ajustes são sensíveis, podendo interferir nas soluções do algoritmo.

O cálculo da emissão de gases do efeito estufa introduz a possibilidade de gerenciar e analisar a emissão destes gases gerados na etapa do transporte. Avaliando principalmente a emissão de gás carbônico (CO₂) abre-se um possível novo ramo para a cooperativa, o de créditos de carbono, baseado no protocolo de Quioto (1997). Além disso, o modelo proposto é uma ferramenta de suporte a gestão de um empreendimento que realiza a reciclagem, evitando que resíduos que ainda tenham potencial de reutilização de seu material sigam para disposição final em aterros sanitários, o qual tem associado uma série de impactos ambientais negativos.

Os benefícios do modelo estão relacionados à elaboração e planejamento de um cronograma fixo para a cooperativa, que além de auxiliar na administração fornecendo suporte para tomadas de decisões em casos de mudanças, como acréscimos ou decréscimos da quantidade de pontos de coleta, também promove facilidade de integração de novos motoristas, como na contratação de novos funcionários ou alterações de cargos, e permite que em mudanças, como aumento ou redução do preço do combustível, não alterem significativamente nosso itinerário proposto.

Referências

1. Souza, M.T.S.d., Paula, M.B.d., Souza-Pinto, H.d.: O papel das cooperativas de reciclagem nos canais reversos pós-consumo. *Revista de Administração de Empresas* **52** (04 2012) 246 – 262
2. CEMPRE (Compromisso empresarial para reciclagem): Ciclossoft 2016 (2016) <http://cempre.org.br/ciclossoft/id/8>, Acessado em 05/07/2017.
3. Ribeiro, H., Besen, G.R.: Panorama da coleta seletiva no brasil: desafios e perspectivas a partir de três estudos de caso. *INTERFACEHS – Revista de Saúde, Meio Ambiente e Sustentabilidade* **2**(4) (2007)
4. Braga, B., Hespanhol, I., Conejo, J.G.L., Mierzwa, J.C., de Barros, M.T.L., Spencer, M., Porto, M., Nucci, N., Juliano, N., Eiger, S.: *Introdução à Engenharia Ambiental*. Second edn. Pearson Prentice Hall (2005)
5. Xiao, Y., Zhao, Q., Kaku, I., Xu, Y.: Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers and Operations Research* **39**(7) (2012) 1419 – 1431

6. Demir, E., Bektaş, T., Laporte, G.: A review of recent research on green road freight transportation. *European Journal of Operational Research* **237**(3) (2014) 775 – 793
7. Lin, C., Choy, K., Ho, G., Chung, S., Lam, H.: Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications* **41**(4) (2014) 1118 – 1138
8. Santos, G.T., Cantão, L.A.P., Cantão, R.F.: Vehicle routing problem with fuel consumption minimization: a case study. In Maturana, S., ed.: *Proceedings of the XVIII Latin-Iberoamerican Conference on Operations Research, CLAIO 2016*. (2016) 802–809 ISBN: 978-956-9892-00-4.
9. Arenales, M.N., Armentano, V., Morabito, R., Yanasse, H.: *Pesquisa Operacional*. Elsevier (2007)
10. Dorigo, M., Gambardella, L.M.: Ant colony system: A cooperative learning approach to the traveling salesman problem. *Trans. Evol. Comp* **1**(1) (April 1997) 53–66
11. Lopes, H.S., de Abreu Rodrigues, L.C., Steiner, M.T.A., eds.: *Otimização por Colônia de Formigas*. In: *Meta-Heurísticas em Pesquisa Operacional*. 1 edn. Omnipax, Curitiba, PR (2013)
12. Gomes, A.: Uma introdução ao a busca tabu, Departamento de Ciência da Computação, Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo (2009) <http://www.ime.usp.br/gold/cursos/2009/mac5758/AndreBuscaTabu.pdf>, Acessado em 20/06/2017.
13. De la Cruz, J.J., Paternina-Arboleda, C.D., Cantillo, V., Montoya-Torres, J.R.: A two-pheromone trail ant colony system—tabu search approach for the heterogeneous vehicle routing problem with time windows and multiple products. *Journal of Heuristics* **19**(2) (Apr 2013) 233–252
14. Adiba, E., Messaoud, E., Ahemd, E.: A hybrid ant colony system for green capacitated vehicle routing problem in sustainable transport. *Journal of Theoretical and Applied Information Technology* **54** (08 2013) 198–208
15. Solomon, M.M.: Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.* **35**(2) (April 1987) 254–265
16. Christofides, N., Mingozzi, A., Toth, P.: The vehicle routing problems. In Christofides, N., Mingozzi, A., Toth, P., Sandi, C., eds.: *Combinatorial Optimizations*. John Wiley & Sons, Nova Iorque (1979)
17. Darquennes, D.: *Implementation and applications of ant colony algorithms*. Master's thesis, Facultés Universitaires Notre-Dame de la Paix, Namur – Institut d'Informatique (2005)
18. Hartung, G.E.: Distâncias entre dois pontos – uca http://portaldoprofessor.mec.gov.br/fichaTecnicaAula.html?pagina=espaco%2Fvisualizar_aula&aula=30254&secao=request_locale=es>, Acessado em 12/08/2015.
19. INMETRO – Instituto Nacional de Metrologia, Qualidade e Tecnologia: Programa brasileiro de etiquetagem – PBE http://www.inmetro.gov.br/consumidor/pbe/veiculos_leves_2016.pdf, Acessado em 20/06/2016.
20. ANP (Agência Nacional do Petróleo, Gás Natural e Biocombustível): Sistema de levantamento de preços http://www.anp.gov.br/preco/prc/Resumo_Por_Municipio_Index.asp, Acessado em 25/05/2017.
21. DATAGEO: Infraestrutura de dados espaciais ambientais do Estado de São Paulo <http://datageo.ambiente.sp.gov.br>, Acessado em 05/07/2017.

Visualização de Dados de Roteamento para Cidades Inteligentes

Heder Dorneles Soares¹, Edcarllos Santos¹, José Viterbo¹, Hugo Fernandes²,
and Igor Machado Coelho²

¹ Universidade Federal Fluminense, Instituto de Computação
{hdorneles,esantos,viterbo}@ic.uff.br

² Universidade do Estado do Rio de Janeiro, Instituto de Matemática e Estatística
{hugo.fernandes,igor.machado}@ime.uerj.br

Resumo Com o aumento do número de dispositivos conectados à Internet, temos diversos objetos do cotidiano fazendo parte da rede (Internet das Coisas), como carros, drones, dispositivos móveis em geral, casas e até mesmo cidades inteiras. Neste contexto de Cidades Inteligentes, novos problemas surgem com a convergência de diversas áreas relacionadas à computação, visualização e tratamento de grandes volumes de dados de interesse público. O transporte urbano é uma atividade fundamental para as relações entre as diversas entidades em uma Cidade Inteligente. O processo de roteamento um conjunto de decisões multidisciplinares, em que diversos algoritmos eficientes (de otimização) tem sido propostos na literatura, bem como diversos arcabouços focados em comunicação entre os nós. Neste contexto, é necessário que os dados coletados de sistemas de localização geográfica sejam obtidos (através de sensores ou bases de dados), processados e entregues de forma compreensível ao usuário ou tomador de decisão. Porém, poucos trabalhos na literatura tratam da integração das diversas ferramentas existentes, o que se torna um gargalo para o desenvolvimento de novas tecnologias acessíveis ao usuário comum. Neste trabalho, propomos a integração de diversos arcabouços especializados em roteamento e Cidades Inteligentes, com o objetivo de desenvolver um artefato de código-aberto que possa contribuir com o desenvolvimento de novas pesquisas na área.

Keywords: Visualização de Dados, Problemas de Roteamento, Internet das Coisas, Cidades Inteligentes

1 Introdução

Cidades Inteligentes (CI) têm sido o foco de diversos trabalhos recentes na literatura, apresentando desafios e novas possibilidades em CI [15], interfaces com tecnologias de otimização [11], tecnologias assistivas [2], e novas plataformas de visualização [3]. Do ponto de vista de Internet das Coisas [20], do inglês *Internet of Things* (IoT), existe uma convergência tecnológica causada pelo aumento do número de dispositivos conectados à Internet, em que os aparelhos operam

ao mesmo tempo como sensores e dispositivos de processamento e troca de informações.

Neste ecossistema digital, novas ferramentas surgem como pequenos blocos que propiciam a disseminação e integração das diversas tecnologias existentes. Entretanto, o desenvolvimento de sistemas para ambientes inteligentes sempre conta com desafios multidisciplinares. Listamos três grandes desafios relacionados ao tema deste trabalho:

- Redes e Comunicação: coleta de dados em diferentes tipos de tecnologia como 3G, 4G, WiFi; suporte a desconexão e troca de redes (*handover*); e desenvolvimento de Middleware para IoT [10]
- Inteligência Computacional: inclui a busca de caminhos mínimos e planejamento de viagem (*route planning*), utilizando algoritmos clássicos como o de Dijkstra e variantes com A* [14], tendo como outra vertente a resolução de alguma variante NP-Difícil do Problema de Roteamento e Veículos [8], usualmente por meio de estratégias heurísticas e metaheurísticas [13]
- Visualização de dados: interface com o usuário, integrando arcabouços visuais para visualização de mapas por meio linguagens de programação web (usualmente via integração com navegadores)

Propomos um sistema de visualização de rotas em mapas integrado com a base de mapas aberta do OpenStreetMap, para visualização de rotas e projetado para integração futura com um *middleware* de IoT para monitoramento e distribuição de dados em cidades inteligentes. De forma a fornecer através de serviços na nuvem, apresentamos a ideia de um sistema completo de monitoramento de nós móveis, com heurísticas para otimização e tomada de decisão, utilizando uma interface de visualização de informações.

O restante do artigo é organizado cinco seções (incluindo esta introdução). A Seção 2 apresenta uma revisão da literatura sobre trabalhos em cidades inteligentes, otimização para roteamento e visualização para sistemas de geolocalização. A Seção 3 a arquitetura proposta integrando comunicação, visualização e otimização em cidades inteligentes. Na Seção 4, o sistema proposto é apresentado, bem como uma integração com um resolvidor de otimização para roteamento. Finalmente, a Seção 5 apresenta as conclusões do trabalho e perspectivas para trabalhos futuros.

2 Revisão de Literatura

Pesquisas no tópico de Cidades Inteligentes tem buscado estratégias tecnológicas para melhorar a qualidade de vida, e resolver problemas causados pelo crescimento da população e das áreas urbanas. Diversos trabalhos exploram tecnologias de comunicação para melhorar os serviços aos cidadãos e agregar valor à administração pública [15].

Diversas aplicações em monitoramento de veículos (caminhões, carros, bicicletas, drones) e logística/despacho aplicam conceitos de computação móvel (ou redes móveis), o que implica em comunicação, compartilhamento de dados e

coordenação entre um número muito grande de nós móveis [10]. Em especial, diversos problemas de otimização e roteamento de veículos podem ser encontrados na literatura [19], considerando restrições como capacidade do veículo, frota homogênea/heterogênea, limitada/ilimitada, demandas de entrega/coleta, janelas de tempo e fracionamento de entregas. Tais problemas usualmente consideram dados estáticos, considerando apenas a distância ou tempo de deslocamento entre os elementos.

Na literatura, o custo (distância ou tempo) entre um dado par de nós é modelada por uma simplificação da realidade, geralmente considerando apenas a distância Euclidiana entre dois pontos ou o tempo para percorrer a distância Euclidiana com velocidade constante. Assim, o planejamento da rota (mesmo com apenas dois pontos) pode se tornar uma tarefa complexa quando são consideradas vias de acesso e sentido das ruas [9]. O problema se torna ainda mais complexo quando ciclovias, vias exclusivas para pedestres e diferentes modais de transporte são combinados em uma única rota [18]. Neste caso, os autores utilizam uma generalização do algoritmo clássico de Ford-Moore-Bellman [4] através de grafos coloridos que representam redes de transporte multimodal.

A Figura 1 apresenta uma comparação entre os dados de otimização com distância Euclidiana (considerados na vasta maioria dos trabalhos da literatura) e uma abordagem que inclui o planejamento da rota. Embora possa existir uma grande diferença entre os valores de tempo e distância encontradas, tais distâncias e tempos após o planejamento da rota podem ser pré-calculados para cada par de nós e armazenados em uma matriz de ordem quadrática.



Figura 1: Comparação entre dados da literatura que consideram apenas a distância euclidiana ou o planejamento da rota [Fonte: autoria própria]

Esta abordagem é capaz de compatibilizar o uso distâncias Euclidianas através de matrizes para diversos casos, mas vale ressaltar que existem algoritmos na literatura que exploram informações Euclidianas durante o processo de roteamento. Rios et al. [17] apresenta um algoritmo de roteamento que é acelerado quando são utilizadas informações Euclidianas juntamente com arquiteturas de computação de alta performance, não aceitando como entrada uma matriz quadrática de custos. Para a resolução do clássico Problema do Caixeiro Viajante, que consiste em visitar um conjunto de cidades (sem repetição) e retornar à origem com um custo mínimo, existem algoritmos de otimização especializados no uso de distâncias Euclidianas, como o solver Concorde [1].

Uma aplicação interessante no contexto de turismo em Cidades Inteligentes é baseada no *Team Orienteering Problem* [5], um problema NP-Difícil. Nesta aplicação, notas são dadas a diversos Pontos de Interesse da cidade e o processo de otimização visa encontrar rotas que maximizem o interesse dos turistas dentro das janelas de tempo disponíveis [12]. De acordo com os autores, os resultados do trabalho confirmam a importância da integração de sistemas de transporte público com o planejamento de viagens. Testes efetuados com dados da cidade de San Sebastián, localizada no País Basco, indicam que rotas de 4 dias são suficientes para visitar a maioria dos pontos de interesse da cidade.

Diversas ferramentas existem para resolver problemas distintos relacionados à visualização de rotas e aquisição de informações geográficas. Algumas ferramentas são proprietárias e outras são de código-aberto, dentre as quais citamos os seguintes serviços:

Geocoding é o processo de converter endereços como “Rua São Francisco Xavier, 524” para coordenadas geográficas, que seria $-22.9113381, -43.2384333$ neste caso. Google Geocoding API e Nominatim são exemplos do serviço.

Navegação identifica a estrada que o veículo está trafegando e fornece informações adicionais sobre a mesma, como por exemplo o limite de velocidade. Google Roads API e o Skobbler são exemplos do serviço.

Planejamento da Rota (*Route Planning*) calcula rotas entre localidades. Pode-se pesquisar por diversos modos de transporte, como veículo próprio, transportes públicos e até mesmo andando. Google Maps Directions e o Open Source Routing Machine são exemplos do serviço

Matriz de Distâncias/Tempos O Google Maps Distance Matrix API fornece uma matriz com tempo de viagem e distância entre múltiplos endereços. Porém há limitações de requisições e inerentes da latência de rede, o que pode inviabilizar o uso do mesmo para grandes quantidades de localidades. Neste sentido o OSRM também fornece tal funcionalidade com a vantagem de ser instalado localmente sem necessidade de uso da rede, sendo capaz processar grandes volumes de informação em um baixo tempo computacional.

Dentre os serviços apresentados, temos interesse do ponto de vista de levantamento de dados e apresentação de rotas para usuários os serviços de Matriz de Distâncias e Planejamento da Rota.

3 Arcabouço Proposto

A proposta de integração entre os diversos serviços é ilustrada pela Figura 2. O arcabouço considera a interação entre a ferramenta de visualização e dois serviços principais: coleta/distribuição de dados e o serviço de rotas. Por sua vez, o serviço de rotas opera com a integração entre um serviço de planejamento de rota (considerando ciclovias, vias de pedestres, etc) e um serviço de otimização que tem enfoque na resolução de Problemas de Roteamento de Veículos.

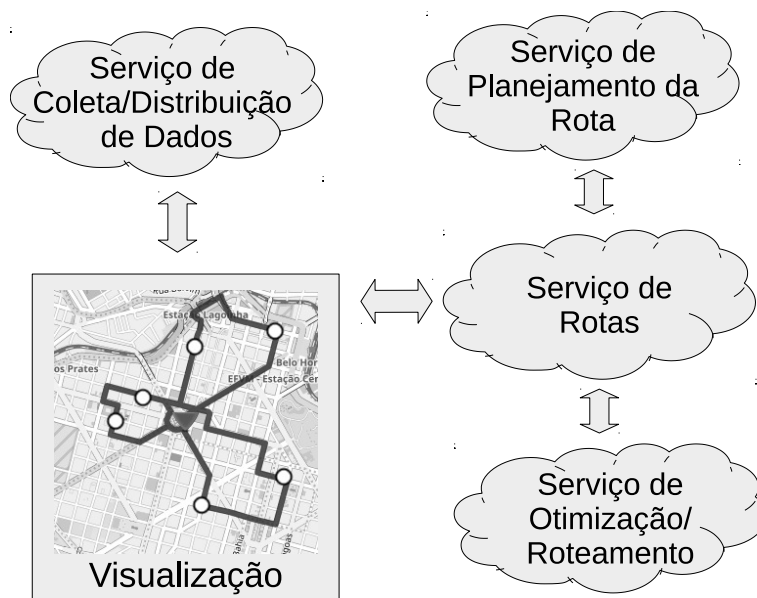


Figura 2: Proposta de integração entre diferentes serviços [Fonte: autoria própria]

A Figura 3 apresenta uma visão geral do arcabouço, sendo organizado em camadas. Neste modelo, a camada de visualização se conecta a um serviço de mapas, sendo na presente versão deste trabalho considerada a base de dados do OpenStreetMap³. Os dados podem ser obtidos de forma gratuita, no qual usuários voluntariamente editam locais, endereços e ruas. A proposta contempla uma integração futura com um serviço de aquisição de informações em tempo real para IoT, como o ContextNet [10], que fornece uma camada de transparente para

³ <http://openstreetmap.org>

acesso a sensores e agentes inteligentes que operam com diferentes protocolos de redes.

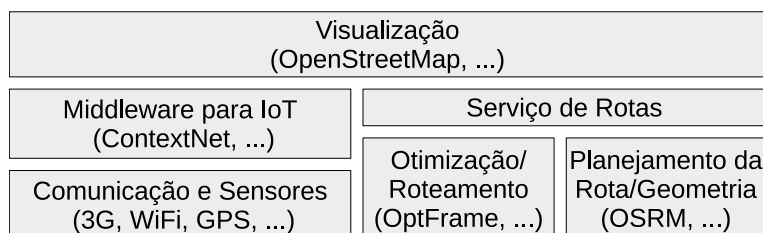


Figura 3: Proposta das camadas do sistema proposto [Fonte: autoria própria]

O serviço de rotas é implementado via a integração entre o Open Source Routing Machine (OSRM) e o arcabouço de otimização OptFrame [6], sendo este um projeto de código-aberto⁴ com implementação C++ de diversas metaheurísticas e algoritmos para resolução de problemas de roteamento de veículos, de classe NP-Difícil.

O OSRM efetua o planejamento da rota, sendo um projeto de código-aberto que implementa em C++ uma engine de alta performance para menores caminhos em redes de rotas. Utilizando algoritmos sofisticados de roteamento com os dados do projeto OpenStreetMap. Neste trabalho, utilizamos um fork temporário⁵ do OSRM gentilmente cedido para os autores do presente trabalho. Esse fork é desenvolvido pelo usuário `niemeier-PSI` e espera-se que seja incorporado ao projeto principal nos próximos meses⁶. Neste fork, foi incluída a possibilidade de construir uma matriz de distancias, dadas as coordenadas dos pontos. O OSRM atua no planejamento de rota (*Route Planning*) e é baseado no algoritmo *Contraction hierarchies* [16], usando técnicas para aumentar a velocidade do roteamento do *caminho mais curto* criando primeiro versões pré-computadas do grafo.

A integração entre os serviços é feita através da ferramenta Docker⁷, de forma a isolar os componentes de forma interoperável. Docker é uma ferramenta recentemente proposta para plataformas baseadas em container, sendo usado principalmente para eliminar possíveis diferenças de um mesmo software executado em máquinas diferentes. O Docker empacota o software de uma maneira que ele possa ser executado isolado do restante do Sistema Operacional, carregando somente configurações e bibliotecas necessárias para a execução do mesmo.

⁴ <https://sourceforge.net/projects/optframe/>

⁵ <https://github.com/niemeier-PSI/osrm-backend/tree/v5.2.7-psi>

⁶ <https://github.com/Project-OSRM/osrm-backend/pull/2764>

⁷ <https://www.docker.com/>

4 Simulação Utilizando o Sistema Proposto

Um protótipo do arcabouço proposto foi implementado em um ambiente Linux Ubuntu 16.04, utilizando as tecnologias apresentadas anteriormente. A ferramenta desenvolvida pode ser encontrada no repositório de código-aberto GitHub sob licença MIT: <https://github.com/Hugbro/cvrp-draw>.

Para ilustrar a proposta e testar o protótipo, sete pontos foram escolhidos na cidade de Belo Horizonte/MG, sendo o objetivo do roteamento criar três rotas limitadas a dois pontos em cada (o primeiro é considerado o ponto de partida, ou depósito, em um contexto de logística de entregas). A proposta de trabalhar com um exemplo de pequeno porte visa unicamente apresentar a integração entre as diferentes ferramentas, mas ressaltamos que problemas de grande porte tem sido resolvidos pelos autores em diversos trabalhos anteriores na literatura [7].

As Tabelas 1-2 apresentam as matrizes de distâncias e tempos relativas às coordenadas consideradas no experimento. Vale ressaltar que, mesmo após um extenso levantamento das tecnologias existentes, os autores não encontraram essa funcionalidade na versão padrão de nenhuma ferramenta de forma a atender demandas de grandes quantidades de nós (centenas e até milhares), o que ocorre frequentemente na prática. A solução a este problema se deve à integração da ferramenta com um fork específico do OSRM, desenvolvido pelo usuário GitHub `niemeier-PSI`.

Tabela 1: Matriz de distâncias obtida do OSRM (via fork)

Distâncias entre pontos (em metros)						
0	659.30	845.23	1526.19	1409.07	1309.22	1136.02
1149.15	0	776.26	2405.92	2055.26	2314.10	905.37
923.99	1554.49	0	1767.20	1645.08	2204.41	1542.01
1843.16	2208.42	2597.90	0	1169.12	1932.94	2983.70
849.19	1441.42	1694.43	899.14	0	1861.63	1985.22
2163.45	2054.13	2541.60	1969.86	2576.91	0	1294.22
1142.14	900.31	1382.76	2398.92	2048.25	2113.21	0

Tabela 2: Matriz de tempos obtida do OSRM (via fork)

Tempos entre pontos (em segundos)						
0	67.8	90.6	128.5	129.6	99.4	106.3
113.6	0	81.9	211.7	188.7	170	86.4
83.6	142.5	0	162.1	148.2	174.1	157.4
156.5	190	229.7	0	106.6	152.3	235.9
70.7	135.8	161.3	79.6	0	159.1	177
180.3	170.3	191.3	164.3	220.4	0	106.2
101.5	78.2	131.8	199.6	176.6	152.7	0

Um segundo experimento visava integrar o serviço de roteamento e otimização (via arcabouço OptFrame) com a ferramenta de visualização de rotas. A etapa de otimização foi integrada via linguagem C++ (nativa do arcabouço OptFrame), porém no futuro pretendemos efetuar a integração através da linguagem de modelagem CombView⁸, atualmente em desenvolvimento.

A Figura 4 apresenta a rota desenhada e exportada em formato gráfico utilizando informações do serviço de rotas desenvolvido. Diversos desafios foram encontrados para a obtenção do mapa final, pois as ferramentas existentes são desenvolvidas para navegadores, e recursos de extração de imagem (*static map*) deixam de funcionar quando elementos externos (como rotas e marcadores) são inseridos no mapa. Para contornar esse problema, a ferramenta de visualização foi integrada ao PhantomJS⁹. PhantomJS é um WebKit scriptável com uma API JavaScript. Tem suporte rápido e nativo para vários padrões web: DOM, CSS, JSON, Canvas e SVG. Em outras palavras: PhantomJS é um navegador web sem interface gráfica.

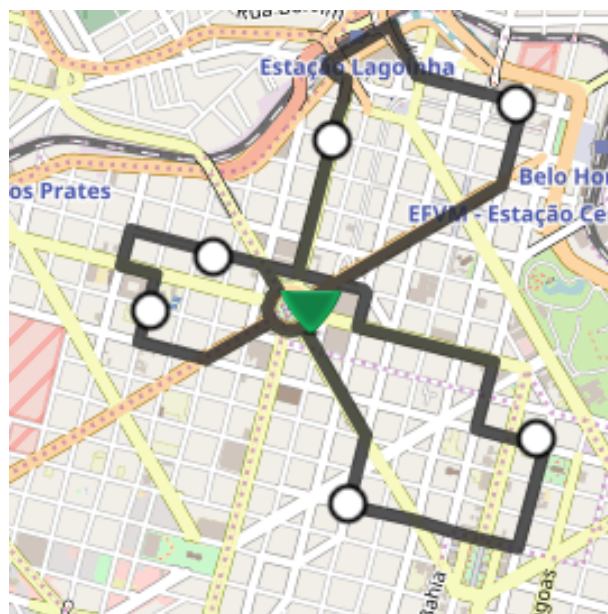


Figura 4: Exemplo de rota obtida pelo arcabouço proposto [Fonte: autoria própria]

⁸ <https://github.com/igormcoelho/language-combview>

⁹ <http://phantomjs.org/>

5 Conclusões e Trabalhos Futuros

Este trabalho apresenta uma proposta de arquitetura para aquisição de dados, roteamento e visualização de rotas em uma Cidade Inteligente. Embora diversas ferramentas de alta qualidade existam para cada uma destas vertentes, a integração entre elas é capaz de trazer grandes desafios.

Dentre os desafios elencados, apresentamos soluções para: aquisição e processamento de grandes porções de informação através da base do OpenStreetMaps e do OSRM (um fork específico fora do ramo principal do projeto OSRM foi considerado); integração da ferramenta de visualização com um arcabouço de otimização, independentemente da necessidade de um navegador.

Em uma Cidade Inteligente, problemas de roteamento estão ligados ao sistema de transporte público e particular, entrega/coleta de produtos, transporte multimodal, carros elétricos com pontos de recarga e roteamento de drones. Já existem diversos trabalhos na literatura com foco na resolução de problemas de otimização em cada uma dessas linhas, porém a maioria considera apenas dados estáticos (muitas vezes subestimados e fora da realidade). Neste sentido, propomos a integração futura da ferramenta com arcabouços de IoT como o ContextNet, abrindo a possibilidade de gerenciar a tomada de decisão com dados em tempo real, considerando dados dinâmicos e conectando as ferramentas de otimização existentes a agentes inteligentes de forma transparente.

Agradecimentos

Os autores agradecem às agências de fomento FAPERJ, CAPES e CNPq pelas bolsas concedidas, e ao usuário do GitHub `niemeier-PSI` por gentilmente ceder uma versão estável do código de cálculo de matrizes de distância para uso no OSRM.

Referências

1. Applegate, D.L., Bixby, R.E., Chvatal, V., Cook, W.J.: The traveling salesman problem: a computational study. Princeton university press (2011)
2. Barbosa, J., dos Santos, J.C., Mengue, L., Telles, M., Barth, M., Righi, R.: Uma simulação de uma cidade inteligente assistiva. In: Proceedings of the XLIX SBPO (2017)
3. Barbosa, J.L.V., dos Santos, J.C., Mengue, L., Telles, M.J., Barth, M.J.: Otimização multi-nível para projeto de redes de acesso para implementação de cidades inteligentes. In: Proceedings of the XLIX SBPO (2017)
4. Bellman, R.: On a Routing Problem. Quarterly of Applied Mathematics 16, 87–90 (1958)
5. Chao, I.M., Golden, B.L., Wasil, E.A.: The team orienteering problem. European Journal of Operational Research 88(3), 464 – 474 (1996), <http://www.sciencedirect.com/science/article/pii/0377221794002894>
6. Coelho, I.M., Ribas, S., Perché, M.H.d.P., Munhoz, P.L.A., Souza, M.J.F., Ochi, L.S.: Optframe: a computational framework for combinatorial optimization problems. In: Proceedings of the XLII SBPO (2010)

7. Coelho, V., Grasas, A., Ramalhinho, H., Coelho, I., Souza, M., Cruz, R.: An ils-based algorithm to solve a large-scale real heterogeneous fleet vrp with multi-trips and docking constraints. *European Journal of Operational Research* 250(2), 367 – 376 (2016), <http://www.sciencedirect.com/science/article/pii/S0377221715008899>
8. Dantzig, G.B., Ramser, J.H.: The truck dispatching problem. *Management Science* 6(1), 80–91 (1959), <https://doi.org/10.1287/mnsc.6.1.80>
9. Delling, D., Sanders, P., Schultes, D., Wagner, D.: Engineering route planning algorithms. *Algorithmics of large and complex networks* 5515, 117–139 (2009)
10. Endler, M., Baptista, G., Silva, L., Vasconcelos, R., Malcher, M., Pantoja, V., Pinheiro, V., Viterbo, J.: Contextnet: context reasoning and sharing middleware for large-scale pervasive collaboration and social networking. In: *Proceedings of the Workshop on Posters and Demos Track*. p. 2. ACM (2011)
11. Figueiredo, A., Veloso, I., Veloso, I., Souza, M., Oliveira, T., Veloso, V.M., Coelho, V., Coelho, V.: Uma visão multicritério para julgamentos em cidades e sociedades inteligentes. In: *Proceedings of the XLIX SBPO* (2017)
12. Garcia, A., Vansteenwegen, P., Arbelaitz, O., Souffriau, W., Linaza, M.T.: Integrating public transportation in personalised electronic tourist guides. *Computers & Operations Research* 40(3), 758 – 774 (2013), <http://www.sciencedirect.com/science/article/pii/S0305054811000967>, *transport Scheduling*
13. Gendreau, M., Potvin, J.Y.: *Handbook of Metaheuristics*, vol. 2. Springer (2010)
14. Goldberg, A.V., Harrelson, C.: Computing the shortest path: A search meets graph theory. In: *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. pp. 156–165. SODA '05, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2005), <http://dl.acm.org/citation.cfm?id=1070432.1070455>
15. Lourenço, H.R., Oliver, M., Oliveira, T., Coelho, V., Tavares, W.: Desafios operacionais e digitais para conectar cidadãos em cidades inteligentes. In: *Proceedings of the XLIX SBPO* (2017)
16. Luxen, D., Vetter, C.: Real-time routing with openstreetmap data. In: *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. pp. 513–516. GIS '11, ACM, New York, NY, USA (2011), <http://doi.acm.org/10.1145/2093973.2094062>
17. Rios, E., Ochi, L.S., Boeres, C., Coelho, V.N., Coelho, I.M., Farias, R.: Exploring parallel multi-gpu local search strategies in a metaheuristic framework. *Journal of Parallel and Distributed Computing* 111(Supplement C), 39 – 55 (2018), <http://www.sciencedirect.com/science/article/pii/S0743731517302009>
18. Shirabayashi, J.V., Yamakami, A., Silva, R.C., Shirabayashi, W.V.: Metodologias de soluções para o problema de redes multimodais aplicadas ao tráfego urbano. In: *Proceedings of the XLIX SBPO* (2017)
19. Vidal, T., Crainic, T., Gendreau, M., Prins, C.: Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research* 231(1), 1–21 (2013), <https://www.cirrelet.ca/DocumentsTravail/CIRRELT-2012-05.pdf>
20. Weiser, M.: The computer for the 21st century. *Mobile Computing and Communications Review* 3(3), 3–11 (1999)