NDN-Waze: Um Sistema Distribuído para Gerenciamento de Tráfego Veicular via Redes de Dados Nomeados

Guilherme B. Araujo[1,2], **Leobino N. Sampaio**[1]

[1]Instituto de Computação – Universidade Federal da Bahia (UFBA)

[2] Centro de Ciências Exatas – Universidade Federal do Recôncavo da Bahia (UFRB)

guilherme.araujo@ufrb.edu.br, leobino@ufba.br

Abstract. *Several transport systems in vehicular networks (VANETs) rely on IP-based models for inter-vehicle communication. However, many approaches are not secure and appropriate for VANETs, using specific technologies, network architectures, or limited communication models. In addition, many applications are forced to deal with complex services that the network layer does not provide. This paper proposes NDN-Waze, a distributed lightweight system for VANET via Named Data Networking (NDN). NDN-Waze is a proof-of-concept design for a vehicular route guidance application. Simulation results show that NDN properties can override traditional IP-based forwarding proposals in VANET.*

Resumo. *Em redes veiculares (VANETs) há vários sistemas de transporte que se apoiam em diferentes modelos dependentes do IP para comunicação entre veículos. Entretanto, muitas abordagens não são seguras e adequadas para o contexto, utilizando-se de tecnologias específicas, arquiteturas de rede ou modelos de comunicação limitados. Além disso, muitas aplicações acabam tendo que lidar com questões complexas de serviços não ofertados pela rede. Este artigo propõe o NDN-Waze, um sistema leve totalmente distribuído para VANET via Named Data Networking (NDN). NDN-Waze é uma prova conceitual completa do design de uma aplicação de orientação de rota veicular. Os resultados de simulação ilustram que propriedades da NDN podem sobrepor propostas tradicionais de encaminhamento de dados baseados no IP em VANET.*

1. Introdução

Rede veicular (VANET) se caracteriza pela mobilidade dos nós, conectividade dinâmica e intermitente entre dispositivos na borda da rede, dificultando a entrega de dados vinculados à localização (endereço) e dependentes somente de comunicação infraestruturada. A arquitetura TCP/IP não foi projetada para lidar com tais características, seja pelo modelo, escassez de endereços IP ou problemas de roteamento. Nesse contexto, *Cooperative Intelligent Transport Systems and Services* (C-ITS) avançaram em direção a um sistema tolerante a certos atrasos e desconexões, para ser viável e escalável [Magdum et al. 2019].

Os modelos centrados no *host* de arquiteturas de rede para comunicação entre veículos (*Inter Vehicle Communication* – IVC) não mudaram significativamente para atender novas demandas e características de redes emergentes [Araujo et al. 2022]. Suporte a mobilidade dos nós, encaminhamento eficiente de dados e segurança fim a fim, sob constantes mudanças de topologia e desconexões frequentes, são desafios que aplicações VANET baseadas no IP [Hussein et al. 2022] têm que lidar para funcionar. Como modelos apoiados sobre IP não tratam de tais questões na camada de rede, uma complexidade

adicional é levada para a camada de aplicação ou subsistemas, resultando na criação de ecossistemas inteiros voltados ao gerenciamento de problemas inerentes do contexto veicular e, muitas vezes, se apoiando em soluções baseadas no melhor esforço.

A arquitetura de Redes de Dados Nomeados (*Named Data Networking* – NDN) baseia-se no paradigma de Redes Centradas na Informação, que prevê a desassociação entre o localizador e o identificador de conteúdo, presente nas redes IP. Essa característica impõe mudanças relevantes na forma de identificação do conteúdo, do roteamento/encaminhamento de dados e da segurança na rede [Sampaio et al. 2021]. Além disso a NDN adota um modelo de comunicação assíncrono orientado pelo receptor, baseado em nomes de dados/conteúdo com suporte nativo para a nomeação de pacotes e segurança em nível dos dados, por meio de criptografia de chave pública, com definição de regras e políticas para diferentes entidades na rede. Diante de tais benefícios de uma arquitetura emergente como a NDN, este trabalho busca elucidar a seguinte questão: existem diferenças entre as arquiteturas IP e NDN para o encaminhamento seguro de dados em VANETs, que possam levar a diferenças significativas nas funcionalidades, desempenho ou qualidade de serviços de uma aplicação distribuída na rede?

Este artigo apresenta uma nova perspectiva para construção de aplicações distribuídas, leves, escaláveis, seguras e eficientes para troca de dados/informações em VANET via NDN. A principal contribuição do trabalho é apresentar, por meio de um estudo de caso prático, uma prova conceitual completa do projeto de uma aplicação C-ITS de orientação de rota veicular para NDN, denominada "NDN-Waze". Um estudo experimental detalhado demonstra que diferentes propriedades da NDN podem sobrepor modelos tradicionais de comunicação baseados no IP para VANET. Na literatura atual há iniciativas e discussões de VANETs sobre a arquitetura NDN [Khelifi et al. 2020], porém visando analisar aspectos específicos do modelo, e.g., *in-network caching*, encaminhamento eficiente de dados para distribuição de conteúdo no contexto veicular etc.

O trabalho está organizado assim: a Seção 2 apresenta fundamentos e discute trabalhos relacionados; a Seção 3 detalha o sistema NDN-Waze proposto como prova conceitual para NDN; a Seção 4 apresenta um estudo experimental e, baseado nos resultados, fomenta discussões sobre VANET via NDN e IP; a Seção 5 conclui o trabalho.

2. Fundamentos e Trabalhos Relacionados

Existem diferentes tipos de aplicações veiculares com características diversas, tais como eficiência de tráfego, sensoriamento, entretenimento [Hussein et al. 2022]. Eficiência de tráfego e orientação de rota veicular são sistemas clássicos para VANETs e comumente são implementados na literatura por meio de soluções C-ITS. Esses sistemas baseiam-se em diferentes arquiteturas IVC e serviços disponíveis na rede para prover soluções otimizadas com requisitos únicos ou multiobjetivos (e.g., rotas rápidas, seguras e com menor consumo de recursos). A Figura 1 ilustra em alto nível a arquitetura proposta por várias aplicações C-ITS baseadas na pilha IP [de Souza et al. 2017].

A transmissão de dados de aplicações VANET via WAVE (*Wireless Access Vehicular Environments*) é implementada por meio de encaminhamento oportunístico de pacotes, salto a salto via *broadcast*, para encontros aleatórios, em um processo conhecido como *store-carry-forward* [Magdum et al. 2019]. A interconexão de diferentes dispositivos e tecnologias IoT (*Internet of Things*) por meio do padrão WAVE é complexa devido ao *design* do modelo, que não considera especificidades do ambiente IoT [Patel et al. 2019].

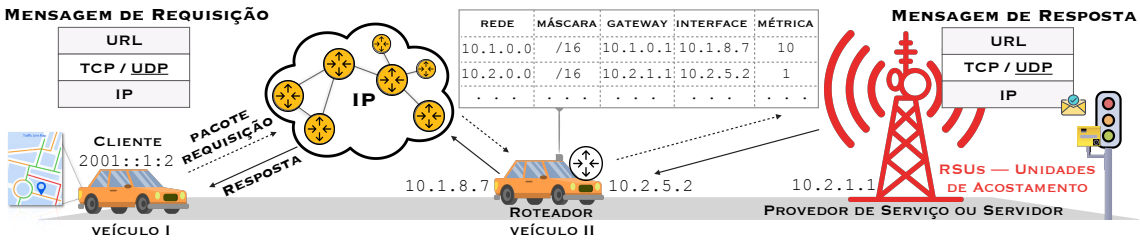


Figura 1. VANET via pilha IP: modelo de comunicação V2X para C-ITS. O nó cliente envia mensagens na rede, especificando o endereço IP do provedor. O veículo II atua como encaminhador, roteando pacotes na rede. Entretanto, oferecer suporte à mobilidade dos nós e segurança dos dados ainda é desafiador.

Alguns trabalhos na literatura [Magdum et al. 2019, Hussein et al. 2022] desenvolveram soluções de encaminhamento/roteamento para VANET via WAVE/IP, adaptando protocolos para entrega de pacotes baseados em endereços IP existentes, para melhorar a taxa de dados, diminuir a latência, aumentar a conectividade entre dispositivos na borda e oferecer mais serviços para a camada de aplicação. Entre as soluções de encaminhamento, o OLSR (*Optimized Link State Routing*) é bem conhecido como um protocolo de roteamento proativo, o qual utiliza informações de estado de *link* para construir de forma dinâmica a topologia e permitir a entrega de pacotes (sob conectividade intermitente) entre origem e destino, em redes *ad hoc* sem fio através do IP. Existem variações e otimizações no OLSR [Toutouh et al. 2012] para VANET com o objetivo de suportar atrasos mais longos na comunicação, devido à conectividade de baixa qualidade em cenários com baixa densidade de veículos. Contudo, apesar das otimizações para entrega de pacotes, fornecer comunicação segura, fim a fim, em VANET via WAVE/IP ainda é desafiador.

Existem aplicações VANET via IP que propõem soluções centralizadas, padrão cliente/servidor, assumindo que sempre haverá conectividade fim a fim por meio de arquiteturas infraestruturadas, como C-V2X (*Cellular Vehicle-to-Everything*), para prover segurança do meio ou acesso estável para infraestruturas de chave pública (PKI) [Hussein et al. 2022]. A fim de viabilizar o funcionamento sobre tecnologia celular, algumas propostas criam ecossistemas inteiros para tratar dos problemas inerentes do ambiente [Storck and Duarte-Figueiredo 2019]. Outras soluções pontuais aplicam *patches* isolados para superar as limitações do IP em VANETs e, assim, fazer funcionar sem convergir em direção a uma visão compartilhada, completa, que atenda aos requisitos atuais e futuros de aplicações distribuídas para Internet e computação de borda. As soluções baseadas em C-V2X ainda estão focadas na tecnologia em si, e não em atender aos requisitos ou serviços de garantia de qualidade e escalabilidade de aplicações distribuídas na borda.

A NDN é uma arquitetura *clean-slate* sem interdependência com o IP atual ou tecnologia específica. Devido às suas propriedades, características e serviços de rede, a NDN pode ser utilizada por aplicações distribuídas em VANETs [Khelifi et al. 2020]. Tais serviços e propriedades são decorrentes de componentes intrínsecos da arquitetura, como a *Pending Interest Table* (PIT) e a *Content Store* (CS), que são responsáveis, respectivamente, por manter uma lista de requisições pendentes, permitindo um plano de encaminhamento otimizado, ciente de estado (*stateful*) e *in-network caching*. Outro ponto positivo é que a NDN também oferece suporte nativo para *multicast*, *multihoming*, mobilidade dos nós na rede e nomeação de dados com uma semântica útil para a camada de rede e aplicação [Sampaio et al. 2021]. Em uma nuvem NDN, cada aplicação nomeia os dados que deseja buscar e a rede também usa os nomes para encaminhamento otimizado.

3. O sistema NDN-Waze

NDN-Waze é um sistema C-ITS distribuído que utiliza diferentes dispositivos e propriedades da arquitetura NDN para oferecer serviços de orientação de rota para motoristas ou veículos autônomos de forma segura na rede veicular. NDN-Waze adota a comunicação V2X (*vehicle-to-everything*) para coleta de dados sobre a infraestrutura e troca de informações sobre orientação de rota entre veículos e unidades de acostamento (*Road Side Units* – RSUs). Assim, os dispositivos espalhados na rede rodoviária atuam como produtores de dados sobre o contexto e fornecem diferentes informações para o sistema das RSUs, como, por exemplo, velocidade média da via e densidade de veículos.

As unidades de acostamento (RSUs) têm dois papéis: atuam como (i) consumidores de dados sobre as condições do tráfego veicular, obtidos por meio de diferentes nós espalhados na infraestrutura (e.g., sensores, radares de velocidade, câmeras de monitoramento) e interconectados localmente por intermédio de diferentes tecnologias, com o objetivo de construir e alimentar a base de dados distribuída na rede; e também como (ii) provedores (produtores) de serviços de internet para navegação e orientação de rota veicular. Os veículos, interessados em rotas rápidas entre origem e destino, atuam como consumidores e encaminhadores de conteúdo na rede. Basicamente, os veículos enviam interesse na rede para serviços de orientação e recomendação de rotas e recebem, por meio de pacotes de dados, as rotas que devem seguir até o destino de interesse.

Para prover serviços de orientação de rota, o sistema NDN-Waze utiliza, em unidades de acostamento, um grafo ponderado da rede veicular, $G_{RV} = (V(G_{RV}), A(G_{RV}))$, para modelagem dos dados e gerenciamento da infraestrutura rodoviária. O conjunto $V = \{v_1, v_2, \ldots, v_i\}$ representa os cruzamentos (vértices), enquanto o conjunto $A = \{a_{12}, a_{13}, \ldots, a_{ij}\}$ corresponde ao conjunto de vias (arestas). Dessa forma, a aresta a_{ij} é definida por um par de vértices subsequentes $(v_i, v_j) \in V(G_{RV})$. Além disso, o conjunto de pesos $D = \{d_{12}, d_{13}, ..., d_{ij}\}$ representa a distância, em metros, de cada aresta $a_{ij} \in A$. Outra camada, ou seja, conjunto de pesos $P = \{p_{12}, p_{13}, ..., p_{ij}\}$, corresponde ao tempo (atual) estimado de viagem para cada via ou aresta do grafo G_{RV}, sendo que essa informação é obtida indiretamente, por meio da distância e velocidade média. Seja $N = \{n_1, n_2, ..., n_i\}$ o conjunto de veículos na rede, uma rota R, seja qual for, $\forall\, n \in N \Rightarrow R = \{a_{ij}, ..., a_{mn}\} \mid R \subseteq A(G_{RV})$. Assim, resumidamente, uma rota veicular para o sistema representa um caminho no grafo que identifica as ruas pelas quais um veículo n deve percorrer, a partir da posição atual até o destino final.

De uma forma geral, o NDN-Waze agrega diferentes protocolos e características da NDN para prover um serviço distribuído completo de orientação de rota, considerando diferentes aspectos necessários para prover um serviço desaa natureza em escala metropolitana. A base de funcionamento da proposta é o esquema de nomeação hierárquico da NDN, enriquecido semanticamente, que também facilita adoção do modelo de confiança, para definição de nós confiáveis distribuídos na rede para produção de dados sobre tráfego veicular. O NDN-Waze ainda se beneficia da abordagem de compartilhamento oportunista, utilizando *caches* veiculares na camada de rede (NDN), para trocar informações de forma mais otimizada. A aplicação também inclui segurança para garantir autenticidade e integridade dos dados na rede, sincronização de bases de dados entre unidades de acostamento vizinhas e comunicação bidirecional entre diferentes dispositivos de monitoramento. As próximas subseções abordam essas propriedades e funções em detalhes.

3.1. Descoberta de Dispositivos e Vizinhança

Os dispositivos na rede veicular (e.g., veículos, unidades de acostamento, sensores IoT) enviam periodicamente mensagens *beacons* de interesse ou controle para alertar vizinhos na rede sobre sua presença. Cada nó pode manter uma tabela de vizinhança atualizada, com base em mensagens *beacons* recebidas recentemente. A remoção de vizinhança ocorre quando um nó deixa de receber novos *beacons* de um vizinho previamente conhecido. As unidades de acostamento e sensores locais também podem estimar a densidade de veículos na área ou velocidade média nas vias por meio dessas mensagens periódicas.

Os *beacons* na NDN são mensagens de medição, portanto não é necessário retornar nenhum pacote de dados por padrão. A periodicidade das mensagens pode ser ajustada, mas por padrão o NDN-Waze utiliza um intervalo de um segundo. O esquema de nomes proposto é `/localhop/beacon/hi-i-am/<node-id>/<road-id>/<pos-x>/<pos-y>/<pos-z>/<speed>`. O prefixo para identificação do pacote é composto por `/localhop/beacon/hi-i-am`. Como a restrição de escopo proposta é `/localhop`, o próprio NFD (NDN *Forwarding Daemon*) [Sampaio et al. 2021] é capaz de gerenciar e limitar a propagação desses pacotes em um único salto, ou seja, não haverá retransmissão de mensagens com este prefixo. Assim, um nó NDN não poderá encaminhar *beacons* em nome de outro na rede, o que evita o estabelecimento de adjacências incorretas. Além disso, o NDN-Waze não precisa lidar com problemas de inundação de mensagens *beacons* na rede (*broadcast storm problem*) em nível de aplicação, já que a arquitetura oferece esse serviço na camada de rede, por meio da limitação de escopo.

No NDN-Waze, cada nó na rede, ao transmitir *beacons*, anexa ao sufixo do nome do pacote seu identificador, posição atual e velocidade momentânea. Lembre-se que o identificador `<node-id>` é composto de duas partes: uma marcação do tipo de dispositivo (e.g., CAR, RSU, IoT-SENSOR) e um identificador único do nó na rede. Toda a estrutura de nomeação de mensagens *beacons* permite que diferentes dispositivos construam dinamicamente, em tempo real, um conhecimento local sobre a topologia da rede. Todos os nós na rede NDN com o aplicativo instalado são configurados com uma entrada na tabela FIB, com o prefixo `/localhop/beacon/hi-i-am`, para encaminhar pacotes de interesse para a aplicação NDN-Waze local processar.

3.2. Monitoramento, Disseminação e Sincronização de Bases de Dados

Dispositivos interconectados por RSUs (e.g., sensores IoT) podem monitorar a infraestrutura rodoviária e ajudar a produzir uma base de dados sobre as condições de mobilidade [Patel et al. 2019]. Contudo, o espaço de nomes para monitoramento do tráfego é padronizado pelo NDN-Waze para qualquer nó na rede (i.e., `/localhop/ndn-waze/RTDB/<action>/<node-id>/<road-id>/<version>`). Assim, o sistema nas RSUs pode manter uma fonte confiável de dados para atualização das informações de cada via.

As mensagens RTDB (*Road Traffic Data Base*) têm uma semântica específica para a aplicação NDN-Waze, são identificadas pelo prefixo `/localhop/ndn-waze/RTDB` e possuem uma limitação de escopo `/localhop`, que impede que um nó retransmita mensagens de monitoramento para fora da área de cobertura e gerenciamento de uma RSU. Além disso, duas ações nomeadas em `/<action>`, *BEEP* e *UPDATE*, dão suporte à sincronização de dados entre dispositivos na rede. O sufixo `/<node-id>/<road-id>/<version>` é usado pela aplicação para controle e gerenciamento das bases de

dados distribuídas, mantendo somente dados atuais (versão recente) nas RSUs.

A disseminação de dados de monitoramento sobre a infraestrutura veicular é o processo pelo qual vários dispositivos na rede, instalados na própria unidade de acostamento ou espalhados na área de cobertura de rede, colaboram com o objetivo de propagar medições e informações de mobilidade de uma área. As unidades de acostamento mantêm a gerência dos dados e RTDB local. A disseminação desses dados ocorre sob demanda, por meio de mensagens de alerta RTDB-BEEP, quando uma condição do tráfego oscila além de um limiar pré-configurado em um sensor ou nó de monitoramento na rede. Para obtenção das informações, em razão do modelo de comunicação *pull-based* da NDN [Khelifi et al. 2020], o envio de mensagens RTDB-UPDATE requer primeiro um alerta (*beep*) para novo conteúdo disponível, seguido por interesses enviados por consumidores (e.g., RSUs), para então o produtor do conteúdo (e.g., IoT-SENSOR) enviar o dado em si.

Essa abordagem proposta para construção e manutenção da base de dados distribuída do sistema NDN-Waze foi concebida para evitar a propagação de pacotes de monitoramento além da área de medição e controle das unidades de acostamento e, também, garantir uma granularidade fina dos dados. A finalidade é identificar os dados de forma única e permitir atualizações pontuais à medida que alterações relevantes acontecem. Essas características garantem a leveza da proposta, não havendo sobrecarga da rede com troca constante de grande volume de dados de monitoramento e controle.

A sincronização da RTDB entre as RSUs é a fase em que as unidades de acostamento utilizam mecanismos de comunicação assíncronos para construir uma visão coletiva, comum e atualizada das condições do tráfego e mobilidade veicular de toda a área monitorada. Por isso, há a necessidade de interconectar as RSUs por intermédio de alguma topologia (e.g., linear, estrela, grade) com tecnologias adequadas ao contexto.

A Figura 2 ilustra em alto nível o diagrama de sequência do processo de disseminação de mensagens e sincronização de dados de monitoramento entre RSUs adjacentes. Na etapa 0, as RSUs estão transmitindo *beacons* na rede veicular (conforme Seção 3.1). Na etapa 1, a RSU I inicia localmente o processamento do *beacon* recebido e atualiza sua tabela de vizinhança, marcando o instante em que o vizinho foi visto. Destaca-se que os nós na tabela de vizinhança com último momento visto acima da periodicidade da mensagem *beacon* são removidos da tabela nessa etapa, pois deixam de ser considerados vizinhos imediatos, ou seja alcançáveis com apenas um salto na rede. Na etapa 2, a RSU I, depois de por exemplo processar mensagens de dados de dispositivos de monitoramento, atualiza sua base de dados local (RTDB). Nesse momento, o módulo de sincronização (sinc.) da aplicação NDN-Waze, na própria RSU I, inicia o processo de sincronização da base de dados distribuída entre RSUs.

As etapas subsequentes destacadas na Figura 2 (passos 3 até 9) detalham o funcionamento do módulo de sincronização e como as RSUs constroem, em conjunto, de forma segura, colaborativa e leve, uma base de dados distribuída completa sobre as condições do tráfego veicular. A etapa 3 começa depois de uma atualização na RTDB local, na qual a RSU I envia um interesse RTDB-BEEP na rede para alertar unidades de acostamento vizinhas sobre uma atualização recente em sua base de dados local. É importante notar que, ao enviar um *beep* na rede, a RSU I especifica claramente no nome do pacote qual dado sofreu alteração e qual é a versão mais recente disponível. No exemplo, `/ufba/RSU/1`

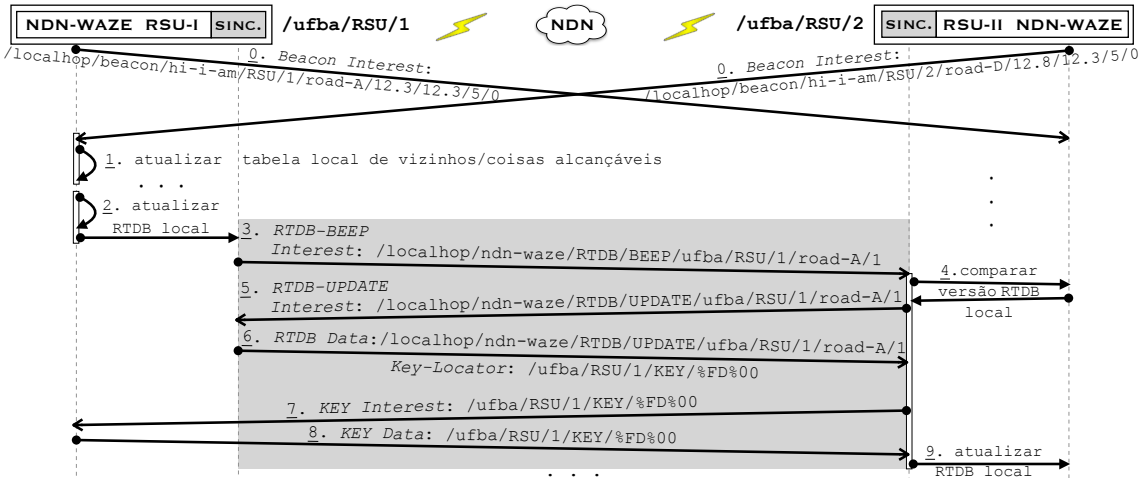


Figura 2. Diagrama de sequência do processo de disseminação e sincronização (sinc.) de dados sobre a infraestrutura rodoviária e mobilidade na rede veicular.

representa o nome completo do nó na rede (i.e., RSU I), o nome `road-A` identifica a infraestrutura rodoviária que sofreu atualização nos dados e, por fim, o marcador `1` mostra qual é o número da versão mais recente do dia para o dado especificado. Na sequência, na etapa 4, a RSU II, ao processar o interesse RTDB-BEEP, compara as versões dos dados armazenados localmente e extraídos do pacote recebido para identificar se já tem o conteúdo mais recente em sua base de dados. Se o número da versão no pacote RTDB-BEEP for maior, a RSU II iniciará o processo de sincronização e atualização de sua RTDB local. Neste cenário ilustrativo, considera-se que a RSU II tem em sua RTDB local o dado `/ufba/RSU/1/road-A/0`. Assim, como o dado local está desatualizado (versão #0), a RSU II enviará um pacote de interesse RTDB-UPDATE na rede com o objetivo de obter a versão mais recente (#1) e manter sua base de dados síncrona (etapa 5). A RSU I, então, responderá ao interesse RTDB-UPDATE com um pacote de dados assinado digitalmente, contendo informações atualizadas (e.g., velocidade média da via/infraestrutura, densidade de veículos na área, tamanho da via em metros, condições climáticas etc.) sobre a infraestrutura em questão (etapa 6). Ao receber o pacote de dados RTDB-UPDATE, a RSU II identifica a chave usada para assinar o pacote e, caso não a tenha em *cache*, envia um novo interesse de chave na rede (etapa 7). Na etapa 8, a RSU I envia sua chave pública por meio de um pacote de dados, sendo responsabilidade da RSU II realizar a validação necessária, de acordo com o modelo de confiança e as regras de validação (conforme detalhado na Seção 3.4). Finalmente, após validar o pacote de dados RTDB-UPDATE, a RSU II pode atualizar de forma segura sua base de dados RTDB local com informações mais recentes sobre uma determinada via (e.g., `/ufba/RSU/1/road-A/1`).

3.3. Orientação de Rota Veicular e Cálculo de Caminhos

O NDN-Waze inclui outro esquema de nomeação semanticamente enriquecido para o serviço de orientação de rota veicular. A Figura 3 ilustra uma rede VANET via NDN com o sistema funcionando, em que o veículo I, interessado em orientação de rota, encaminha interesses na rede até alcançar uma unidade de acostamento, responsável por prover o serviço. Contudo, *caches* oportunísticas no meio podem conter o conteúdo ou dado de interesse do veículo, abreviando assim a comunicação até uma unidade de acostamento.

A Figura 3 ilustra o esquema de nomeação hierárquico completo (prefixo + sufixo)

para o sistema de orientação de rota. O prefixo `/service/its/ndn-waze` caracteriza o serviço na rede. Porém, para o sistema funcionar, também são necessárias todas as informações presentes no sufixo. Assim, o sufixo define diferentes parâmetros relevantes para o contexto da aplicação, em que `<PointA>` representa o ponto atual (origem) do nó e `<PointD>` identifica o ponto de destino. Lembrando que, nesse contexto veicular, origem e destino são identificadores únicos (IDs) da infraestrutura rodoviária. Por fim, a janela temporal (i.e., `<time-window>`) serve para indexar requisições no tempo para a aplicação identificar os dados e momento. Como as condições do tráfego rodoviário mudam durante o dia, a indexação dos dados transitórios no tempo é necessária para evitar fornecer informações de rotas veiculares desatualizadas (obsoletas) armazenadas em *caches*. A janela atual é um número inteiro calculado pelos nós na rede por meio do *timestamp* atual por um intervalo temporal fixo, configurado a priori pela própria aplicação NDN-Waze durante a inicialização do sistema. Por padrão, NDN-Waze define o valor da janela para 30 segundos, mas reconfigurações futuras podem ser agendadas.

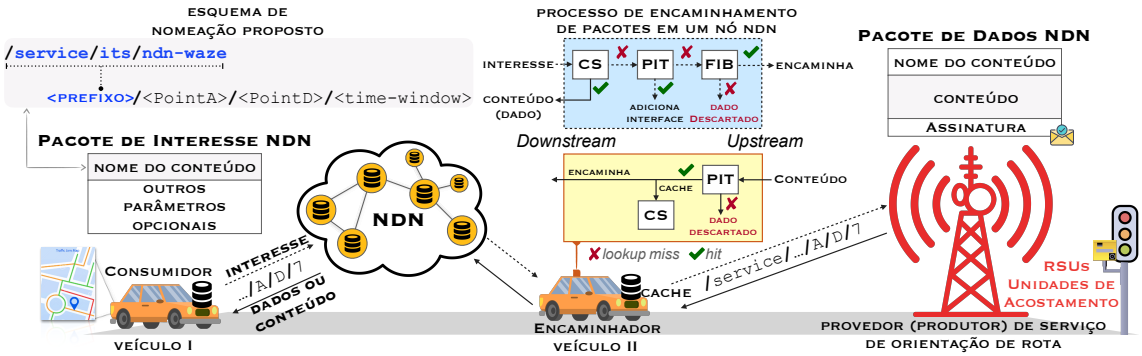


Figura 3. VANET via NDN: modelo de comunicação V2X para C-ITS. Veículo I envia interesse na rede; veículo II encaminha o pacote para alcançar um provedor.

As unidades de acostamento que fornecem o serviço de orientação de rota veicular são configuradas com uma entrada na tabela FIB com o prefixo `/service/its/ndn-waze`, a fim de encaminhar pacotes de interesse para a aplicação NDN-Waze local processar e calcular o caminho veicular mais adequado entre a origem e o destino. O Algoritmo 1 apresenta, em alto nível, o processamento de pacotes de interesse do serviço de orientação de rota veicular nas RSUs. As entradas do algoritmo são o pacote de interesse, o grafo ponderado G_{RV} usado para modelar a rede veicular e a base de dados RTDB local da própria unidade de acostamento, que é utilizada para carregar os pesos no grafo ponderado G_{RV}. Essas entradas são lançadas após um evento de encaminhamento de pacote de interesse ocorrer na `faceId` local na qual a aplicação NDN-Waze está monitorando (i.e., configurada). A linha 2 do algoritmo mostra a validação do prefixo e sufixo do nome e, em caso de falha, o pacote é descartado na linha 6. As linhas 3–4 ilustram os atributos extraídos do nome para o cálculo do caminho adequado. A linha 8 é responsável por validar a janela temporal atual, visando identificar e ignorar requisições antigas e desatualizadas para evitar o desperdício de recursos na disseminação de conteúdo na rede que não é mais relevante para o contexto atual.

O método VERIFICARPESOSGRAFO, linha 9, é responsável por atualizar os pesos das arestas do grafo ponderado G_{RV} com as devidas informações sobre o tempo atual de viagem na respectiva infraestrutura. Todas as informações atualizadas no grafo ponderado vêm da base de dados RTDB (conforme Seção 3.2). Na linha 10, a chamada de

CALCULARCAMINHO abstrai os detalhes do algoritmo de menor caminho no grafo para calcular rotas rápidas de acordo com os atributos já extraídos do nome. NDN-Waze utiliza por padrão uma abordagem baseada no algoritmo DSP (*Dynamic Shortest Path*), onde as rotas são calculadas combinando o menor caminho e o menor tempo de viagem, mas outras propostas disponíveis na literatura [de Souza et al. 2017] podem ser adicionadas ao módulo para usar outros critérios ou pesos, inclusive. As linhas 12–14 mostram o cálculo do tempo total de viagem, em segundos, para a rota R selecionada. Por fim, nas linhas 15–17, um pacote de dados de resposta é preparado para ser enviado na rede. Os parâmetros passados para o novo pacote de dados são: a rota completa (caminho) R para o veículo percorrer, o tempo do percurso e o pacote de interesse de entrada, com alguns atributos relevantes para resposta (e.g., nomeação). A chamada do método ASSINARPACOTENDN (linha 16) encapsula os detalhes do processo de assinatura do pacote de dados NDN, em que a chave privada da respectiva RSU é usada para assinar o conteúdo, e, em seguida, na linha 17, enviar o pacote na rede. Assim, o Algoritmo 1 produz como saída a rota R e o tempo total de percurso por meio de R. A complexidade do Algoritmo 1, no caso do dado não estar na *cache*, é $O(n^2)$, pois a função CALCULARCAMINHO no grafo será invocada.

Algoritmo 1: PROCESSAMENTO DE PACOTES DE INTERESSE DE ORIENTAÇÃO DE ROTA VEICULAR NAS UNIDADES DE ACOSTAMENTO

Entrada: $Pct_Interesse$, $Grafo\ G_{RV}$, $RTDB$
Saída: $Caminho\ R = \{a_{ij}, \dots, a_{mn}\} \mid R \subseteq A(G_{RV}),\ \sum p_{ij} \in R$

1 **início**
2 **se** PREFIXO($Pct_Interesse$) = "`/service/its/ndn-waze`" $\wedge$ SUFIXO(*Pct_Interesse*) **então**
3 $Ponto_Atual, Ponto_Destino, Janela_Interesse \leftarrow$ SUFIXO($Pct_Interesse$, 3);
4 $Janela_Atual \leftarrow (inteiro)\ Timestamp \div Tamanho_Janela_Temporal$;
5 **senão**
6 ERROESQUEMANOMES();
7 **fim**
8 **se** $Janela_Interesse = Janela_Atual$ **então**
9 VERIFICARPESOSGRAFO(G_{RV}, $RTDB$);
10 $R =$ CALCULARCAMINHO(G_{RV}, $Ponto_Atual$, $Ponto_Destino$);
11 $Tempo_Total_Rota = \emptyset$;
12 **para** *cada aresta* $a_{ij} \in R \wedge$ *respectivo peso* $p_{ij} \in P$ **faça**
13 SOMATORIO($Tempo_Total_Rota$, CUSTO(a_{ij}, P));
14 **fim**
15 $Pct_Dados =$NOVOCONTEUDO($R, Tempo_Total_Rota, Pct_Interesse$);
16 ASSINARPACOTENDN(Pct_Dados, $Cert_Digital$);
17 ENVIAR(Pct_Dados);
18 **fim**
19 **fim**

3.4. Modelo de Segurança e Confiança

Em conformidade com a arquitetura NDN, assumimos que existe um processo de inicialização nesta etapa, conhecido como *bootstrap*, para a instalação da aplicação e regras de segurança para a validação dos dados. Basicamente, a inicialização de segurança na NDN é um processo pelo qual as entidades obtêm âncoras de confiança da rede, certificados digitais e aprendem as políticas de confiança de um determinado domínio. Lembre-se que toda aplicação e todo nó que participa de uma comunicação NDN é uma entidade, e toda entidade tem propriedade sobre um ou mais espaços de nomes [Sampaio et al. 2021].

Na NDN, todo pacote de dados é assinado digitalmente, cobrindo o nome, conteúdo e metadados. O campo `KeyLocator` em cada pacote de dados contém a chave pública que foi usada para assinar o pacote de dados NDN [Khelifi et al. 2020]. O processo para validar um pacote de dados recebido pode ser enumerado: (i) o cliente busca a chave especificada no `KeyLocator`; (ii) o cliente valida as propriedades da chave pública, como expiração, revogação e algoritmos de criptografia, e autentica a chave no modelo de confiança proposto (âncoras de confiança e regras de validação); (iii) o cliente verifica a assinatura digital no pacote de dados e valida a autenticidade e integridade do pacote antes de processá-lo; (iv) se o modelo de confiança falhar, o pacote é descartado por não cumprir os requisitos de segurança definidos.

O NDN-Waze conta com um esquema de segurança (Figura 4, rótulo A) para trocas de mensagens assinadas digitalmente, garantindo a autenticidade e integridade de todo o sistema. Basicamente, a aplicação define, na camada 3 da NDN em nível de rede, as políticas de confiança que especificam quais entidades são confiáveis para produzir quais dados, que tipo de chave deve ser usado para determinado espaço de nomes e para qual finalidade. Uma lista de regras de validação, escrita em uma linguagem de domínio específico, flexível, composta por expressões regulares NDN [Khelifi et al. 2020], permite ao nó consumidor atestar e validar um pacote recebido, verificando se este é assinado pela chave correta, de acordo com as políticas e o modelo de confiança definidos.

A Figura 4 ilustra o modelo de confiança proposto para o sistema e demonstra a relação entre as chaves. O Departamento de Transportes (DoT) controla todo o modelo de confiança. Assim, o certificado raiz, autoassinado pela chave DoT, é usado como âncora de confiança da rede (Figura 4, rótulo B). Na NDN, o âncora de confiança é uma Autoridade de Certificação (*Certification Authority* – CA) para um determinado espaço de nomes. A CA vai assinar certificados de diferentes dispositivos na rede, como veículos, unidades de acostamento, e sensores de monitoramento, e certificar que eles são atores confiáveis no sistema (Figura 4, rótulo C). De acordo com o modelo de confiança e as regras de validação instaladas nos nós, um pacote de dados de orientação de rota, sob este domínio hierárquico NDN-Waze, só é válido se for assinado pela chave de uma RSU (Figura 4, rótulo D). Portanto, apenas unidades de acostamento vão propagar orientações de rota veicular válidas na rede. Caso contrário, os veículos irão descartar o pacote por falha de autenticação e integridade no processo. Além disso, os pacotes de dados de monitoramento (i.e., RTDB) só são validados se forem assinados por RSUs ou sensores espalhados na infraestrutura (Figura 4, rótulos E e F).

4. Avaliação Experimental

Experimentos foram realizados neste trabalho para avaliar o modelo VANET via NDN e VANET via IP. A qualidade de serviço de orientação de rota da proposta NDN-Waze também foi avaliada. Esta seção descreve a metodologia, projeto dos experimentos e métricas usadas. Ao final, os resultados obtidos por meio de simulações são apresentados.

4.1. Metodologia

O arcabouço NDN4IVC[1], versão 1.1, foi utilizado para simulação de VANET. O NDN4IVC é um ambiente customizado para simulação de redes veiculares de dados nomeados (VNDN), permitindo a comunicação bidirecional entre o simulador de rede NDN (ns-3/ndnSIM) e o simulador de mobilidade urbana (SUMO) [Araujo et al. 2022].

[1]Disponível em: https://github.com/insert-lab/ndn4ivc — https://youtube.com/@ndn4ivc

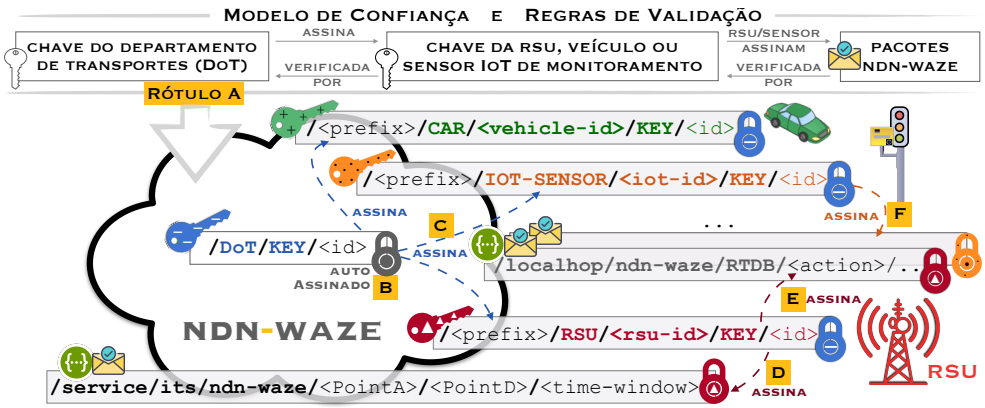


Figura 4. Ilustração do modelo de confiança NDN-Waze: (A) relação entre chaves e entidades; (B) âncora de segurança da rede (CA auto-assinada – DoT); (C) DoT assina certificados de entidades confiáveis na rede; (D, E) RSUs podem assinar dados válidos de monitoramento e orientação de rotas veiculares; (F) Dados de monitoramento assinados por sensores IoT também são válidos no modelo.

Para a simulação dos cenários VANET via IP, a pilha de protocolos do IP disponível no ns-3 também foi instalada no ambiente, em um enlace IEEE 802.11p. Diferentemente do modelo VANET via NDN (Figura 3), em que a aplicação NDN-Waze foi implementada por meio dos serviços disponíveis em nível de rede da NDN, para os cenários VANET via IP, a aplicação NDN-Waze foi implementada em nível de aplicação, por intermédio de mensagens URL de requisição/resposta (veja Figura 1), utilizando UDP como protocolo de transporte e protocolo OLSR para manter a tabela de rotas atualizada dinamicamente nos nós na rede. Nas simulações, ainda se assume que as RSUs estão interconectadas em grade por tecnologias específicas e também fazem o papel de um dispositivo IoT fixo de monitoramento (e.g. sensor/detector, radar, câmera), provendo dados ao sistema de condições atuais de tráfego e mobilidade (e.g., velocidade média) nas vias.

O *trace* de mobilidade utilizado contém dados de tráfego da cidade de Colônia, Alemanha, e está disponível para o SUMO por meio do projeto TAPASCologne[2]. Para fins de simulação, optou-se por um recorte no *dataset*, feito por ferramentas disponíveis no próprio SUMO (e.g., `netconvert`, `cutTrips`), de uma área central de 9km^2. Além disso, para garantir a ocorrência de congestionamento veicular, foi utilizado o intervalo de 12h–14h, onde a densidade de veículos é considerável, com picos de 2340 veículos simultaneamente. A simulação é encerrada quando o último veículo no intervalo termina.

Os experimentos realizados neste trabalho foram conduzidos por meio do modelo fatorial completo [Jain 1991]. O projeto dos experimentos considerou os seguintes fatores e respectivos níveis: [Fator **A** - Arquitetura] = {VANET via NDN, VANET via IP}; [Fator **B** - Densidade] = {30%, 90%}; [Fator **C** - # RSUs/Topologia] = {4, 16}. Os fatores e níveis foram definidos com base em características de aplicações veiculares para proporcionar diferentes cenários de simulação e avaliar o efeito dessas variáveis no ambiente. Os níveis do Fator B (Densidade) representam uma variação no percentual total de veículos na área simulada do cenário (100% representa a densidade original). Todos os experimentos foram analisados a partir do intervalo de confiança[3], da média e do desvio padrão aferidos. Esses parâmetros são utilizados como base para o cálculo da soma dos quadrados, resultando na influência de cada fator nas variáveis de resposta. A Tabela 1 mostra os principais parâmetros de simulação dos experimentos e a topologia das RSUs.

[2]Disponível em: http://kolntrace.project.citi-lab.fr

[3]Utilização da distribuição *t-student* com 10 replicações por experimento e α= 0,05%.

Tabela 1. Parâmetros de Simulação

Tráfego e Mobilidade Urbana
Cenário (SUMO): Colônia, Alemanha, área central simulada de 9 km^2
Número total de veículos: $\approx$ 33700 (picos de 2340 simultaneamente)
IEEE 802.11p — ns3::OcbWifiMac
de antenas tx/rx, ganho tx/rx, limiar de detecção de energia: 1, 1dB, -96 dBm
Canal de comunicação: CH176 (*Service CHannel* 3, Frequência 5.880 GHz)
Potência de transmissão — txPower: 45.6 dBm (raio de comunicação $\approx$ 500 m)
Largura de banda, taxa, modulação: 10 MHz, 6 Mbps, OFDM
Modelo de perda de propagação / atraso de perda: *two-ray ground*, *constant-speed*
Cenários via Internet Protocol (IP)
Protocolo de roteamento e transporte: OLSR/UDP
Cenários via NDN
Estratégia de encaminhamento: *multicast* Tamanho da *cache*: 1000 pkts
NDN-Waze
Intervalo entre *beacons*: 1 s Duração da janela *time-window*: 30 s
 # RSUs, Topologia, Distância (d) e Posicionamento
04 = Grade2x2 (d=500 m): RSU1 (12.3km,12.3km), RSU4 (12.8km,12.8km)
16 = Grade4x4 (d=500 m): RSU1 (12.3km,12.3km), RSU16 (13.8km,13.8km)

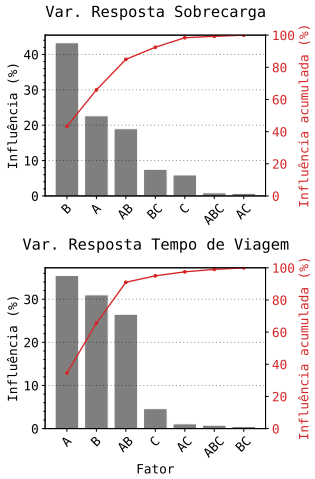


Figura 5. Influência/Fator

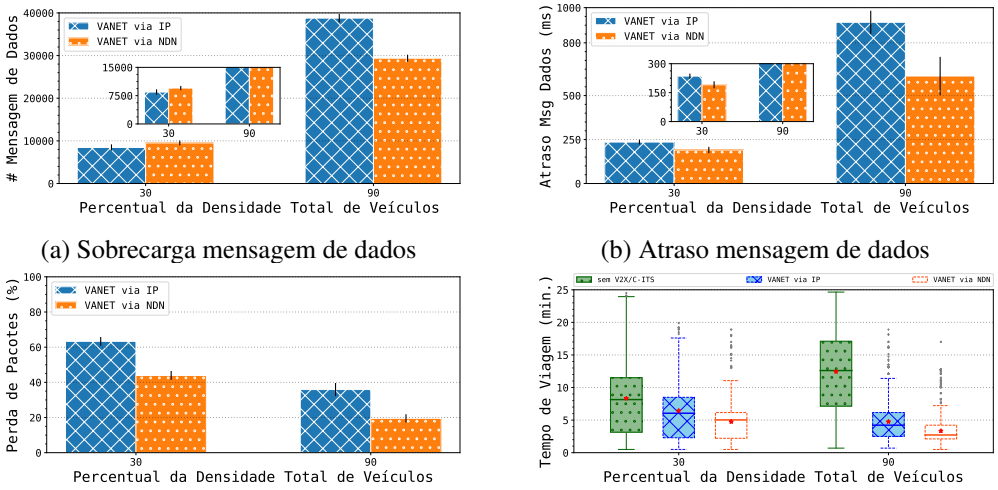
As métricas utilizadas tiveram dois objetivos: (i) analisar a escalabilidade das arquiteturas para VANET e (ii) avaliar a qualidade do serviço de orientação de rota – NDN-Waze. Para o caso (i), as variáveis de resposta foram: sobrecarga (total de mensagens de dados transmitidas), atraso (tempo médio de entrega das mensagens) e taxa de perda de pacotes. Em (ii), a variável de resposta mais relevante foi tempo médio de viagem.

Ao analisar os experimentos, verificou-se primeiramente a influência dos fatores sobre as métricas propostas. Para as variáveis de resposta das métricas de escalabilidade (i), como era esperado, o fator de maior influência foi a Densidade (Fator B), seguido da Arquitetura (Fator A). Contudo, os experimentos também mostraram uma forte interação dos Fatores A e B, avaliada em mais detalhes na Seção 4.2. A Figura 5 ilustra as influências dos fatores para sobrecarga (métrica de escalabilidade) e qualidade de serviço da aplicação NDN-Waze (métrica QoS). Como em ambas as variáveis de resposta o Fator A (Arquitetura) teve alta influência nos resultados, essas informações confirmam a hipótese de que a escolha da arquitetura produz, sim, alterações significativas nos resultados.

4.2. Análise dos Resultados e Discussões

As Figuras 6a, 6b, 6c permitem avaliar a escalabilidade das arquiteturas para VANET. A Figura 6a mostra a sobrecarga de mensagens de dados na rede. Nota-se que em cenários com baixa densidade de veículos, a influência da arquitetura (Fator A) é pequena. Nesse contexto de baixa densidade de veículos na rede, com 30% da densidade total, como não existe cobertura (conectividade) das RSUs em todo cenário de simulação, para o raio de comunicação de 500 metros (veja a Tabela 1), muitos veículos enviam interesses na rede mas não conseguem alcançar provedores de serviço, nem por meio da comunicação *vehicle-to-vehicle*. Isso explica o comportamento similar observado. Entretanto, verifica-se que em cenários com maior densidade de veículos existe influência da arquitetura. Em cenários com 90% da densidade, VANET via IP apresenta sobrecarga de 32% comparado com VANET via NDN. Esse aumento é atribuído a questões de roteamento *stateless* do IP e à falta de *caches* de rede, disponíveis na NDN, provocando mais retransmissões.

O comportamento das arquiteturas em relação à variável de resposta atraso pode ser observado na Figura 6b. Os resultados agrupados pela densidade mostram que mesmo variando a densidade de veículos na rede, a arquitetura VANET via NDN apresentou um menor atraso para a entrega de mensagens de dados de orientação de rota. Tal comportamento indica uma maior eficiência da NDN para encaminhamento de pacotes em VANET.



(a) Sobrecarga mensagem de dados

(b) Atraso mensagem de dados

(c) Perda de pacotes

(d) Tempo médio de viagem

Figura 6. Resultados avaliação variáveis de resposta, agrupados pela densidade

A Figura 6c mostra perda de pacote por densidade (Fator B). Em cenários com baixa densidade de veículos, VANET via NDN apresenta uma taxa de perda de aproximadamente 43%, enquanto que VANET via IP supera os 60%. Em configurações com maior densidade de veículos, pode-se verificar que VANET via NDN ainda performa melhor, com perda média de aproximadamente 20%, com VANET via IP se aproximando dos 36%. Espera-se taxa de perda menor em áreas congestionadas, devido maior número de veículos e menor mudança de topologia. No caso específico da aplicação NDN-Waze, não há problemas com cenários com baixa densidade, pois veículos computam rotas iniciais nas próprias unidades de bordo, e quando forem entrando em área de cobertura de RSU ou atingindo regiões congestionadas vão ter suporte do serviço de orientação de rota.

Para a variável de resposta qualidade de serviço, VANET via NDN também ajudou a aplicação NDN-Waze a performar melhor nos experimentos. A Figura 6d mostra que em cenários VANET via NDN, a aplicação NDN-Waze alcança melhores tempos médios de viagem, mesmo em densidades diferentes. Por exemplo, em cenários VANET via NDN com 90% da densidade total de veículos, quando comparado com cenários sem V2X/C-ITS (*baseline*, sem alteração de caminho) e VANET via IP, a aplicação NDN-Waze reduz o tempo médio de viagem dos veículos em quase 78% e 34%, respectivamente.

5. Conclusões e Trabalhos Futuros

Este trabalho apresenta o *design* completo de uma aplicação C-ITS de orientação de rota veicular para redes de dados nomeados. Além disso, contribui com maior nível de detalhes para o conhecimento sobre a arquitetura NDN, endereçando problemas relevantes do contexto veicular em que a NDN resolve de forma nativa, por meio de serviços de rede (camada 3), facilitando o desenvolvimento de aplicações leves, seguras e distribuídas para VANET. Um estudo experimental detalhado também avaliou vantagens do modelo VANET via NDN sobre VANET via IP, para encaminhamento de dados na rede por meio da comunicação V2X entre veículos e RSUs. Os experimentos demonstram vantagens da arquitetura NDN em relação à escalabilidade e qualidade de serviço de aplicações no topo da pilha de protocolos. Métricas tradicionais de rede como número de mensagens transmitidas, atraso médio e perda de pacotes dão sustentação à hipótese de que a arquitetura para comunicação entre veículos interfere diretamente no desempenho, eficiência e funcionamento de serviços de certas aplicações distribuídas em VANETs. Por exemplo:

em cenários com maior densidade de veículos, os resultados de simulações comprovam que VANET via NDN chega a diminuir o tempo médio de viagem dos veículos em 34%, comparado com a arquitetura VANET via IP. Além disso, ao agrupar cenários com 90% da densidade de veículos, verifica-se que VANET via NDN, comparado com VANET via IP, reduz a sobrecarga em 24%, reduz atraso médio para entrega de dados em 33%, mantendo a taxa de perda média de pacotes no menor patamar, i.e., 20%. Em trabalhos futuros, pretende-se escalonar veículos estacionários para atuar como RSUs virtuais. Planeja-se também adicionar ao módulo CALCULARCAMINHO, Algoritmo 1, mais informações sobre o nível de congestionamento para cálculo otimizado de rotas veiculares.

Agradecimentos

Os autores gostariam de agradecer o apoio concedido do CNPq (N° 432064/2018-4, 316208/2021-3, 402854/2022-5) e FAPESB (N^o TIC0004/2015).

Referências

Araujo, G., Peixoto, M., and Sampaio, L. (2022). Ndn4ivc: A framework for simulations of realistic vanets applications through ndn. In *Proceedings of the 9th ACM Conference on Information-Centric Networking*, ICN'22, page 162–164, New York, USA. ACM.

de Souza, A. M., Brennand, C. A., Yokoyama, R. S., Donato, E. A., Madeira, E. R., and Villas, L. A. (2017). Traffic management systems: A classification, review, challenges, and future perspectives. *International Journal of Distributed Sensor Networks*, 13(4).

Hussein, N. H., Yaw, C. T., Koh, S. P., Tiong, S. K., and Chong, K. H. (2022). A comprehensive survey on vehicular networking: Communications, applications, challenges, and upcoming research directions. *IEEE Access*, 10:86127–86180.

Jain, R. (1991). *The art of computer systems analysis - techniques for experimental design, measurement, simulation, and modeling.* Wiley professional computing. Wiley.

Khelifi, H., Luo, S., Nour, B., Moungla, H., Faheem, Y., Hussain, R., and Ksentini, A. (2020). Named data networking in vehicular ad hoc networks: State-of-the-art and challenges. *IEEE Communications Surveys & Tutorials*, 22(1):320–351.

Magdum, S. S., Sharma, M., Kala, S. M., Antony Franklin, A., and Tamma, B. R. (2019). Evaluating dtn routing schemes for application in vehicular networks. In *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*.

Patel, P., Narmawala, Z., and Thakkar, A. (2019). A survey on intelligent transportation system using internet of things. In Shetty, N. R., Patnaik, L. M., Nagaraj, H. C., Hamsavath, P. N., and Nalini, N., editors, *Emerging Research in Computing, Information, Communication and Applications*, pages 231–240, Singapore. Springer Singapore.

Sampaio, L., Freitas, A. E. S., Brito, I. V. S., Araújo, F. R. C., and Ribeiro, A. V. (2021). Revisitando as icns: Mobilidade, segurança e aplicações distribuídas através das redes de dados nomeados. In *Minicursos do XXXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 51–100. SBC, Porto Alegre, RS, Brasil.

Storck, C. R. and Duarte-Figueiredo, F. (2019). A 5g v2x ecosystem providing internet of vehicles. *Sensors*, 19(3).

Toutouh, J., Garcia-Nieto, J., and Alba, E. (2012). Intelligent olsr routing protocol optimization for vanets. *IEEE Transactions on Vehicular Technology*, 61(4):1884–1894.