

## Um Middleware para Provisionamento de Contextos para Redes Veiculares\*

Fabício A. Silva<sup>1,2</sup>, Thais Regina M. B. Silva<sup>2</sup>, Linnyer B. Ruiz<sup>3</sup> e Antonio A. F. Loureiro<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação - Universidade Federal de Minas Gerais (UFMG)

<sup>2</sup>Campus Florestal – Universidade Federal de Viçosa (UFV)

<sup>3</sup>Departamento de Informática - Universidade Estadual de Maringá (UEM)

{fabricio.asilva, loureiro, thaisrb}@dcc.ufmg.br, linnyer@gmail.com

**Resumo.** *As aplicações para redes veiculares são consideradas cientes de contexto pois precisam de informações de interesse sobre os veículos, motoristas, passageiros e o ambiente para operarem satisfatoriamente e assim beneficiarem seus usuários, seja indicando a melhor rota ou alertando sobre acidentes e congestionamentos. Dessa forma, essas aplicações precisam de um provedor de contextos responsável por coletar, analisar e disponibilizar esse tipo de dado para as mesmas. Além disso, as aplicações para redes veiculares apresentam uma demanda por contextos lógicos (isto é, contextos que não são coletados por sensores físicos) e são consideradas coletivas, uma vez que seus recursos podem ser compartilhados simultaneamente por dois ou mais usuários. Este trabalho apresenta um middleware, chamado ConProVa, composto por módulos capazes de lidar com diversas questões relativas ao provisionamento de contextos, incluindo a inferência de contextos lógicos. O ConProVa também possui funcionalidades para detectar e tratar situações de conflitos de interesse, comuns em aplicações cientes de contexto coletivas. Uma instância do middleware foi avaliada para uma aplicação de detecção de congestionamento causado por acidentes e os resultados mostraram que foi possível aumentar a satisfação dos usuários, reduzindo o tempo total de trajeto e a emissão de gás carbônico.*

**Abstract.** *Vehicular ad-hoc network applications are context-aware since they need environmental and local contexts to operate properly and help their users on many aspects, like indicating the best route to take or warning about accidents and traffic congestion situations. Therefore these applications need a context provider system responsible for collecting, analyzing, reasoning and making the contexts available to them. In addition to this, vehicular applications require logical contexts (i.e., contexts that are not collected directly from physical sensors) and are collective because their resources are shared among many users. In this work we propose a smart context provisioning middleware, called ConProVA, that, in addition to provide contexts to the applications, it is able to infer logical contexts and deal with conflicts of interest. An instance of ConProVA was evaluated through simulation considering an accident detection and avoidance application. The results showed that it was possible to improve the users' satisfaction by reducing total travel time and carbon dioxide emission when accidents occur.*

---

\*Este trabalho conta com o apoio financeiro do CNPq por meio do INCT NAMITEC, processo 573.738/2008-4

## 1. Introdução

O estudo das redes veiculares está se tornando mais comum a cada dia. Esse tipo de rede tem como objetivo permitir a comunicação sem fio entre veículos automotores e também entre veículos e estações fixas infra-estruturadas situadas nas margens das ruas e estradas [Campista et al. 2009]. Muitas aplicações podem ser desenvolvidas para as redes veiculares e assim trazer benefícios para os usuários, como por exemplo identificação e controle de congestionamento, assistência no controle de vagas de estacionamento, auxílio e alertas de riscos de acidentes, disponibilização de conteúdo interativo aos passageiros, dentre várias outras. A utilidade desse tipo de aplicação para a sociedade com o objetivo de se ter uma melhoria da qualidade e da segurança no trânsito está fazendo com que vários grupos de pesquisadores se dediquem a essa área.

As aplicações para redes veiculares são consideradas cientes de contexto. Para funcionarem corretamente e de acordo com as necessidades, as aplicações desenvolvidas para esse tipo de rede requerem dados de interesse chamados contextos dos veículos, de seus usuários e do ambiente. Exemplos de tais contextos incluem a velocidade, posição, direção, nível de combustível e aceleração do veículo, o nível de sonolência e as intenções dos motoristas, e o clima (chuvoso ou ensolarado, por exemplo) no ambiente em que o veículo se encontra. Sem essas e outras informações contextuais, seria mais difícil tomar decisões acertadas.

Outra característica que pode ser observada nas aplicações de redes veiculares é a sua demanda por contextos lógicos. Contextos lógicos são informações contextuais que não podem ser coletadas diretamente por sensores físicos, e portanto devem ser inferidas com o uso de alguma técnica computacional [Ye et al. 2012]. Como exemplos de contextos lógicos no escopo das redes veiculares, pode-se citar a atividade desempenhada pelos usuários, as intenções do motorista, a ocorrência de um acidente e a identificação de uma vaga de estacionamento livre. Com o uso desse tipo de contexto em conjunto com outras informações contextuais, as decisões tomadas tendem a ser mais precisas.

Além de serem cientes de contexto, as aplicações para redes veiculares também podem ser classificadas como coletivas. Esse tipo de rede é composta por vários veículos automotores disputando recursos comuns como largura de banda, espaço em ruas, conteúdos online, dentre outros. Com isso, conflitos de interesse podem ocorrer quando mais de um veículo está interessado no mesmo recurso que é restrito e não pode ser alocado a todos os interessados. Por exemplo, quando dois ou mais veículos possuem interesse em uma vaga de estacionamento em determinada região e obviamente uma única vaga não pode ser alocada a mais de um veículo ao mesmo tempo, é importante decidir de maneira justa qual veículo será atendido.

O objetivo desse trabalho é definir e avaliar um *middleware* inteligente, chamado ConProVA (do inglês **C**ontext **P**rovisioning for **V**ehicular **A**pplications), para o provisionamento de contextos para aplicações de redes veiculares. Por serem cientes de contexto, essas aplicações precisam de uma funcionalidade para prover os contextos a elas. Esse provedor deve ser inteligente no sentido de ser capaz de inferir contextos lógicos e resolver conflitos de interesse quando necessário, já que essas são duas demandas das redes veiculares, como descrito anteriormente. O ConProVA foi projetado para atender a essas demandas, sendo também capaz de divulgar, por meio do paradigma de comunicação *Publish/Subscribe*, eventos ocorridos de acordo com as solicitações dos interessados. Além

disso, o *middleware* é flexível para ser adaptada a diferentes aplicações e demandas e se preocupa com o volume de dados transmitidos para não sobrecarregar a rede.

Um *middleware* pode ser considerado uma boa solução para provisionamento de contexto, pois permitirá que os desenvolvedores das aplicações foquem somente na aplicação sem se preocuparem com a coleta, processamento e armazenamento dos elementos de contexto. Além disso, um *middleware* irá permitir a comunicação entre tecnologias heterogêneas de diferentes fabricantes, já que atua em uma camada entre a aplicação e a plataforma de hardware, permitindo uma maior integração dos veículos e conseqüentemente uma melhor qualidade dos serviços.

Este artigo é composto por cinco seções e está organizado como descrito abaixo. A seção 2 apresenta os principais trabalhos relacionados encontrados na literatura. A seção 3 descreve os requisitos, arquitetura e detalhes do *middleware* proposto. A seção 4 contém todos os detalhes da avaliação realizada para verificar a viabilidade da proposta e os resultados obtidos. Finalmente, a seção 5 apresenta os comentários finais e alguns potenciais trabalhos futuros.

## 2. Trabalhos Relacionados

Muitos pesquisadores já propuseram *middlewares* para redes veiculares. No entanto, conforme descrito abaixo, as propostas não contemplam aspectos importantes que são cobertos pelo ConProVA.

[Nour et al. 2011] propuseram um *middleware* responsável por coletar, armazenar e disponibilizar contextos para as aplicações em redes veiculares. Esse *middleware* é organizado em camadas com responsabilidades específicas como coletar, filtrar, analisar e armazenar contextos. Além da organização em camadas, não foram apresentados detalhes e nenhuma avaliação foi realizada.

O sistema CarTel, proposto por [Bychkovsky et al. 2006], é composto por um computador embutido com sensores responsáveis pela coleta de dados dos carros como posição, disponibilidade de conexão WiFi, desempenho do motor, dentre outros. Os dados coletados são enviados a um servidor por meio de conexões sem fio intermitentes. Além da proposta, não foram apresentados mais detalhes e apenas uma versão simplificada da solução foi avaliada em uma demonstração.

[Riva 2006] propôs Contory, um *middleware* para prover contextos para *smartphones*. Apesar de não ser específico para redes veiculares, essa proposta apresenta características relacionadas ao trabalho que valem a pena descrevê-la. No Contory, são considerados três opções para coleta de dados: internos (locais em relação ao dispositivo), externos (coletados de um servidor) e distribuídos (coletados de elementos de rede conectados via Bluetooth ou WiFi). Além disso, foi proposta uma definição de campos que representam um contexto, como tipo, valor, data/hora da coleta, tempo de vida, origem e um campo opcional. Também foi proposta uma linguagem de consulta similar ao SQL que permite buscar e filtrar contextos de acordo com critérios escolhidos pelo interessado.

[Leontiadis et al. 2009] argumentam em seu trabalho que o paradigma de comunicação *Publish/Subscribe* (Pub/Sub) [Eugster et al. 2003] é o mais adequado para redes veiculares uma vez que cada veículo está interessado em eventos específicos e podem se inscrever para receber notificações sobre esses eventos. Além disso, dada a ca-

racterística assíncrona desse paradigma, um veículo pode se inscrever para algum evento mesmo se o fornecedor do mesmo não estiver conectado simultaneamente, o que tende a ser comum nas redes veiculares já que a topologia é dinâmica devido a mobilidade dos veículos. Dadas essas justificativas, foi proposto um *middleware* que adota o paradigma Pub/Sub e que considera a localização e o horário das publicações e notificações. Foram propostas as primitivas de comunicação sobre como se inscrever e como notificar a ocorrência de eventos. Além disso, foi proposta uma solução flexível para o casamento entre uma notificação e uma inscrição utilizando árvore de mapeamento. Porém, esse *middleware* assume que todos os contextos estão disponíveis para uso e possui foco somente na parte de comunicação.

Além das propostas de *middlewares* descritas, é importante apresentar trabalhos relacionados à inferência de contextos lógicos e resolução de conflitos, que são dois tópicos importantes para a definição do ConProVA. [Ye et al. 2012] fizeram um levantamento detalhado de várias técnicas e aplicações para inferência de contextos lógicos. Dentre essas técnicas está a *Naive Bayes*, que é adotada neste trabalho. Porém, nenhuma das aplicações de inferência descritas pode ser considerada flexível para diferentes demandas e aplicações, como é proposto no ConProVA. Em relação à identificação e resolução de conflitos, o trabalho desenvolvido por [Silva et al. 2010] propõe uma arquitetura que adota vários algoritmos em conjunto para identificar e resolver conflitos de acordo com a disponibilidade de recursos em aplicações ubíquas. Além dessa proposta para aplicações ubíquas, não foram encontrados outros trabalhos com o objetivo de resolver conflitos em redes veiculares.

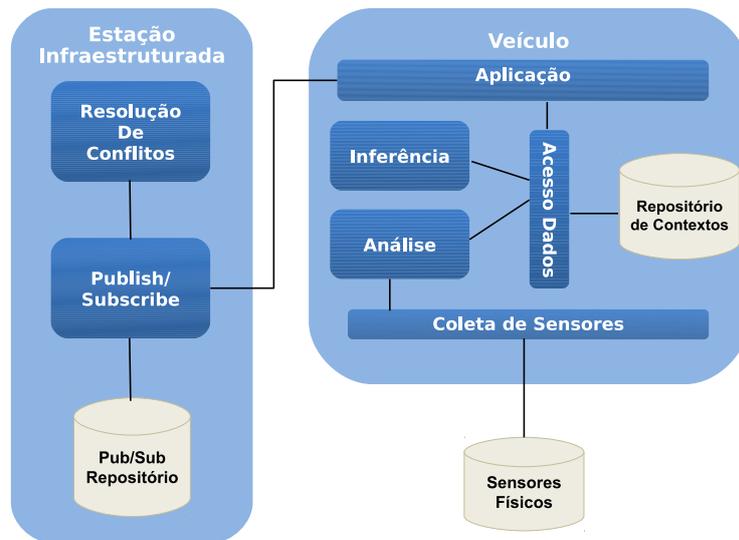
Todas as propostas descritas possuem suas contribuições para o estado da arte da área. Porém, elas não tratam questões importantes para as redes veiculares. Nenhuma solução trata da demanda das aplicações por contextos lógicos. Além disso, não é considerada a natureza coletiva das redes veiculares, em que conflitos de interesse podem ocorrer pela demanda por recursos restritos como vagas de estacionamento e largura de banda. Por último, também não é tratado o problema da grande quantidade de informação redundante que pode ser gerada em casos de tráfego intenso. Todos esses problemas são considerados e tratados pelo ConProVA, como será descrito na próxima seção.

### 3. ConProVA

Esta seção descreve os detalhes do *middleware* proposto, chamado ConProVA (**Context Provisioning for Vehicular Applications**). Dadas as características das redes veiculares, os principais requisitos do ConProVA são modularidade, inferência de contextos lógicos, tratamento de conflitos de interesse e compartilhamento de informação entre veículos. Além de atender a esses requisitos, o *middleware* proposto é flexível para ser adaptado de acordo com diferentes aplicações e demandas.

#### 3.1. Arquitetura do ConProVA

ConProVA foi organizado em componentes com características e interfaces específicas, como mostra a figura 1. É assumida uma arquitetura de comunicação cliente/servidor com múltiplos servidores em que os clientes são os veículos e os servidores são as estações fixas infraestruturadas localizadas nas ruas e estradas. Essas estações possuem maior capacidade computacional, são estáticas e conectadas entre si. Os veículos são carros ou qualquer outro automóvel com capacidade de comunicação sem fio, um sistema de



**Figura 1. Arquitetura de componentes do ConProVA**

posicionamento global e uma unidade de processamento. Os veículos são capazes de trocar mensagens por meio de comunicação sem fio com as estações fixas e entre si.

A vantagem em se utilizar uma arquitetura de comunicação cliente/servidor da maneira proposta é o fácil compartilhamento de informações entre os veículos pelos servidores com o mínimo de troca de mensagens, economizando assim largura de banda, que é um recurso escasso nas redes veiculares. De outra forma, seria preciso uma grande quantidade de mensagens roteadas entre múltiplos veículos. Por outro lado, essa arquitetura requer uma infraestrutura física que tem um custo de implantação. No entanto, considerando as grandes e médias cidades já cobertas pela rede celular, o custo dessa infraestrutura poderá ser reduzido mesmo considerando que as redes veiculares utilizam tecnologia de comunicação diferente.

Os componentes que fazem parte do ConProVA são:

**Coleta de Sensores:** responsável por implementar as interfaces entre o *middleware* e os dispositivos de sensores físicos como GPS, sensor de velocidade, uma rede de sensores sem fio, dentre outros;

**Análise:** responsável por processar os dados coletados pelos sensores para organizar os contextos ou agregar os dados relacionados. Pode utilizar os contextos armazenados no repositório de contextos para agregá-los com os novos dados e assim manter um histórico de eventos;

**Acesso aos Dados:** responsável por controlar o acesso ao repositório de contextos. Também é responsável por representar os dados coletados em forma de contextos de acordo com a especificação definida por [Riva 2006] e adotada pelo ConProVA por se tratar de uma representação completa e validada, composta por itens que descrevem informações espaciais, temporais e ambientais dos contextos;

**Inferência:** responsável por inferir contextos lógicos com base em dados contextuais existentes. A técnica selecionada para ser adotada por esse módulo é *Naive Bayes*, que é

baseada na regra de Bayes:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \quad (1)$$

onde  $X$  é a evidência e  $H$  é a hipótese. No escopo do ConProVA, a evidência é a informação contextual conhecida e a hipótese é o contexto lógico a ser inferido. Em outras palavras, é calculada a probabilidade de um contexto lógico ser válido dado o conhecimento de um conjunto de dados contextuais obtido por outros meios.

A técnica de *Naive Bayes* foi escolhida por ser um modelo de classificação simples e por assumir a independência condicional entre as evidências. Dessa forma, o cálculo é reduzido a:

$$P(H|X) = P(H) \times \prod_{1 \leq i \leq n} P(x_i|H) \quad (2)$$

onde  $x_i$  é o valor de um dos atributos da evidência (isto é, dados contextuais conhecidos no ConProVA) e  $n$  é a quantidade de evidências existentes. Com isso, é necessário saber somente a probabilidade das evidências quando é conhecida a hipótese ( $P(x_i|H)$ ). Apesar de sua simplicidade, *Naive Bayes* tem alcançado bons resultados quando comparada a outras técnicas de classificação em aprendizado de máquinas, de acordo com [Han and Kamber 2006].

Para cada contexto lógico a ser inferido, é necessário conhecer os valores probabilísticos dos atributos da evidência usados pelo *Naive Bayes*. Além disso, esses valores diferem de contexto para contexto e também de aplicação para aplicação. Por exemplo, a probabilidade de ocorrência de acidentes depende das características específicas de cada cidade. Para se ter uma solução flexível do módulo de inferência, um arquivo de configuração em formato XML foi definido para conter as informações de entrada do *Naive Bayes* para cada contexto lógico a ser inferido. Este arquivo deve ser editado e configurado de acordo com as necessidades específicas de contextos lógicos pelo projetista da aplicação.

**Resolução de Conflito:** responsável por identificar e resolver conflitos de interesse quando necessário. Vários recursos disponíveis para redes veiculares são restritos em quantidade. Com isso, quando mais de um veículo tem interesse em um mesmo recurso que não possa ser alocado a todos os interessados, um conflito ocorre e deve ser solucionado. O módulo de resolução de conflitos é responsável por tratar esse tipo de problema.

Similarmente à inferência de contextos lógicos, as características dos conflitos variam de acordo com as aplicações. Portanto, esse módulo também deve ser flexível o suficiente para ser adotado por diferentes aplicações com o mínimo de esforço possível. Para atender a essa demanda, o ConProVA utiliza uma função de utilidade configurável. Essa função é definida como:

$$U = \sum_{1 \leq i \leq n} c_i \times w_i \quad (3)$$

onde  $c_i \in C$ ,  $w_i \in W$ ,  $C$  é o vetor de contextos usados para o cálculo da utilidade e  $W$  é o vetor de pesos para cada contexto em  $C$ , e  $|W| = |C| = n$ , que é a quantidade de contextos utilizados para o cálculo da utilidade. Os vetores  $C$  e  $W$  são definidos pelo projetista de acordo com as características das aplicações e dos conflitos em um arquivo de configuração em formato XML.

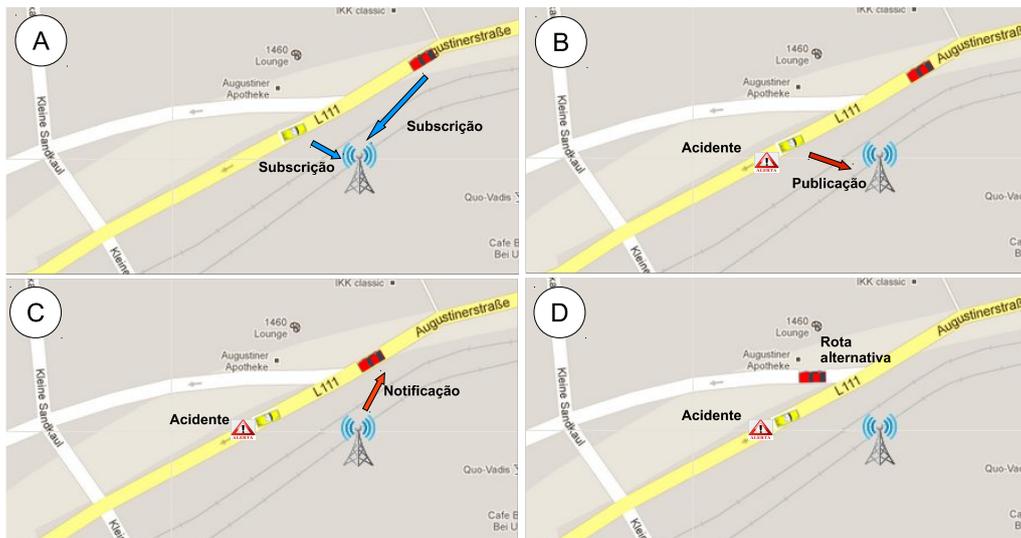
O valor da utilidade é calculada para todos os veículos em conflito e indica a importância dada a cada um deles para o recurso disputado. Com as utilidades dos veículos calculadas, o algoritmo soluciona o conflito selecionando os  $M$  veículos interessados com maior utilidade, onde  $M$  é a quantidade disponível do recurso que causou o conflito.

**Publish/Subscribe:** responsável pelo compartilhamento de informações entre os veículos. Um dos objetivos do *middleware* ConProVA é disponibilizar contextos e eventos para outros veículos, contribuindo assim para as tomadas de decisões. Para isso, o paradigma de comunicação *Publish/Subscribe* baseado em conteúdo foi utilizado por se tratar de uma solução interessante para redes veiculares. Algumas das suas principais características, como representação das mensagens, função de casamento entre eventos e interesses e a disponibilização dos eventos aos interessados, são baseadas no trabalho desenvolvido por [Leontiadis et al. 2009], que já definiu e avaliou um esquema de comunicação Pub/Sub para redes veiculares. É assumido que todos os veículos possuem um sistema de navegação com GPS e mapa detalhado da cidade em que se encontram. Além disso, as estações fixas possuem o mapa detalhado e são capazes de calcular rotas (incluindo alternativas) para atender aos veículos. Abaixo é descrito como cada funcionalidade desse módulo foi definida e a figura 2 ilustra um exemplo simplificado do funcionamento desse módulo.

**Subscrição:** quando um veículo tem interesse em alguma informação, ele cria uma mensagem de subscrição e a envia para as estações fixas da sua vizinhança naquele momento. Essa mensagem contém campos que permitem a identificação da subscrição, os eventos de interesse, a localização do veículo, seu destino final e a sua identificação, dentre outros. Para manter a subscrição sempre atualizada, os veículos periodicamente enviam seus interesses às estações próximas. O intervalo entre um envio e outro é dinâmico e inversamente proporcional à velocidade do veículo, uma vez que quanto mais rápido o veículo estiver, mais rápido seu estado e localização irão mudar. Por outro lado, um veículo mais lento não terá alterações significativas no seu estado, incluindo localização, em um curto período e portanto o intervalo pode ser maior. Essa dinamicidade no intervalo de reenvio das subscrições reduz a quantidade de mensagens redundantes enviadas, principalmente em situações de tráfego intenso, diminuindo assim o uso de largura de banda. A periodicidade do envio é dada pela fórmula  $intervalo = R/v$  onde  $v$  é a velocidade do veículo e  $R$  é o raio de cobertura das estações fixas. Caso  $intervalo < min$  ou  $intervalo > max$ , o valor do intervalo é atribuído ao limite inferior ( $min$ ) ou superior ( $max$ ), respectivamente. Os valores de  $R$ ,  $min$  e  $max$  são definidos especificamente para cada aplicação.

**Recepção de Subscrição:** quando uma estação fixa recebe uma mensagem de subscrição de um veículo, primeiramente ela verifica no seu repositório de Pub/Sub se a mensagem já foi recebida anteriormente e pode ser descartada. Em caso negativo, é verificado também o prazo de expiração da mensagem para que sejam descartadas mensagens já expiradas. Caso a mensagem não seja descartada por nenhum dos dois motivos descritos, ela é armazenada no repositório e compartilhada com as outras estações fixas, e será utilizada posteriormente quando algum evento ocorrer.

**Publicação:** quando algum evento ocorre, o veículo prepara uma mensagem de publicação contendo seu identificador, o contexto que originou o evento, a sua localização, o tempo de expiração do evento, e envia essa mensagem às estações de sua vizinhança. Quando o evento não é mais válido, o veículo então envia uma mensagem cancelando a



**Figura 2.** (A) os veículos enviam mensagem de subscrição à estação fixa para receberem notificação sobre o trajeto. (B) Em seguida, um acidente ocorre e é inferido pelo veículo amarelo, que envia uma publicação à estação. (C) A estação verifica quais veículos possuem interesse nessa publicação (no caso, o veículo vermelho), e então envia uma notificação ao mesmo. (D) Finalmente, ao receber a notificação, o veículo vermelho segue por uma rota alternativa.

publicação enviada anteriormente.

**Recepção de Publicação:** quando uma estação fixa recebe uma mensagem de publicação, as subscrições existentes no repositório são comparadas para verificar se há casamento de alguma subscrição com a publicação recebida. Se o evento representado pela publicação recebida possui um número restrito de recursos (por exemplo, o evento pode ser a disponibilidade de uma vaga de estacionamento ou a indicação de uma rota alternativa com restrição de capacidade) e o número de subscrições interessadas no evento é maior que o número de recursos disponíveis, então o módulo de resolução de conflito é invocado. Após ser executado, o módulo de resolução de conflitos irá retornar uma lista de subscrições que deverão ser atendidas. Para finalizar, a estação envia uma mensagem de notificação a todos os veículos atendidos contendo os detalhes do evento.

**Recepção de Notificação:** quando um veículo recebe uma notificação, ele verifica se o seu interesse no evento é ainda válido ou não. Se for, ele responde à estação notificadora confirmando a sua aceitação da notificação.

#### 4. Avaliação

O objetivo da avaliação realizada é mostrar como a adoção do ConProVA pode aumentar a satisfação dos usuários das redes veiculares. Para isso, uma aplicação de inferência de acidentes e indicação de congestionamentos causados pelos mesmos foi implementada em um ambiente de simulação. É importante informar que não é objetivo deste trabalho propor uma solução para esse problema específico, mas sim mostrar que o ConProVA pode ser utilizado para resolvê-lo. No entanto, para o cenário avaliado neste trabalho, é possível demonstrar que o *middleware* proposto pode ser configurado de acordo e obter bons resultados.

ConProVA foi simulado por meio da ferramenta de simulação de rede OM-NET++ [Varga 2001] e o gerador de tráfego SUMO [Behrisch et al. 2011]. As simulações foram realizadas considerando um modelo de tráfego real da cidade de Colônia, na Alemanha [Uppoor and Fiore 2011]. Esse modelo de tráfego foi escolhido por ser validado e estar disponível publicamente. Como as pesquisas relacionadas a redes veiculares são incipientes, existem poucos modelos disponíveis, e muitos deles não são validados.

#### 4.1. Definições do ConProVA

A seguir é descrito como cada um dos principais módulos do ConProVA foram configurados nos cenários de simulação.

**Aplicação:** Uma aplicação de identificação e indicação de congestionamento causado por acidentes foi implementada. Os veículos que adotam ConProVA são capazes de inferir a ocorrência de acidentes, o que é considerado um contexto lógico já que não existe sensor capaz de medir essa informação, e publicar esse evento para que outros veículos evitem o local do acidente, seguindo rotas alternativas até o destino final. Foi considerada a existência de veículos comuns e outros de emergência, como ambulâncias e viaturas.

**Módulo de Inferência:** Este módulo é responsável por inferir a ocorrência de acidentes. Para isso, o arquivo de configuração do módulo foi criado contendo as probabilidades de entrada para o modelo de *Naive Bayes*. Esses valores probabilísticos de entrada foram calculados com base em informações da prefeitura da cidade de Colônia [Hall 2012]. Segundo essas informações, durante o ano de 2010 ocorreram 43345 acidentes nessa cidade. Além disso, de acordo com o mapa da cidade utilizado nas simulações, Colônia possui 823 semáforos e 69205 vias (ruas, avenidas, dentre outros). Com base nesses números, nos detalhes da cidade e informações intuitivas, como a maior probabilidade de ocorrência de acidentes em dias de semana e em horários de maior movimento, foi possível estimar as probabilidades utilizadas como entrada para o *Naive Bayes*.

O modelo utilizado neste trabalho considera que um veículo pode estar se movendo em baixa velocidade principalmente por duas razões: ocorrência de acidente ou situação de semáforo. Então, dada a evidência da velocidade baixa do veículo, é calculada a probabilidade de o mesmo estar em situação de acidente ou de semáforo. A maior probabilidade é a mais provável e então considerada válida como contexto lógico. Em outras palavras, na fórmula 2,  $H = \{A, S\}$ , onde  $A$  e  $S$  significam *acidente* e *semáforo* respectivamente, e  $X = VB_t$  onde  $VB_t$  significa *Velocidade Baixa em um intervalo de tempo  $t$* . Ou seja, a ocorrência de acidente ou situação de semáforo são as hipóteses (contextos lógicos a serem inferidos) e o período de tempo em velocidade baixa é a evidência (dados contextuais conhecidos).

Para se ter uma maior precisão da inferência, a variável de evidência  $VB_t$  é composta por três atributos,  $VB_{t_1}$ ,  $VB_{t_2}$  e  $VB_{t_3}$ , que definem a quantidade de tempo em que o veículo está se movendo em baixa velocidade e  $t_1 < t_2 < t_3$ . Com isso, é possível ter mais informações sobre a situação do veículo e assim aumentar a chance de uma estimativa correta. Nesse caso, a evidência  $VB_{t_i}$  com  $i \in \{1, 2, 3\}$  será válida se o veículo estiver em velocidade baixa por um tempo  $t$  em que  $t_{i-1} < t < t_i$  ( $t_0 = 0$  por definição). Um veículo é considerado em velocidade baixa se está 10% abaixo da velocidade média da via em que se encontra.

Substituindo as hipóteses e evidências na fórmula 2, temos então:

$$P(H = h|VB_t) = P(H = h) \times \prod_{1 \leq i \leq 3} P(VB_{t_i}|h) \quad (4)$$

onde  $h \in \{A, S\}$  é a hipótese, que pode ser *acidente* ou *semáforo*. O valor máximo entre  $P(H = A|VB_t)$  e  $P(H = S|VB_t)$  é considerado o mais provável e é inferido como contexto lógico.

**Módulo de Resolução de Conflitos:** Como descrito anteriormente, a aplicação implementada identifica alguma situação de congestionamento causado por acidentes e publica esse evento para que outros veículos evitem o local do acidente seguindo rotas alternativas. No entanto, algumas vias das rotas alternativas podem ter uma capacidade máxima e, caso muitos veículos sigam por essa via, a mesma pode se sobrecarregar, causando congestionamento. Nesse caso, o módulo de resolução de conflito deve decidir quais veículos deverão seguir quais rotas, de forma a não sobrecarregá-las.

Para ajudar aos veículos de emergência (ambulâncias e viaturas policiais) alcançarem seus destinos o quanto antes, a informação de prioridade é utilizada como contexto na função de utilidade do módulo de resolução de conflitos. Além disso, a distância entre os veículos e as rotas alternativas também é usada. Nesse caso, veículos com maior prioridade e mais próximos das melhores rotas serão alocados para elas, e a quantidade de veículos alocada será tal que não sobrecarregará a rota, evitando que as rotas alternativas se congestionem também.

Então, na função de utilidade da fórmula 3, temos que  $C = \{prioridade, distancia\}$ ,  $W = \{w_p, w_d\}$  e o cálculo é dado por:

$$U = prioridade \times w_p + distancia \times w_d \quad (5)$$

onde  $w_p$  é o peso dado ao valor da prioridade e  $w_d$  é o peso dado à distância do veículo para a rota em questão. Nos cenários simulados,  $w_p \gg w_d$  para dar preferência aos veículos de emergência, que serão atendidos a não ser quando estiverem muito distantes das melhores rotas.

**Módulo Publish/Subscribe:** Todos os veículos periodicamente enviam uma mensagem de subscrição para receberem notificações de eventos relacionados à situação do tráfego na sua rota. O intervalo entre o envio das subscrições é dinâmico, calculado de acordo com a definição na seção 3, e varia entre  $min = 10s$  e  $max = 50s$ , tendo o raio de cobertura das estações infraestruturadas atribuído para  $R = 400m$ .

As estações infraestruturadas armazenam as subscrições recebidas em seus respectivos repositórios. Quando um evento (isto é, um acidente) é inferido pelo módulo de inferência, o veículo responsável envia uma publicação descrevendo o evento às estações. Quando uma publicação é recebida por uma estação infraestruturada, ela busca em seu repositório por subscrições que tenham interesse nesse evento. Em caso de encontrar, os veículos que enviaram tais subscrições são notificados por meio de mensagens de notificação pela estação.

Antes de enviar a notificação, a estação invoca o módulo de resolução de conflitos que seleciona os veículos de forma a satisfazer a função de utilidade e não sobrecarregar as rotas alternativas. Então, além de informar aos veículos sobre o evento, a estação também indica qual rota alternativa os mesmos devem seguir.

Tabela 1. Configuração da Simulação

Parâmetro	Valor
Percentual de veículos de emergência	10%
Quantidade de acidentes	Poisson com $\lambda = \lambda_A \in \{15, 120\}$
Duração dos acidentes	Poisson com $\lambda = \lambda_D \in \{10min, 30min\}$
Localização das estações fixas	Em grade, cobrindo toda a região
Protocolos de comunicação	IEEE 1609.4/802.11p [Eckhoff and Sommer 2012]
Cenários	<b>ConProVA:</b> versão completa do ConProVA <b>Comum:</b> Nenhuma das funcionalidades do ConProVA

## 4.2. Configurações da Simulação

Para avaliar a adoção do *middleware* proposto, foram realizadas simulações com o modelo real de tráfego da cidade de Colônia, na Alemanha [Uppoor and Fiore 2011]. Mais de 2000 veículos partem de suas posições iniciais entre as 6:00 e 6:30 da manhã. A simulação termina quando todos os veículos chegam ao seu destino final, o que representa aproximadamente 1h30min, dependendo da quantidade e da gravidade dos acidentes ocorridos. A quantidade e a gravidade (medida em tempo de duração do acidente, ou em outras palavras, o tempo gasto para que o acidente seja removido e as vias envolvidas sejam liberadas) dos acidentes segue uma distribuição de *Poisson* com o parâmetro  $\lambda$  selecionado para cobrir situações variadas, com muitos ou poucos acidentes, e com a duração dos mesmos variando para simular diferentes gravidades. O valor de  $\lambda$  para a quantidade de acidentes foi escolhido com base em estatísticas da prefeitura da cidade de Colônia. A tabela 1 apresenta as configurações utilizadas.

## 4.3. Resultados

Cada cenário foi simulado 33 vezes e os gráficos apresentam as médias e os respectivos desvios padrão.

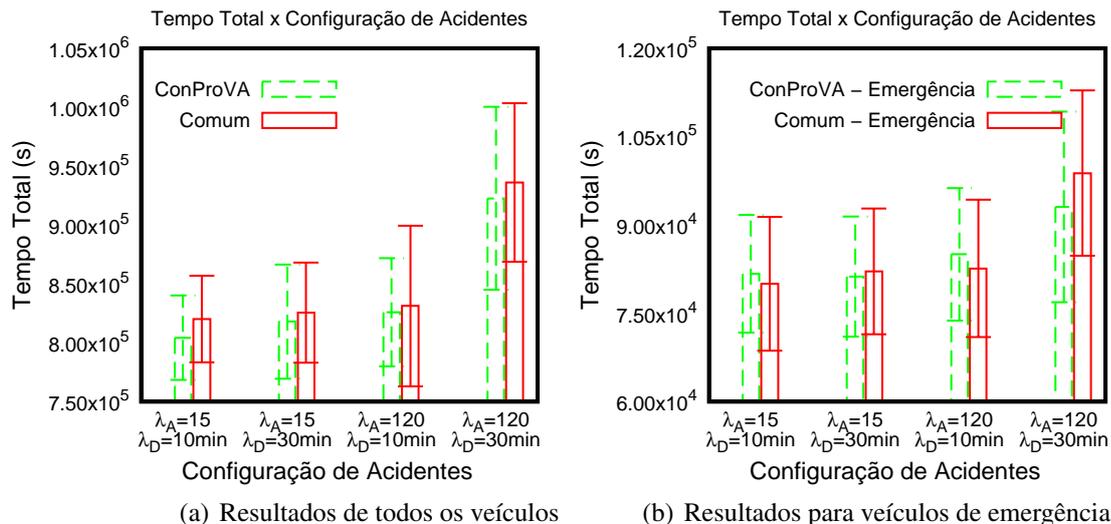


Figura 3. Tempo total de trajeto.  $\lambda_A$  = parâmetro da distribuição de Poisson para o número de acidentes e  $\lambda_D$  = parâmetro da distribuição de Poisson para a duração dos acidentes

**Tempo total de trajeto:** O tempo total de trajeto mede quanto tempo todos os veículos

precisam para alcançar os seus respectivos destinos finais. Em outras palavras, é a soma do tempo gasto por todos os veículos até que eles alcancem o seu destino final. Como o modelo de mobilidade utilizado é de um dia real, os mesmos veículos possuem a mesma origem e destino em todas as simulações. A figura 3(a) mostra que a adoção do ConProVA ajudou a reduzir o tempo médio total em todos os cenários. Isso ocorre pois, devido à inferência e notificação de acidentes, alguns veículos trocaram seu trajeto para uma rota alternativa, evitando assim os congestionamentos causados pelos acidentes.

Considerando somente os resultados dos veículos de emergência apresentados na figura 3(b), ConProVA contribuiu para a redução do tempo total quando a duração dos acidentes é maior ( $\lambda_D = 30\text{min}$ ). Quando os acidentes são menos severos e o congestionamento causado por eles é rapidamente solucionado, às vezes é mais adequado simplesmente esperar na rota congestionada ao invés de escolher uma rota alternativa que pode ser mais demorada em situações normais de tráfego.

**Emissão de  $CO_2$  e distância total:** A emissão de  $CO_2$  mede a quantidade de gás carbônico originado dos veículos que foi depositada no ar durante o trajeto dos mesmos. Essa informação é relevante nos dias atuais devido à grande preocupação com a qualidade do ar e a sustentabilidade. Para se ter uma ideia, na cidade de Belo Horizonte-MG, no ano de 2010 foram emitidos 3,75 milhões de toneladas de  $CO_2$  no ar, sendo que 71% desse montante é proveniente de veículos<sup>1</sup>. Esse número é 18% maior se comparado com o ano de 2007. Portanto, é de interesse da sociedade que se tenha soluções para diminuir a emissão de gases poluentes no ar.

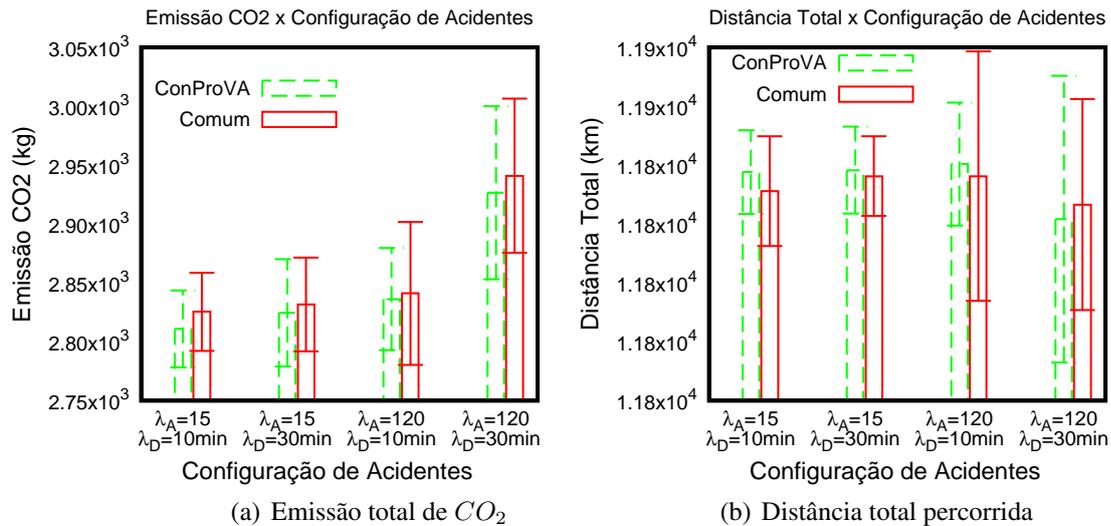
Nesse trabalho, quantidade de  $CO_2$  emitida é calculada com base em vários fatores como a velocidade e aceleração dos veículos, de acordo com a fórmula definida por [Cappiello et al. 2002]. Como apresentado na figura 4(a), os cenários em que os veículos adotam o ConProVA emitiram menos  $CO_2$  do que os cenários comuns. Isso ocorre pois, em geral, os veículos que adotam o ConProVA gastam menos tempo no tráfego por escolherem rotas alternativas quando acidentes ocorrem e são inferidos. Essa diminuição da emissão de  $CO_2$  ocorre mesmo com os veículos tendo que percorrer uma distância maior, como mostra a figura 4(b), devido às rotas alternativas serem, em geral, mais longas que as rotas tradicionais.

**Sobrecarga de comunicação:** Para avaliar o impacto da adoção do ConProVA em relação à largura de banda necessária, o número de mensagens trocadas foi contabilizado e é apresentado na tabela 2. Como pode ser observado, quanto maior a probabilidade de acidentes ( $\lambda_A = 120$ ), maior o número de mensagens trocadas pois são enviadas mais mensagens de subscrição e publicação. Porém, a taxa de mensagens trocadas por unidade de tempo (aproximadamente 1 mensagem a cada 23 segundos) não é significativa para sobrecarregar a rede, sendo que os benefícios vistos anteriormente superam essa troca extra de mensagens.

## 5. Conclusões e Trabalhos Futuros

Este trabalho apresentou ConProVA, um *middleware* para provisionamento de contextos para redes veiculares. ConProVA foi definido considerando as características específicas das redes veiculares e a sua arquitetura é flexível para ser facilmente adotado por diferen-

<sup>1</sup> fonte: 2<sup>o</sup> Inventário de Emissão de Gases de Efeito Estufa, realizado em Belo Horizonte em Dezembro de 2012



**Figura 4. Emissão de  $CO_2$  e distância percorrida.**  $\lambda_A$  = parâmetro da distribuição de Poisson para o número de acidentes e  $\lambda_D$  = parâmetro da distribuição de Poisson para a duração dos acidentes

**Tabela 2. Número médio de mensagens/segundo enviadas por veículo**

$\lambda_D$	$\lambda_A$	Subscrição	Publicação
		Msg/Segundo/veículo	Msg/Segundo/veículo
10min	15	$0,043 \pm 0,012$	$0,0005 \pm 0,003$
10min	120	$0,043 \pm 0,013$	$0,0008 \pm 0,003$
30min	15	$0,042 \pm 0,013$	$0,0006 \pm 0,004$
30min	120	$0,044 \pm 0,015$	$0,0014 \pm 0,005$

tes aplicações. Além de possibilitar o provisionamento de contextos físicos e de implementar o paradigma de comunicação *Publish/Subscribe* para publicação e notificação de eventos, o *middleware* proposto possui duas outras funcionalidades importantes em redes veiculares: inferência de contextos lógicos e resolução de conflitos de interesse. Essas duas funcionalidades são os principais diferenciais do ConProVA em relação a soluções encontradas na literatura.

Simulações feitas utilizando um cenário real de tráfego mostraram que a adoção do ConProVA possibilitou o aumento da satisfação dos usuários ao diminuir o tempo que os veículos gastam para chegar ao destino final e reduzir a emissão de gás carbônico quando acidentes ocorrem e conseqüentemente, causam congestionamento.

Como trabalhos futuros, pode-se listar o refinamento dos módulos de inferência e resolução de conflitos, a proposta de técnicas para medir a qualidade dos contextos (QoC) providos e a adoção do ConProVA junto a outras aplicações.

## Referências

- Behrisch, M., Bieker, L., Erdmann, J., and Krajzewicz, D. (2011). Sumo - simulation of urban mobility: An overview. In *SIMUL 2011, The Third International Conference on Advances in System Simulation*, pages 63–68, Barcelona, Spain.
- Bychkovsky, V., Chen, K., Goraczko, M., Hu, H., Hull, B., Miu, A., Shih, E., Zhang, Y., Balakrishnan, H., and Madden, S. (2006). The cartel mobile sensor computing

- system. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, SenSys '06, pages 383–384, New York, NY, USA. ACM.
- Campista, M. E. M., Moraes, I. M., Rubinstein, M. G., and Duarte, O. C. M. B. (2009). Redes Veiculares: Princípios, Aplicações e Desafios. In *Minicursos do XXVII Simpósio Brasileiro de Redes de Computadores.*, chapter 5, pages 199–254. Recife.
- Cappiello, A., Chabini, I., Nam, E., Lue, A., and Abou Zeid, M. (2002). A statistical model of vehicle emissions and fuel consumption. In *Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on*, pages 801 – 809.
- Eckhoff, D. and Sommer, C. (2012). A Multi-Channel IEEE 1609.4 and 802.11p EDCA Model for the Veins Framework. In *5th ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2012)*. ACM.
- Eugster, P. T., Felber, P. A., Guerraoui, R., and Kermarrec, A.-M. (2003). The many faces of publish/subscribe. *ACM Comput. Surv.*, 35(2):114–131.
- Hall, C. C. (2012). Cologne city hall. Available in <http://www.stadt-koeln.de/en/>.
- Han, J. and Kamber, M. (2006). *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Leontiadis, I., Costa, P., and Mascolo, C. (2009). A hybrid approach for content-based publish/subscribe in vehicular networks. *Pervasive and Mobile Computing*, 5(6):697 – 713. PerCom 2009.
- Nour, S., Negru, R., Xhafa, F., Pop, F., Dobre, C., and Cristea, V. (2011). Middleware for data sensing and processing in vanets. In *Emerging Intelligent Data and Web Technologies (EIDWT), 2011 International Conference on*, pages 42 –48.
- Riva, O. (2006). Contory: A middleware for the provisioning of context information on smart phones. In van Steen, M. and Henning, M., editors, *Middleware 2006*, volume 4290 of *Lecture Notes in Computer Science*, pages 219–239. Springer Berlin / Heidelberg.
- Silva, T. R. M. B., Ruiz, L. B., and Loureiro, A. A. F. (2010). Uma arquitetura para resolução de conflitos coletivos em sistemas ubíquos e cientes de contexto. In *XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 119–132.
- Uppoor, S. and Fiore, M. (2011). Large-scale urban vehicular mobility for networking research. *2011 IEEE Vehicular Networking Conference (VNC)*, pages 62–69.
- Varga, A. (2001). The OMNET++ discrete event simulation system. In *Proceedings of the European Simulation Multiconference*, pages 319–324, Prague, Czech Republic. SCS – European Publishing House.
- Ye, J., Dobson, S., and McKeever, S. (2012). Situation identification techniques in pervasive computing: A review. *Pervasive and Mobile Computing*, 8(1):36–66.