

Capítulo

4

Redes Neurais de Grafos como Gêmeo Digital de Rede: Aplicações em Redes 5G Avançado e em 6G

Antonio José Silvério (Claro), Hélio M. Barroso (Cisco) e Miguel Elias M. Campista (UFRJ)

Resumo

As redes 5G-Advanced e 6G impõem requisitos rigorosos de desempenho à infraestrutura de telecomunicações. Serviços avançados, fatiamento de rede, virtualização e mobilidade exigem a coordenação contínua de recursos entre rede de acesso de rádio (Radio Access Network – RAN), núcleo de rede móvel e fixa, borda/nuvem e transporte, reduzindo a eficácia de abordagens baseadas apenas em ciclos lentos, estados agregados e validações predominantemente manuais. Este minicurso apresenta os Gêmeos Digitais de Rede (Network Digital Twins – NDTs) como arquiteturas para representar, sincronizar e validar o comportamento da infraestrutura antes da atuação física. Em vez de réplicas integrais ou simuladores isolados, NDTs integram dados, modelos, mapeamentos, interfaces e gestão do ciclo de vida para compor gêmeos de domínio ou serviço com granularidade e fidelidade adequadas à tomada de decisão. As redes neurais de grafos são introduzidas para explorar a topologia e as relações entre as entidades da rede. Por meio do aprendizado em grafos formados por nós, enlaces e recursos, o mecanismo de passagem de mensagens permite apoiar tarefas de previsão, classificação, modelagem de desempenho e decisão. A abordagem foca no transporte IP, abrangendo previsão de demanda, planejamento de capacidade e engenharia de tráfego. Três estudos de caso ilustram essas capacidades: a rede recorrente convolucional de grafos espaço-temporais combina dinâmica espaço-temporal para prever matrizes de tráfego; a família RouteNet estima indicadores-chave de rede (Key Performance Indicators – KPIs), tais como atraso, jitter e perda em análises what-if; e o TEAL utiliza FlowGNN, aprendizado multiagente e Método dos Multiplicadores de Direções Alternadas (Alternating Direction Method of Multipliers – ADMM) para produzir alocações candidatas de tráfego em grande escala. Por fim, examinam-se os desafios operacionais tais como sincronização, observabilidade, custo computacional e explicabilidade além do papel das arquiteturas ETSI ZSM (Zero-touch network and Service Management) e dos níveis de autonomia do TM Forum. Conclui-se que a automação e a autonomia de redes resultam da integração sistêmica entre engenharia, dados, otimização e modelos, e não de um algoritmo isolado.

4.1. Introdução

As Redes móveis de nova geração são frequentemente apresentadas por meio de taxas de transmissão, densidades de dispositivos e metas de latência cada vez mais ambiciosas. Esses indicadores são importantes, mas não descrevem sozinhos o problema de engenharia que se forma por trás deles. Para o operador, a dificuldade central não está apenas em entregar mais capacidade, e sim em coordenar recursos de rádio, transporte, funções virtualizadas, computação de borda e políticas de serviço em condições que variam no tempo e no espaço. A infraestrutura precisa acomodar simultaneamente aplicações de alto volume, comunicações críticas, dispositivos de baixa potência, serviços imersivos, cargas de inteligência artificial e conectividade distribuída em ambientes terrestres e não terrestres. Essa transformação não invalida as práticas tradicionais de planejamento e operação. Séries históricas, margens de segurança, matrizes de tráfego, simulações de falha, engenharia de tráfego e expansão preventiva de capacidade continuam indispensáveis. O problema é que essas práticas foram concebidas para ciclos de decisão relativamente lentos, fronteiras administrativas mais claras e requisitos que podiam ser tratados de forma predominantemente agregada. À medida que serviços e infraestrutura se tornam mais programáveis e interdependentes, previsões estáticas, estados incompletos e validações demoradas resultam em decisões menos confiáveis ou tardias.

Para se ter uma ideia, a Recomendação ITU-R M.2083 organizou a visão do IMT-2020 (padrão 5G) em três cenários básicos de uso: a banda larga melhorada (*enhanced Mobile Broadband* – eMBB), as comunicações de missão crítica (*Ultra-Reliable and Low-Latency Communications* – URLLC) e a comunicação massiva de máquina (*massive Machine-Type Communications* – mMTC). Esses cenários não correspondem a três redes independentes, mas representam combinações distintas de capacidade, latência, confiabilidade, mobilidade, densidade e eficiência energética [ITU-R, 2015]. Já o 5G-Advanced, iniciado pelo 3GPP no Release 18, não substitui essa estrutura, mas amplia a variedade de condições em que ela precisa operar, aprofundando o fatiamento de redes e o suporte a sistemas não terrestres [3GPP, 2025b]. O Release 19 dá continuidade a esse movimento ao consolidar a inteligência artificial tanto para a otimização da rede (*AI for Network*) quanto como carga de trabalho suportada (*Network for AI*), introduzindo os Gêmeos Digitais de Rede (*Network Digital Twins* – NDT) para viabilizar o gerenciamento em ciclos fechados (*closed loops*) [3GPP, 2026b]. A relevância dessas evoluções reside no severo aumento das dependências entre as funcionalidades, cenário que o 6G (IMT-2030) estende a seis cenários que consolidam o sensoriamento e a comunicação integrados (*Integrated Sensing and Communication*) e a conectividade ubíqua [ITU-R, 2023]. Essa engenharia real envolve compromissos complexos, exigindo uma evolução gradual e economicamente sustentável que corrija lacunas de complexidade sem fragmentar a infraestrutura [NGMN Alliance, 2023].

Essa natureza multidomínio exige que os requisitos fim-a-fim percebidos pelo usuário sejam decompostos entre a rede de acesso, o núcleo da rede, a borda computacional e o transporte. O fatiamento de rede (*network slicing*) torna essa interdependência explícita, demandando que o sistema de gestão traduza variações de capacidade em requisitos coordenados de sub-redes e de transporte, conforme estabelece a norma TS 28.530 [3GPP, 2025a]. De forma análoga, a garantia de latência em ambientes de computação de borda depende de uma estreita coordenação entre o posicionamento das

aplicações e o orçamento de atraso da interface aérea e do transporte [ETSI, 2019a]. Adicionalmente, a introdução de redes não terrestres insere dinamicamente enlaces mutáveis e coberturas descontínuas, forçando o planejamento a lidar com múltiplos regimes operacionais simultâneos [3GPP, 2025b].

Diante desse dinamismo, o planejamento de capacidade enfrenta o desafio de projetar a demanda sob tráfegos não estacionários e espacialmente móveis. A visibilidade dessa demanda na infraestrutura IP costuma ser incompleta, pois os contadores de interface revelam a carga do enlace, mas ocultam a matriz de origem e destino, historicamente estimada por técnicas de tomografia de tráfego, como o *Tomogravity* [Zhang et al., 2003a]. Há também a necessidade de distinguir o tráfego medido da demanda real oferecida, uma vez que gargalos e mecanismos de controle reajustam o tráfego observado e mascaram a real necessidade de expansão. O sobreprovisionamento (*overprovisioning*) surge como uma resposta tradicional a essas incertezas, mas a busca por eficiência energética e sustentabilidade tensiona esse equilíbrio, pois desligar recursos para poupar energia reduz a margem protetora contra picos inesperados.

Dessa forma, a evolução das Redes Móveis pressiona quatro funções clássicas: predição (*forecasting*), planejamento de capacidade (*capacity planning*), engenharia de tráfego (*Traffic Engineering* – TE) e análise *what-if*. A engenharia de tráfego opera como um problema multiobjetivo sob restrições para transformar a capacidade disponível em desempenho utilizável, equilibrando riscos de falhas e metas de atraso [Bogle et al., 2019]. Contudo, os requisitos estritos do 5G-Advanced e 6G reduzem a janela temporal útil para essa análise. Embora o fluxo de telemetria e novos indicadores qualifiquem a observabilidade [3GPP, 2025b], os dados brutos precisam ser rapidamente convertidos em decisões operacionais. É nesse ponto que a análise *what-if* se torna o gargalo, pois modelos analíticos simplificados sacrificam a fidelidade e simulações detalhadas são lentas demais para janelas críticas. Essa limitação justifica o emprego de modelos de desempenho aprendidos, como a arquitetura *RouteNet*, capazes de estimar indicadores de rede em alta velocidade a partir de topologias e de matrizes de tráfego [Rusek et al., 2020]. A automação e as malhas de controle supervisionadas surgem, portanto, não para substituir a engenharia clássica, mas para fornecer ambientes céleres de validação antes de qualquer atuação na rede real [3GPP, 2026b].

Este minicurso está organizado para abordar o encadeamento de desafios envolvidos nas redes móveis 5G Avançado e 6G. A Seção 4.2 define os gêmeos digitais de rede como a arquitetura unificadora para integrar dados, modelos, experimentação e validação. Em seguida, a Seção 4.3 apresenta as redes neurais de grafos (*Graph Neural Networks* – GNNs) como a família de modelos ideal para representar topologias de rede e suas dependências internas. Na sequência, a Seção 4.6 examina arquiteturas como a *Spatiotemporal Graph Convolutional Recurrent Network* (SGCRN), a família *RouteNet* e o *framework TEAL* como respostas complementares e profundas aos problemas práticos de previsão de demanda, modelagem de comportamento e decisões automatizadas em engenharia de tráfego. Os desafios em aberto e as tecnologias habilitadoras são apresentados na Seção 4.7 e a relação deste trabalho com as futuras Redes Autônomas é discutida na Seção 4.8. Por fim, a Seção conclui este minicurso.

4.2. Gêmeos Digitais de Redes 5G Avançado e 6G

Um Gêmeo Digital de Redes (*Network Digital Twin* – NDT) deve ser entendido por sua função operacional, não pelo grau de realismo visual nem por uma tecnologia específica. Ele delimita uma contraparte, organiza dados e modelos, mantém mapeamentos, expõe interfaces, gerencia execuções e fornece um ambiente no qual hipóteses e ações podem ser avaliadas sem expor diretamente a rede. Em 5G/6G, gêmeos orientados a domínio permitem representar a RAN, o Core, o transporte e o edge com entidades e escalas apropriadas. Gêmeos orientados a serviço ou fatia de rede (*slice*) coordenam essas perspectivas para verificar requisitos fim a fim. A composição evita exigir uma cópia monolítica de toda a operadora, mas introduz desafios de interoperabilidade, confiança e consistência que serão retomados no capítulo sobre limitações e tecnologias habilitadoras. Resta compreender como representar computacionalmente as relações presentes nesses sistemas. Os elementos da rede não são observações independentes: caminhos compartilham enlaces, usuários interferem entre si, funções dependem de recursos e fatias de redes compartilham infraestrutura.

Um Gêmeo Digital de Rede não é uma rede neural de grafo (*Graph Neural Network* – GNN), um simulador ou uma cópia integral da rede. É uma arquitetura orientada a casos de uso que associa uma contraparte física a dados, modelos, mapeamentos, interfaces, gestão e aplicações, criando um espaço seguro para analisar, experimentar, prever e validar decisões antes de sua eventual aplicação. A discussão da introdução suscita três perguntas: que demanda chegará, como a rede se comportará e que decisão deverá ser tomada. Nenhuma delas pode ser respondida apenas por contadores, alarmes ou inventários. Dados descrevem partes do estado observado, mas não revelam diretamente como uma configuração alternativa afeta o serviço, como uma falha se propaga entre domínios ou que risco acompanha uma ação proposta. Entre observar a rede e modificá-la, existe, portanto, uma lacuna de modelagem e experimentação. O Gêmeo Digital de Rede (*Network Digital Twin* – NDT) ocupa esse espaço.

Um gêmeo digital organiza dados e modelos de modo que a rede, uma parte dela ou um serviço possa ser representado digitalmente e submetido a cenários que seriam caros, lentos ou perigosos na infraestrutura ativa. O resultado pode apoiar planejamento, testes, treinamento, análise “*what-if*”, garantia e otimização. A atuação sobre a rede pode ocorrer depois, por um operador, controlador, orquestrador ou controlador inteligente de rede de acesso via rádio (*RAN Intelligent Controller* – RIC), mas não deve ser confundida com a função do próprio gêmeo digital. Nas redes 5G-Advanced e nas futuras redes 6G, essa distinção é particularmente importante. A rede de acesso via rádio (*Radio Access Network* – RAN), núcleo, transporte, borda e nuvem possuem objetos, estados e escalas temporais distintos. Um serviço fim-a-fim depende da composição desses domínios, mas não exige necessariamente uma réplica centralizada e perfeita de toda a operadora. Gêmeos digitais orientados a domínio, subsistema, fatia ou serviço podem compartilhar abstrações, previsões e respostas a consultas, preservando o detalhamento apenas onde necessário.

4.2.1. Modelo de Rede ao Gêmeo Digital de Rede

O grupo de pesquisa de gerenciamento de redes do IETF (*Internet Engineering Task Force*), IRTF/NMRG (*Internet Research Task Force/Network Management Research Group*), define os gêmeos digitais de rede como uma representação digital usada no contexto de redes, cuja contraparte física é uma rede de dados. A ITU-T utiliza o termo Gêmeo Digital de Rede e o descreve como uma representação virtual da rede física, baseada em dados, modelos e interfaces, útil para análise, diagnóstico, emulação e controle. O 3GPP, por sua vez, restringe sua definição ao domínio móvel: uma réplica virtual de uma rede móvel, ou de parte dela, que captura atributos, comportamento e interações. As formulações diferem em escopo e ênfase, mas convergem na associação entre uma contraparte real, uma representação digital e mecanismos que permitem utilizar essa representação na operação [IRTF NMRG, 2026, ITU-T, 2022a, 3GPP, 2025c, 3GPP, 2026a].

A definição de Gêmeo Digital de Rede, porém, não deve ser reduzida à ideia de uma imagem visual ou de um inventário. Um modelo estático pode representar nós, enlaces, células, funções, capacidades e políticas, mas pode permanecer desconectado do estado atual e do ciclo operacional. Um simulador reproduz comportamentos por meio de modelos matemáticos ou da definição de eventos; um emulador executa componentes realizáveis, software de rede ou implementações aproximadas do sistema real. Ambos podem compor um NDT, mas nenhum deles, isoladamente, garante mapeamento com a rede, gestão de ciclo de vida ou integração com aplicações operacionais.

A literatura também diferencia modelo digital (*digital model*), sombra digital (*digital shadow*) e gêmeo digital (*digital twin*) pela direção e pela automação dos fluxos de dados. Em uma taxonomia didática, o modelo digital é atualizado manualmente; a sombra recebe dados do sistema físico, mas não necessariamente participa de um fluxo de retorno; o gêmeo admite sincronização parcial ou total em ambas as direções. Essa classificação não é universal e deve ser lida como um contínuo de integração, não como um teste binário.

O tempo real exige cautela semelhante. A norma do ITU-T Y.3090 enfatiza o mapeamento interativo em tempo real, e vários trabalhos sobre o 6G adotam essa visão como objetivo. Entretanto, o próprio *draft* do IRTF afirma que a interação em tempo real não é obrigatória em todos os casos. Uma avaliação de configuração pode utilizar dados congelados e operar como uma *sandbox* isolada; um gêmeo de planejamento pode operar com agregações horárias; um gêmeo de mobilidade pode exigir atualizações em segundos ou milissegundos. A pergunta correta não é se todo NDT opera continuamente em tempo real, mas se sua frequência, granularidade e atraso são compatíveis com a decisão apoiada.

4.2.2. Elementos e Arquitetura Funcional do NDT

O *draft* do IRTF/NMRG caracteriza o NDT por cinco elementos: dados, modelos, mapeamento, interfaces e lógica. A ITU-T Y.3090 organiza a arquitetura em três camadas: infraestrutura física, camada de gêmeos e aplicações, separando, na camada de gêmeos, os domínios de dados, modelos e gestão. O 3GPP acrescenta uma formalização operacional baseada em funções reutilizáveis, em execuções de cenários e em relatórios. A arquitetura adotada nesta seção combina essas perspectivas sem supor que exista uma única implementação obrigatória [IRTF NMRG, 2026, ITU-T, 2022a, 3GPP, 2025c, 3GPP, 2026a].

Tabela 4.1. Diferenciação didática entre modelos, simuladores, emuladores, sombras digitais e NDTs.

Artefato	Atualização pela rede	Reprodução de comportamento	Cenários “ <i>what-if</i> ”	Integração operacional
Modelo/inventário	manual ou eventual	limitada	limitados	geralmente fraca
Simulador	por importação	por abstração matemática	sim	não obrigatória
Emulador	por configuração	por componentes realizáveis	sim	não obrigatória
Sombra digital	predominantemente físico → digital	possível	possível	unidirecional ou parcial
NDT	conforme o caso de uso	sim, no escopo relevante	sim	dados, gestão, aplicações e eventual retorno

A contraparte física define o objeto do gêmeo. Ela pode ser uma rede inteira, uma região, uma RAN, um núcleo de rede, uma rede de transporte, uma infraestrutura de borda, uma fatia de rede ou apenas os recursos necessários para representar determinado serviço. O escopo precisa ser definido antes da coleta e da modelagem. Caso contrário, a ambição de reproduzir tudo aumenta o custo, reduz a capacidade de validação e pode incluir detalhes sem utilidade para a decisão [IRTF NMRG, 2026, Lin et al., 2023].

O mapeamento dos NDT associa cada entidade digital à contraparte física e registra quais atributos são observados, inferidos ou sintetizados. O mapeamento também controla o escopo e a validade temporal. Em redes virtualizadas, uma função pode mudar de hospedeiro, pode escalar ou mesmo desaparecer. Em RAN, usuários e canais variam continuamente. Em fatiamento de redes, o mesmo recurso físico pode sustentar várias redes lógicas. O IRTF distingue mapeamento vertical, entre rede e gêmeo digital, e horizontal, entre gêmeos. O valor operacional do NDT depende menos de uma aparência visual idêntica à rede e mais da consistência dessas relações [IRTF NMRG, 2026, ITU-T, 2022a].

Os modelos podem ser básicos ou funcionais. Modelos básicos descrevem elementos, topologia, recursos, configurações e dependências. Modelos funcionais reproduzem ou estimam o comportamento para análise, diagnóstico, previsão, garantia e otimização. Os modelos podem ser analíticos, baseados em filas, simulação de eventos, emulação de software, modelos físicos, grafos de conhecimento (*knowledge graphs*), otimização, aprendizado de máquina ou combinações desses mecanismos. O 3GPP diferencia explicitamente simulação, baseada em modelos matemáticos, de emulação, baseada em componentes realizáveis. A escolha depende de fidelidade, custo, escala e fenômeno de interesse [ITU-T, 2022a, 3GPP, 2025c, Lin et al., 2023]. A gestão do NDT coordena dados, modelos, recursos computacionais, versões, segurança, sessões de uso e associações com aplicações. Ela também precisa monitorar o próprio gêmeo digital: precisão, consumo, validade, deriva (*drift*) e falhas internas. Um modelo que já não representa a rede não pode continuar produzindo recomendações com o mesmo nível de confiança. O relatório da RAN aberta (*Open RAN – O-RAN*) sobre habilitadores propõe o gerencia-

mento de confiabilidade (*trustworthiness management*) e acompanhamento do desvio dos dados (*data drift*) justamente para quantificar a distância entre entradas e resultados do gêmeo digital e o sistema real [ITU-T, 2022a, 3GPP, 2026a, O-RAN nGRG, 2024a].

As aplicações formulam consultas e consomem resultados. Planejamento, testes, “*what-if*”, soluções de problemas (*troubleshooting*), previsão, garantia, treinamento e otimização são aplicações recorrentes. Elas podem enviar uma configuração candidata, um cenário de falha, uma intenção ou requisitos de desempenho. O gêmeo digital retorna KPIs, riscos, violações, dados sintéticos ou comparações entre alternativas. Almasan et al. sintetizam esse fluxo ao representar a topologia, o tráfego, o roteamento e as políticas como entradas de um modelo que produz atraso, utilização ou anomalias para aplicações de rede [Almasan et al., 2022].

Interfaces conectam a rede, a camada de gêmeo digital, as aplicações e, eventualmente, outros gêmeos. As interfaces devem permitir coleta periódica, por assinatura, sob demanda ou orientada a eventos; também devem expor capacidades, requisitos e resultados de forma interoperável. No O-RAN, as interfaces O1, O2 e E2 podem alimentar e conectar um DT-RAN (*Digital Twin-RAN*) a um Gerenciador e Orquestrador de Serviços (*Service Management and Orchestration* – SMO), RIC em tempo não real (*Non-Real-Time RIC* – Non-RT RIC), RIC em tempo quase real (*Near-Real Time RIC* – Near-RT RIC) e, respectivamente, suas aplicações rApps e xApps. Para ambientes multidomínio, interfaces entre gêmeos precisam trocar abstrações suficientes sem exigir exposição irrestrita de dados internos [IRTF NMRG, 2026, ITU-T, 2022a, O-RAN nGRG, 2024a].

Finalmente, a lógica transforma dados e modelos em análise e decisão. O NDT pode atuar como *sandbox* dentro de um ciclo interno (*inner loop*) que testa alternativas, enquanto um ciclo externo (*outer loop*) detecta desvios e coordena a atuação. No modelo do 3GPP, o NDT apoia funções de decisão, mas não configura diretamente a rede ativa. A configuração é autorizada e executada por outros componentes. Essa separação permite inserir aprovação humana, políticas, *guardrails* e *rollback* sem reduzir o gêmeo a um controlador [IRTF NMRG, 2026, 3GPP, 2025c].

4.2.3. Escopo, Ciclo de Vida e Qualidade do NDT

A formalização do 3GPP ajuda a separar a capacidade persistente de modelagem de uma execução específica. A NDT *Function* oferece recursos de simulação, emulação ou modelagem. Um NDT *Job* descreve o cenário a ser executado, incluindo escopo, entradas, alterações virtuais, período e requisitos de precisão ou latência. A execução produz um ou mais NDT *Reports*. Uma nova tarefa (*job*) não implica necessariamente criar uma plataforma ou réplica independente; ele pode reutilizar a mesma função e os mesmos modelos sob parâmetros diferentes [3GPP, 2026a]. Essa separação permite tratar o NDT como serviço. Uma função pode ser mantida e calibrada ao longo do tempo, enquanto tarefas são criadas para testar uma política de economia de energia, reproduzir uma tempestade de sinalização (*signaling storm*), avaliar uma falha, gerar dados para treinamento ou comparar configurações. O ciclo inclui criação, inicialização, sincronização, execução, suspensão, conclusão, relato, atualização e encerramento. Cada etapa consome recursos e precisa ser auditável [3GPP, 2026a].

A fidelidade não é uma propriedade única. Ela depende do fenômeno modelado e

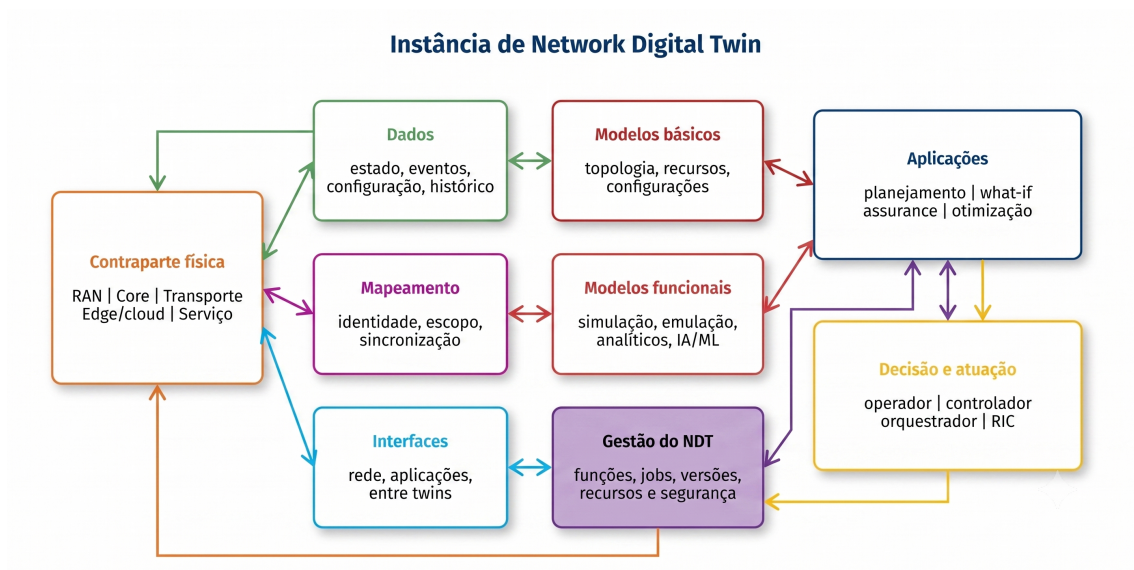


Figura 4.1. Arquitetura funcional consolidada de uma instância de Gêmeo Digital de Rede (Network Digital Twin – NDT). Elaboração própria da figura a partir de IRTF/NMRG, ITU-T Y.3090, 3GPP e literatura acadêmica.

das métricas de interesse. Um gêmeo digital para cobertura pode representar propagação, antenas e obstáculos, mas dispensar detalhes completos do núcleo da rede. Um gêmeo para planejamento de capacidade pode operar com fluxos agregados e horizontes longos; um gêmeo digital para avaliar a transferência de sinal (*handover*) precisa de mobilidade e rádio em escala curta. O relatório O-RAN mostra que um mesmo elemento pode ser representado por simulador, emulador ou modelo de IA, e que apenas as funções relevantes ao estudo precisam ser reproduzidas [O-RAN nGRG, 2024a, Lin et al., 2023].

Há um compromisso inevitável entre granularidade, sincronização, custo e escala. Clonar integralmente pequenas redes com software real pode produzir elevada fidelidade, mas é impraticável para redes extensas. Modelos matemáticos e aprendidos reduzem custo, mas precisam demonstrar validade para topologias, cargas e condições não vistas. Lin et al. distinguem interfaces em tempo real, quase tempo real e não tempo real [Lin et al., 2023]. O IRTF recomenda controlar o ritmo e o escopo do espelhamento, o 3GPP permite tarefas sobre estados passados, presentes ou futuros, inclusive alimentados por previsão [IRTF NMRG, 2026, Lin et al., 2023, 3GPP, 2026a].

A qualidade do gêmeo digital precisa ser tratada como dado operacional. Métricas de erro, confiança, cobertura do domínio de operação, desvio de entrada e diferença entre previsão e medição devem acompanhar o resultado. No exemplo de mobilidade do relatório O-RAN, uma recomendação de *handover* não deve ser avaliada apenas pela confiança do modelo de aprendizado de máquina, mas também por quão próximo o DT-RAN está do sistema real. Quando o "drift" cresce, o gêmeo digital deve ser recalibrado, limitando seu escopo ou rebaixando a autonomia da aplicação que o consome [O-RAN nGRG, 2024a].

4.2.4. NDTs Orientados a Domínio

O NDT não precisa representar todos os domínios com a mesma profundidade. Cada twin seleciona entidades, dados e modelos adequados ao problema. Lin et al. descrevem DTNs modulares para células, RAN, transporte, núcleo, borda, nuvem ou toda a rede fim-a-fim. O O-RAN nGRG reforça que diferentes gêmeos operam com informações, regras e escalas temporais distintas; uma solução única não atende igualmente a todos os casos [Lin et al., 2023, O-RAN nGRG, 2024b].

No DT-RAN, as entidades podem incluir O-RU, O-DU, O-CU, RICs, células, usuários, canais, antenas, recursos de rádio, mobilidade e ambiente físico. Seus modelos combinam propagação, tráfego, funções RAN, nuvem e comportamento dos usuários. O relatório O-RAN seleciona cinco casos prioritários: treinamento, avaliação e garantia de IA/ML; automação de testes; planejamento; economia de energia; e otimização específica por site. O DT-RAN pode gerar cenários, testar xApps e rApps, comparar o posicionamento de antenas, calibrar a cobertura e avaliar políticas de desligamento sem arriscar a rede ativa [O-RAN nGRG, 2024b].

A representação do ambiente físico diferencia muitos DT-RAN. Mapas 3D, LiDAR, câmeras, CAD, topografia, materiais e modelagem por traçado de raios ajudam a reproduzir reflexão, difração, bloqueios e interferência. O modelo precisa ser calibrado com medições como as métricas fundamentais de potência e qualidade de sinal, como o RSRP (*Reference Signal Received Power*), RSSI (*Received Signal Strength Indicator*), SINR (*Signal to Interference plus Noise Ratio*), vazão e localização. Quanto maior a área e a fidelidade geométrica, maior o custo; modelos simplificados ou aprendidos podem complementar o ray tracing quando o caso de uso não exige uma reconstrução espacial completa [O-RAN nGRG, 2024a].

No representação da rede de núcleo de rede fixa ou móvel (*Core*) twin, as entidades são funções, sessões, interfaces, políticas, recursos computacionais e fluxos de sinalização. Casos do 3GPP incluem avaliação de tempestades de sinalização, configurações, operações de maior risco, políticas de automação e geração de dados. O gêmeo digital pode representar crescimento abrupto de requisições, indisponibilidade de funções, limites de capacidade e respostas de controle antes que mudanças sejam realizadas em produção [3GPP, 2025c, 3GPP, 2026a].

No transporte do gêmeo digital, os objetos dominantes são roteadores, enlaces, filas, caminhos, demandas, capacidades, SRLGs e políticas. As aplicações incluem planejamento de capacidade, análise de falhas, predição de atraso e perda, avaliação de rotas e engenharia de tráfego. O estudo de Almasan et al. utiliza um modelo baseado em GNN como NDT de desempenho: topologia, matriz de tráfego e roteamento são usados para prever atraso e alimentar um otimizador. Essa linha será aprofundada na família RouteNet [Almasan et al., 2022].

No gêmeo digital de borda e nuvem, o foco recai sobre aplicações, servidores, aceleradores, VNFs, CNFs, CPU, memória, armazenamento, enlaces e posicionamento. As decisões de *offloading*, escala, migração e localização de funções combinam recursos de rede e computação. Em um O-RAN em nuvem, o gêmeo digital também precisa representar o acoplamento entre funções virtuais e os hospedeiros físicos que sustentam

seu desempenho [O-RAN nGRG, 2024a, Lin et al., 2023].

Tabela 4.2. Escopos típicos de gêmeos digitais orientados a domínio. As escalas são indicativas e dependem do caso de uso.

Gêmeo digital	Entidades e estados dominantes	Aplicações típicas	Escala temporal indicativa
RAN	células, UEs, canal, mobilidade, interferência, energia	planejamento, handover, potência, testes e energy saving	ms a minutos
Núcleo de Rede (Core)	funções, sessões, políticas, signaling e compute	capacidade, falhas, configurações, assurance e scaling	segundos a horas
Transporte	enlaces, filas, caminhos, demandas, capacidades e SRLGs	CP, what-if, falhas, KPIs e TE	segundos a meses
Borda/nuvem	aplicações, hospedeiros, CPU/GPU, armazenamento e enlaces	alocação, transferência de processamento, dimensionamento e autorecuperação	ms a horas

4.2.5. NDTs de Serviço, Fatiamento e Colaboração Multidomínio

Um usuário não consome isoladamente a Rede de acesso rádio, o núcleo de rede móvel ou fixa ou transporte; ele consome um serviço. Por isso, a visão fim-a-fim não deve ser construída apenas pela soma de inventários. Um gêmeo digital de serviço (*service twin*) seleciona os recursos, dependências, funções, caminhos e KPIs relevantes para determinado serviço. Uma fatia de rede representada no gêmeo digital (*slice twin*) faz o mesmo para uma rede lógica com requisitos próprios, preservando o mapeamento entre componentes virtuais e recursos físicos compartilhados.

Esse gêmeo (*twin*) pode perguntar se a capacidade suporta uma nova demanda, que domínio limita o SLA, como uma falha altera a experiência ou se uma expansão apenas desloca o gargalo. A especificação técnica 3GPP TS 28.530 já exige que mudanças de capacidade de uma fatia de rede sejam decompostas em alterações de RAN e Core e em requisitos para o transporte. O NDT acrescenta a possibilidade de experimentar a mudança e avaliar suas consequências antes de executá-la. [3GPP, 2025a]

A composição pode assumir formas diferentes. Em uma abordagem monolítica, uma única plataforma modela todos os domínios. Ela oferece uma visão integrada, mas enfrenta escala, propriedade de dados, diversidade de fornecedores e sincronização. Em uma composição hierárquica, gêmeos de domínio fornecem abstrações a um gêmeo de serviço. Em uma colaboração *peer-to-peer*, cada gêmeo responde a consultas e compartilha capacidades, previsões ou restrições sem entregar toda a sua representação interna.

O Release 20 da especificação técnica 3GPP TS 28.561 formaliza a colaboração hierárquica e peer-to-peer entre NDTs, dentro ou entre domínios de gestão. O IRTF admite múltiplos NDTs por rede, centralizados ou distribuídos, e mapeamentos horizontais entre gêmeos. Os relatórios O-RAN propõem NDTs especializados para RAN, núcleo de rede, borda e nuvem. combinados por uma camada de orquestração. Essas fontes sustentam uma conclusão prudente: a composição multidomínio é necessária, mas ainda não existe uma solução universal para semântica, confiança, consistência e resolução de

conflitos [3GPP, 2026a, IRTF NMRG, 2026, O-RAN nGRG, 2024b].

Para um serviço imersivo de baixa latência, por exemplo, o RAN gêmeo pode estimar cobertura, associação e mobilidade; o transporte gêmeo, atraso, perda e capacidade residual; o gêmeo de núcleo de rede sessões, função de plano de usuário (*User Plane Function* – UPF) e políticas; o gêmeo de borda, alocação e recursos computacionais. O gêmeo de serviço distribui requisitos e agrega respostas. A decisão final precisa considerar tanto as previsões individuais quanto a idade, a confiança e o escopo de cada gêmeo digital.

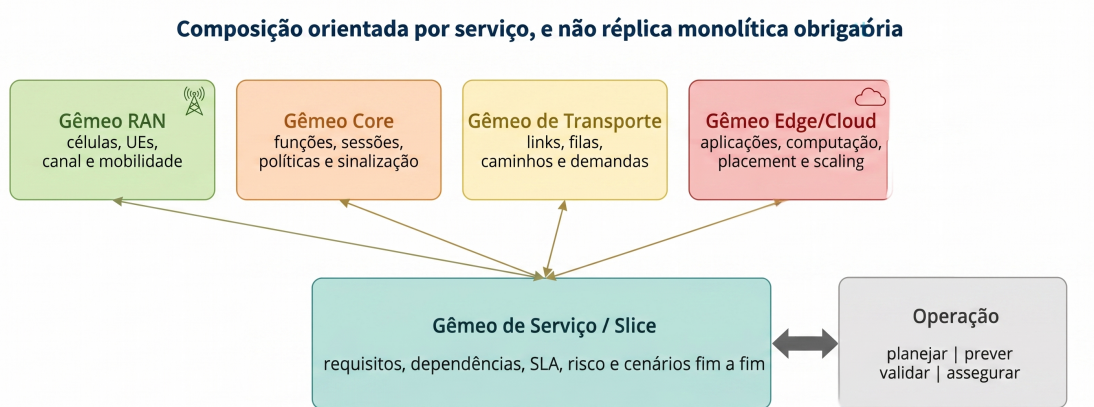


Figura 4.2. Composição de gêmeos digitais de domínio em um gêmeo digital orientado a serviço ou fatia de rede.

4.2.6. Casos de uso e papel de IA e GNNs

Os casos de uso podem ser agrupados em quatro classes:

- **Planejar:** inclui cobertura, implantação, capacidade, alocação, energia e fatias de rede.
- **Prever:** inclui demanda, mobilidade, falhas, KPIs e congestionamento.
- **Experimentar:** inclui “*what-if*”, configurações, injeção de falhas, testes de funções e geração de dados.
- **Operar:** inclui garantia de serviço, recomendação, otimização e ciclos fechados supervisionados.

Na RAN, o Grupo de Pesquisa de Próxima Geração O-RAN (*next Generation Research Group* – nGRG) mostra que o NDT pode fornecer *sandbox* para treinamento e validação de IA/ML, testar componentes e interfaces, gerar dados sintéticos, calibrar modelos de propagação e comparar políticas de energia. No 3GPP, os casos incluem tempestades de sinalização, configuração, operações de risco, verificação de funções de automação e geração de dados para eventos raros. Em redes IP, Almasan et al. mostram o gêmeo digital

como modelo substituto (*surrogate*) de desempenho em um processo iterativo de otimização [O-RAN nGRG, 2024b, O-RAN nGRG, 2024a, 3GPP, 2025c, Almasan et al., 2022].

A inteligência artificial não é requisito para que exista um NDT. Simulação, emulação, modelos analíticos, regras e conhecimento especialista podem compor sua camada de modelos. IA e GNNs tornam-se importantes quando o comportamento é difícil de representar por fórmulas, quando o espaço de cenários é grande ou quando se deseja acelerar uma simulação. Elas podem estimar o estado, prever a demanda, detectar anomalias, construir modelos substitutos e apoiar decisões [IRTF NMRG, 2026, 3GPP, 2025c, O-RAN nGRG, 2024a].

As GNNs são particularmente adequadas quando a saída depende de topologia e relações entre entidades. Roteadores, enlaces e caminhos formam grafos; células, usuários e interferências também; funções, serviços e recursos virtuais criam grafos heterogêneos. No NDT, a GNN ocupa a camada de modelos. Ela não substitui dados, mapeamento, gestão, interfaces, validação ou atuação [Almasan et al., 2022]. Essa distinção organiza os estudos de caso do restante do minicurso. A SGCRN é apresentada como modelo espaço-temporal de previsão; RouteNet e suas evoluções, como modelos de comportamento e desempenho; TEAL, como mecanismo de decisão de engenharia de tráfego. O NDT é a arquitetura que permite que esses modelos sejam alimentados, avaliados, combinados e utilizados em um ciclo operacional controlado.

Um Gêmeo Digital de Redes (*Network Digital Twin* – NDT) deve ser entendido por sua função operacional, não pelo grau de realismo visual nem por uma tecnologia específica. Ele delimita uma contraparte, organiza dados e modelos, mantém mapeamentos, expõe interfaces, gerencia execuções e fornece um ambiente no qual hipóteses e ações podem ser avaliadas sem expor diretamente a rede.

Em 5G/6G, gêmeos orientados a domínio permitem representar a RAN, o Core, o transporte e o edge com entidades e escalas apropriadas. gêmeos orientados a serviço ou slice coordenam essas perspectivas para verificar requisitos fim a fim. A composição evita exigir uma cópia monolítica de toda a operadora, mas introduz desafios de interoperabilidade, confiança e consistência que serão retomados no capítulo sobre limitações e tecnologias habilitadoras.

Resta compreender como representar computacionalmente as relações presentes nesses sistemas. Os elementos da rede não são observações independentes: caminhos compartilham enlaces, usuários interferem entre si, funções dependem de recursos e slices compartilham infraestrutura. O próximo capítulo apresenta as Graph Neural Networks como uma família de modelos concebidos para aprender sobre essas estruturas relacionais.

4.3. Fundamentos de Redes Neurais de Grafos

As GNNs introduzem um viés indutivo relacional: em vez de tratar cada medição como uma variável independente, aprendem representações a partir das entidades, de seus atributos e das relações que as conectam. A escolha do grafo determina quais dependências o modelo pode aprender. Uma GNN não começa pela escolha de uma arquitetura, mas pela definição da estrutura relacional do problema. O grafo determina o que pode ser

trocado por *message passing*; a arquitetura determina como essa informação será agregada; o treinamento determina a tarefa; a avaliação delimita onde o resultado pode ser utilizado.

A seção anterior mostrou que um NDT pode combinar modelos analíticos, simulação, emulação, otimização e aprendizado de máquina. A GNN ocupa uma posição específica nessa arquitetura: ela é um modelo capaz de processar dados estruturados como grafos. Não substitui a coleta, o mapeamento físico-digital, a gestão do gêmeo digital, as interfaces ou a validação operacional. Essa distinção é importante porque muitos fenômenos de rede não podem ser explicados por medições independentes. A utilização de um enlace depende das demandas que o atravessam; o atraso de um caminho depende dos recursos compartilhados ao longo da rota; a carga de uma célula depende dos usuários, da mobilidade e da interferência; uma fatia de rede depende do mapeamento entre funções virtuais e recursos físicos. Em todos esses casos, a relação entre as entidades é parte do problema.

As GNNs foram desenvolvidas para incorporar esse tipo de estrutura ao aprendizado. Elas compartilham parâmetros entre entidades semelhantes e atualizam suas representações por meio da troca de mensagens computacionais entre elementos conectados. Esse mecanismo, denominado *message passing*, é o eixo da seção e a base para compreender Rede Recorrente Convolutiva de Grafos Espaço-temporal (*Spatio-temporal Graph Convolutional Recurrent Network* – SGCRN), RouteNet e TEAL.

4.3.1. Grafos de Rede

Um grafo pode ser representado por $G = (V, E)$, em que V é o conjunto de entidades e E é o conjunto de relações. Cada entidade v pode possuir atributos x_v , como capacidade, carga, potência ou tipo; cada relação (u, v) pode possuir atributos e_{uv} , como banda, atraso, distância, ganho de canal ou forma de dependência. Atributos globais também podem representar contexto, como, por exemplo, horário, cenário de falha ou objetivo de operação. Battaglia et al. organizam essa representação em entidades, relações e regras de composição, constituindo um viés indutivo relacional para o aprendizado [Battaglia et al., 2018].

Não existe, porém, um único grafo natural para uma rede. A topologia física é apenas uma das representações possíveis. Roteadores e enlaces são adequados para conectividade, falhas e capacidade, mas não mostram diretamente como os caminhos competem por recursos. Em uma representação caminho-enlace, caminhos e enlaces formam conjuntos de entidades distintas, conectados quando um caminho utiliza um determinado enlace. Essa estrutura permite que demandas sejam agregadas nos recursos compartilhados e que o estado desses recursos retorne aos caminhos.

A forma do grafo também expressa hipóteses sobre o sistema. Um grafo direcionado diferencia origem e destino; um grafo ponderado atribui intensidade às relações; um multigrafo admite vários enlaces ou tipos de relação entre as mesmas entidades; um grafo bipartido separa conjuntos, como caminhos e enlaces; um grafo heterogêneo preserva tipos diferentes de nós e arestas. Essas escolhas funcionam como um viés indutivo: orientam o modelo sobre quais interações devem ser compartilhadas e quais diferenças precisam ser preservadas. A figura 4.3 mostra um exemplo de grafo de rede móvel com

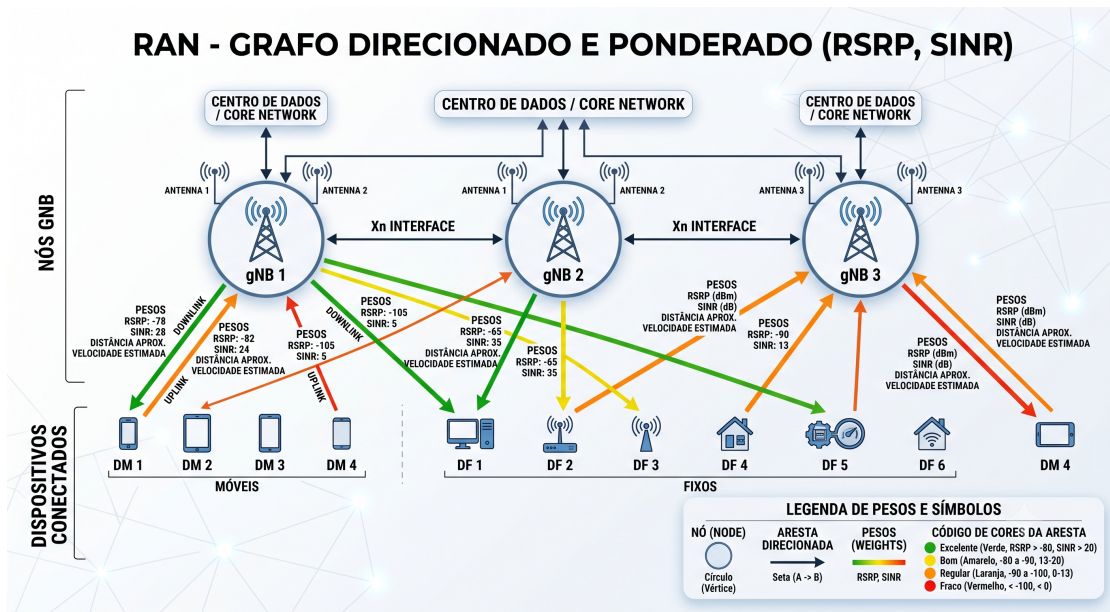


Figura 4.3. Exemplo de um grafo direcionado e ponderado de uma RAN.

as estações rádio base 5G (gNodeB) e dispositivos conectados móveis (DM) e fixos (DF), os nós do grafo. As arestas são representadas pela conexão da estação rádio base com o dispositivo. A ponderação das arestas tem pesos representando parâmetros de Potência do Sinal de Referência Recebido (*Reference Signal Received Power* — RSRP) e ruído (*Signal to Interference plus Noise Ratio* — SINR).

Na RAN, células, usuários e fontes de interferência podem constituir um grafo de associação ou de interferência. A relação pode representar serviço, proximidade ou ganho de canal, e seus atributos podem incluir RSRP, SINR, potência e disponibilidade de recursos. Em redes virtualizadas, aplicações, funções, servidores, enlaces e fatias de rede formam grafos heterogêneos, nos quais relações como hospedagem, encaminhamento, compartilhamento e dependência possuem significados distintos.

A dimensão temporal acrescenta outra camada. Um grafo de pontos de presença (*PoPs*), células ou pontos de borda (*edge sites*) pode permanecer relativamente estável enquanto seus atributos evoluem ao longo de uma série temporal; em outros casos, as próprias arestas mudam devido à mobilidade, falhas ou reconfiguração. Modelos espaço-temporais procuram aprender, simultaneamente, a dependência estrutural entre entidades e a evolução de seus estados. Uma mesma infraestrutura de rede pode ser representada por diferentes grafos de acordo com a modelagem de parâmetros da rede e os objetivos de resultado.

A representação também precisa refletir a granularidade da saída. No objetivo de aprendizado ou modelo de predição, denominado doravante tarefa, na tarefa por nó, o objetivo pode ser prever a carga de uma célula ou classificar a condição de uma função. Em uma tarefa por aresta, pode-se estimar a qualidade, a capacidade residual ou o risco de falha de um enlace. Um caminho ou fluxo pode possuir uma saída própria, como atraso, *jitter* ou perda. Finalmente, uma função de leitura/agregação global *readout* em

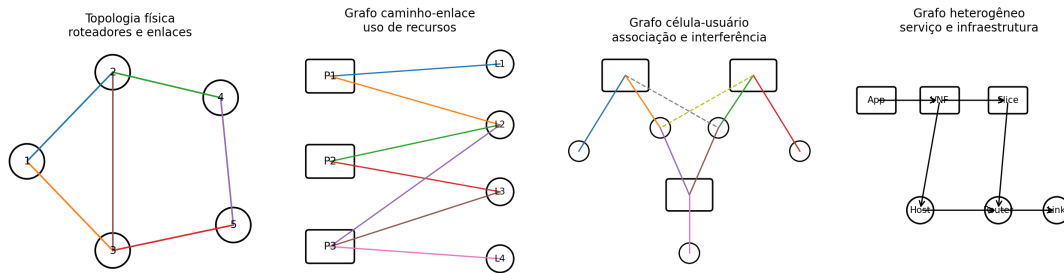


Figura 4.4. A mesma infraestrutura pode originar grafos distintos conforme o comportamento do dado estrutural e objetivo de aprendizado do modelo

GNNs é o componente responsável por resumir todas as informações dos nós individuais em um único vetor pode produzir uma estimativa para todo o grafo, como o consumo de energia, o risco ou o desempenho agregado. Escolher nós, relações e saídas é uma decisão de modelagem, não uma etapa automática. Essa decisão condiciona a informação disponível. Um grafo roteador-enlace pode ser suficiente para conectividade, mas não distingue duas classes que usam filas diferentes. Um grafo de tráfego pode representar demandas, mas não localizar congestionamento sem o mapeamento para recursos físicos. A representação mais detalhada nem sempre é a melhor: ela aumenta o custo, os requisitos de dados e a dificuldade de generalização. O princípio é representar apenas as entidades e as relações necessárias aos objetivos operacionais de rede. A figura 4.4 ilustra diferentes representações.

4.4. Funcionamento do *Message Passing*

O formalismo de *Message Passing Neural Networks* unifica diversas arquiteturas de GNN em duas fases gerais: troca iterativa de mensagens e geração da saída. Cada entidade inicia com um estado, ou (*embedding*), derivado de seus atributos observados. Para um enlace, esse estado pode incluir capacidade e utilização; para um caminho, demanda e classe de serviço; para uma célula, carga, potência e banda disponível [Gilmer et al., 2017].

Na equação a seguir, $\mathbf{h}_v^{(0)}$ é o estado inicial (ou *embedding* inicial) do nó/entidade v no instante 0 (antes de qualquer troca de mensagens). É representado como um vetor que sintetiza as informações daquela entidade no início do processo. v é o índice da entidade ou nó do grafo que está sendo analisado (por exemplo, um roteador, célula, enlace ou caminho). O sobrescrito (0) indica a iteração zero (passo inicial) da GNN, ou seja, a etapa prévia à aplicação das camadas de message passing. Já $\mathbf{x}_v$ é o vetor de atributos brutos observados da entidade v (como capacidade, utilização, demanda ou atraso de propagação). Em suma, a equação estabelece que, na camada inicial, o embedding do nó é simplesmente a própria entrada com as suas características originais.

$$\mathbf{h}_v^{(0)} = \mathbf{x}_v. \quad (1)$$

Na iteração k , uma função aprendida constrói a mensagem enviada pela entidade u à entidade v . A mensagem pode combinar o estado de origem, o estado de destino e os

atributos da relação. Ela não é um pacote real nem uma mensagem de protocolo: é um vetor computacional que resume informações relevantes para a tarefa.

$$\mathbf{m}_{u \rightarrow v}^{(k)} = \phi^{(k)}(\mathbf{h}_u^{(k)}, \mathbf{h}_v^{(k)}, \mathbf{e}_{uv}). \quad (2)$$

As mensagens recebidas por v são agregadas. Soma, média e máximo são exemplos simples; mecanismos de atenção aprendem pesos diferentes para cada relação. Como os vizinhos de um nó formam um conjunto sem ordem natural, a agregação precisa produzir o mesmo resultado independentemente da sequência nas quais as mensagens são apresentadas.

$$\bar{\mathbf{m}}_v^{(k)} = \text{AGG}\left(\left\{\mathbf{m}_{u \rightarrow v}^{(k)} : u \in \mathbf{N}(v)\right\}\right). \quad (3)$$

Em seguida, uma função de atualização combina o estado anterior com a mensagem agregada. Ela pode ser implementada por uma camada densa, uma unidade recorrente ou outra função diferenciável. O resultado é um novo *embedding* que incorpora o contexto local:

$$\mathbf{h}_v^{(k+1)} = \psi^{(k)}(\mathbf{h}_v^{(k)}, \bar{\mathbf{m}}_v^{(k)}). \quad (4)$$

Uma camada local permite que uma entidade dependa de seus vizinhos imediatos. Após K camadas, o campo receptivo teórico alcança entidades a até K saltos. Isso não significa que toda a informação distante será preservada: gargalos topológicos e vetores de dimensão fixa podem comprimir o contexto, como discutido adiante.

As mesmas funções são aplicadas a entidades de um mesmo tipo. Esse compartilhamento reduz o número de parâmetros e permite executar a arquitetura em grafos de tamanhos diferentes. A propriedade relevante é de permutação: quando a numeração dos nós é alterada, as saídas por entidade são renumeradas correspondentemente (equivariância) enquanto uma saída global pode permanecer inalterada (invariância). Essa propriedade não garante, por si só, precisão em qualquer topologia nova.

A saída depende da tarefa. Um decodificador pode atuar sobre o *embedding* de um nó, de uma aresta ou de uma entidade caminho. Ao se desejar uma propriedade global, uma função de *readout* agrega uma nova incorporação de *embeddings* finais. Portanto, o *message passing* não implica necessariamente reduzir toda a rede a um único vetor.

Considere um grafo caminho-enlace. Na primeira etapa, cada caminho envia aos enlaces informações sobre demanda, classe e sequência de uso. Cada enlace agrega a contribuição dos caminhos que o atravessam e atualiza um estado que representa a competição por capacidade. Na etapa seguinte, os enlaces retornam esse estado aos caminhos; cada caminho combina as condições dos recursos utilizados e produz uma estimativa fim-a-fim. Esse fluxo alternado não reproduz pacotes individualmente, mas aprende a interação estrutural entre demanda e recursos compartilhados.

Algumas arquiteturas repetem etapas distintas de *message passing* entre tipos de entidades. Em grafos heterogêneos, uma função pode tratar mensagens caminho-para-

enlace e outra, mensagens enlace-para-caminho. Essa especialização preserva a semântica das relações e será essencial para compreender RouteNet-Fermi, que adiciona filas à representação.

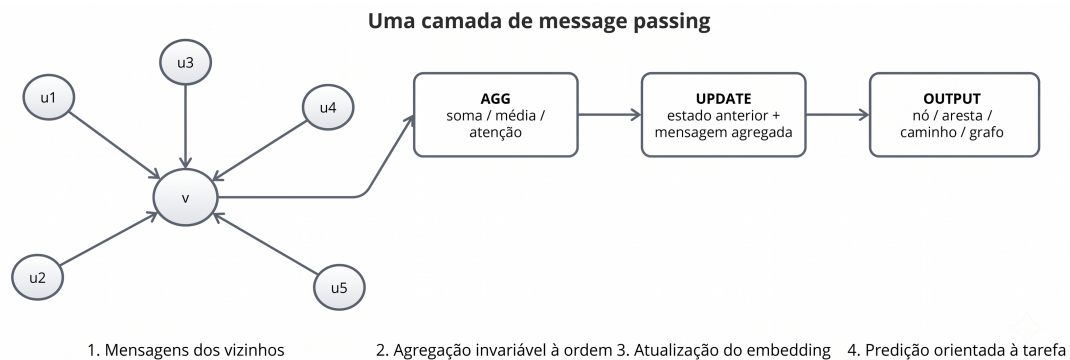


Figura 4.5. Etapas de uma camada de *message passing* e geração de saída orientada à tarefa.

4.5. Aprendizado e Avaliação da GNN

Os *embeddings* são representações latentes aprendidas para servir à tarefa do objetivo operacional de rede, como, por exemplo, maximizar a potência do sinal de referência recebido (RSRP). Seus componentes não correspondem necessariamente a métricas diretamente interpretáveis. Durante o treinamento, os parâmetros das funções de mensagem, agregação aprendida, atualização e decodificação são ajustados para reduzir uma função de perda. Essa perda pode ser o erro entre atraso previsto e observado, a diferença entre carga futura e real, ou a probabilidade atribuída à classe correta de uma falha.

Uma GNN possui 4 fases de um ciclo de vida: a primeira fase é o *treinamento*, momento em que o modelo analisa vários exemplos com respostas corretas para ajustar suas conexões internas e “aprender” os padrões da rede. A segunda fase é a *validação*, onde o modelo é testado enquanto ainda está sendo ajustado. Nessa fase, decide-se a configuração inicial do modelo, como a sua profundidade (quantas camadas utilizar) e a dimensão da sua memória (dimensão dos “embeddings”). Na terceira fase, a de *Testes*, o desempenho do modelo é avaliado em dados totalmente inéditos, que não foram utilizados nem no estudo nem nas configurações iniciais. Na quarta e última fase, a inferência com o modelo já está pronta e trabalhando no mundo real. Ele recebe novos cenários da rede e gera previsões instantâneas. A GNN não apenas aplica o que aprendeu; ela não continua aprendendo a cada nova execução.

O aprendizado da GNN pode ser realizado de duas formas: “Aprendizado Transutivo” (O modelo conhece o mapa inteiro), em que a GNN já conhece toda a estrutura do mapa da rede desde o início, mas apenas algumas “respostas” ou “etiquetas” (rótulos) estão escondidas. Ela aprende deduzindo as respostas que faltam dentro desse mesmo mapa conhecido. A outra forma é “Aprendizado Indutivo” (o modelo aprende regras gerais), em que o modelo aprende a “lógica” de como a informação flui em um grafo. Com isso, ele consegue ser aplicado em novos nós ou até em mapas (topologias) totalmente novos

que ele nunca viu antes [Hamilton, 2020].

A divisão dos dados também deve respeitar a estrutura do problema. Separar aleatoriamente amostras temporais pode vazar informação futura; separar apenas matrizes de tráfego não mede generalização topológica; testar em grafos maiores exige controlar se os padrões locais e a distribuição de graus permanecem comparáveis. Métricas médias precisam ser complementadas por percentis, erros por classe e análise de cenários críticos, pois uma pequena média pode esconder falhas nos casos que determinam o acordo de nível de serviço (*Service Level Agreement* – SLA).

4.5.1. Variantes do Problema

As variantes das GNN não formam uma escala linear de evolução. Cada uma introduz um mecanismo adequado a determinada estrutura de dados ou dependência:

- **Rede Convolutacional de Grafo (*Graph Convolutional Network* – GCN) e Rede Neural de Passagem de Mensagem (*Message Passing Neural Network* – MPNN).** A GCN de Kipf e Welling [Kipf e Welling, 2017] combina atributos dos nós com a vizinhança por uma agregação normalizada e eficiente com custo linear no número de arestas. Ela oferece uma introdução compacta às convoluções em grafos. A MPNN é mais geral: permite construir mensagens a partir de atributos ricos de nós e arestas e definir atualizações orientadas ao domínio. Essa flexibilidade prepara a representação de caminhos, enlaces e filas utilizada pela família RouteNet.
- **Rede de Atenção em Grafos (*Graph Attention Networks* – GAT).** A GAT aprende coeficientes diferentes para os vizinhos conectados) [Veličković et al., 2018], por meio de atenção mascarada local (*Masked Attention*) e múltiplas cabeças de atenção (*Attention Head*). Em RAN, isso permite ponderar relações de interferência com intensidades distintas; em fatiamento de redes, pode atribuir importância desigual a recursos físicos. Os coeficientes, porém, descrevem ponderações internas e não constituem explicação causal. Atenção local e global responde a problemas distintos. Na GAT, um nó considera apenas os vizinhos definidos pela adjacência; o custo é proporcional ao número de relações observadas. Em um transformador de grafos *Graph Transformer* global, pares distantes podem interagir diretamente, mas a topologia precisa ser reintroduzida por codificações posicionais, estruturais ou por um ramo de *message passing*. Para interferência entre células próximas, a atenção local pode bastar; para dependências fim a fim, uma combinação local-global pode ser útil.
- **GNNs espaço-temporais:** Nessa família, uma operação no grafo modela relações espaciais ou estruturais, enquanto recorrência, convolução temporal ou atenção modela a evolução no tempo. As Redes Convolucionais de Grafos Espaço-Temporais (*Spatio-Temporal Graph Convolutional Networks* – STGCN) [Yu et al., 2018] substituem unidades recorrentes por blocos convolucionais temporais e foram propostas para tráfego rodoviário, mas seu princípio é transferível para séries por ponto de presença (*Point of Presence* – PoP), célula ou edge site [Yu et al., 2018]. A SG-CRN é enquadrada nessa família. A dimensão temporal pode ser tratada de várias

formas. Unidades recorrentes, como Unidade Recorrente com Portas (*Gated Recurrent Unit* – GRU) e Memória de Longo e Curto Prazo (*Long Short-Term Memory* – LSTM), mantêm um estado que evolui ao longo dos passos; convoluções temporais processam janelas em paralelo; mecanismos de atenção selecionam instantes relevantes; *Transformers* temporais procuram dependências mais longas. A escolha depende do horizonte, da frequência de amostragem, da necessidade de inferência e de como o erro será propagado em previsões multietapa.

- **GNNs heterogêneas e hierárquicas:** Quando usuários, células, funções, servidores e fatias de rede possuem tipos e relações distintos, modelos heterogêneos podem utilizar parâmetros dependentes da meta-relação. O *Heterogeneous Graph Transformer* é um exemplo que combina heterogeneidade, atenção e codificação temporal [Hu et al., 2020]. Modelos hierárquicos, por sua vez, agregam entidades em múltiplas escalas, como usuário, célula, região e domínio. São opções naturais para NDTs multidomínio, mas não obrigatórias: múltiplos modelos ou gêmeos de domínio também podem ser compostos. A Figura 4.6 exemplifica uma GNN hierárquica e heterogênea para o caso de uso de O-RAN com fatiamento de rede.
- **GNNs espectrais:** Abordagens espectrais interpretam atributos como sinais no grafo e utilizam propriedades do Laplaciano para construir filtros. Elas fornecem uma base importante para processamento de sinais em grafos e para modelos que incorporem um *Graph Fourier Transform*. Contudo, operações dependentes da base espectral de uma topologia específica podem ter menor portabilidade para grafos distintos.
- **Modelos Transformer para Grafos (Graph Transformers).** A atenção global cria caminhos diretos de interação entre entidades distantes, mas perde a estrutura implícita da adjacência se nenhuma informação adicional for fornecida. Arquiteturas como GraphGPS combinam codificações posicionais ou estruturais, *message passing* local e atenção global [Rampášek et al., 2022]. Essa combinação pode aliviar dependências de longo alcance, ao custo de maior processamento e complexidade de modelagem. Não é um substituto universalmente superior às MPNNs.

A GNN combinada com aprendizado por reforço codifica o estado relacional da rede e o agente escolhe ações. A composição é útil quando decisões alteram o estado futuro, como alocação de tráfego, energia ou recursos. Ainda assim, representação, política e viabilidade são problemas separados: uma boa codificação não corrige uma recompensa mal especificada, nem garante que a ação respeite restrições.

4.5.2. Limitações Práticas das GNNs

As GNNs apresentam algumas limitações práticas de implementação, tais como:

- **Profundidade e excesso de suavização (*Over-smoothing*):** Aumentar o número de camadas amplia o campo receptivo teórico, mas agregações repetidas podem tornar *embeddings* de entidades diferentes excessivamente semelhantes. Conexões residuais, normalização, mecanismos recorrentes e combinação de camadas intermediárias podem reduzir o efeito. Porém, a profundidade adequada depende da

ARQUITETURA DE GNN HIERÁRQUICA E HETEROGÊNEA - FATIAMENTO DE REDE O-RAN

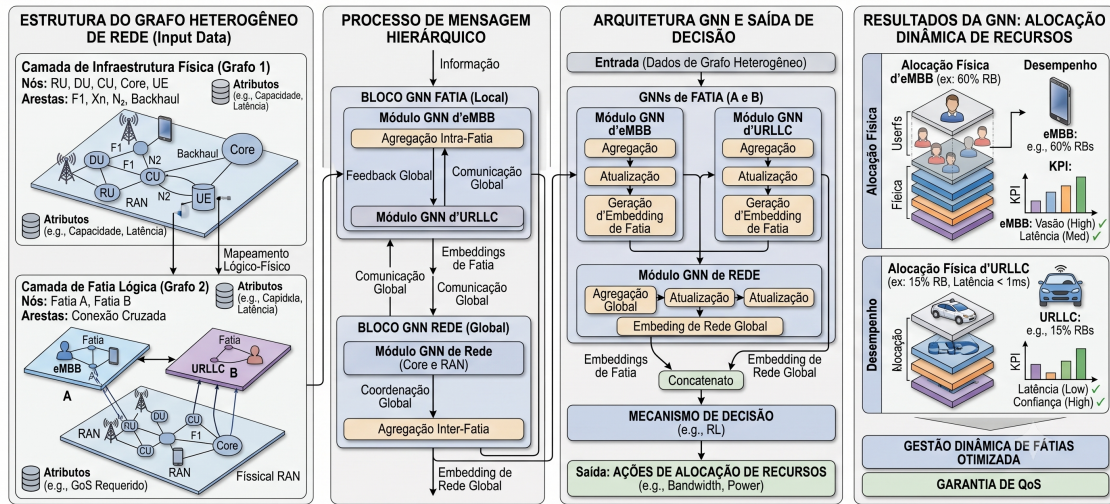


Figura 4.6. Exemplo GNN hierárquica e heterogênea representando uma O-RAN com fatiamento de rede (*slicing*).

tarefa e da arquitetura. O problema não é apenas visualmente “igualar” nós. Se células com papéis diferentes recebem representações quase idênticas, o classificador perde a capacidade de distinguir *hotspot*, borda de cobertura e célula ociosa. No transporte, os enlaces de núcleo e de acesso podem tornar-se indistinguíveis, apesar de terem impactos operacionais distintos. A mitigação precisa preservar a identidade e, ao mesmo tempo, permitir a troca de contexto.

- **Supercompressão (*Over-squashing*):** Quando o número de entidades no campo receptivo cresce rapidamente, um grande volume de informação precisa atravessar poucas relações e ser comprimido em vetores de dimensão fixa. Alon e Yahav identificam esse gargalo como uma causa para a dificuldade de propagar sinais de longo alcance [Alon e Yahav, 2021]. Em redes, o fenômeno pode afetar caminhos longos, propagação de falhas e dependências fim-a-fim. Atenção global, alterações na estrutura do grafo (*rewiring*) e representações hierárquicas podem aliviar o problema, mas introduzem custo ou alteram a estrutura utilizada pelo modelo. Um exemplo é a predição de atraso para um caminho longo atravessado por muitos fluxos concorrentes. O estado do caminho precisa refletir condições locais distribuídas, mas a informação pode ser comprimida repetidamente antes de chegar ao decodificador. Aumentar apenas a dimensão do *embedding* nem sempre resolve o gargalo, porque a limitação também depende da geometria do grafo e das arestas por onde a informação flui.
- **Escala e dinâmica:** Grafos com milhões de entidades não podem, em geral, ser treinados integralmente em uma única memória. Amostragem de vizinhos, *mini-batches*, decomposição e modelos distribuídos reduzem o custo, mas podem omitir relações críticas. Mudanças de topologia também exigem cautela: métodos espaci-

Tabela 4.3. Variantes de GNN relevantes e funções que desempenham nos estudos de caso.

Família	Mecanismo central	Aplicação no minicurso	Cautela principal
GCN / MPNN	agregação local; mensagens com atributos	base de RouteNet e modelagem de desempenho	contexto distante pode ser comprimido
GAT	atenção aprendida entre vizinhos	interferência, slicing e relações desiguais	atenção não equivale a explicação
ST-GNN	grafo + dinâmica temporal	forecasting e SGCRN	horizonte, drift e mudança de topologia
Heterogênea	parâmetros por tipos e relações	RAN-Core-edge-slice	mais dados e complexidade semântica
Hierárquica	agregação em múltiplas escalas	célula-região-domínio	pooling pode perder informação
Espectral	filtros baseados no Laplaciano	GFT-MPNN e sinais no grafo	dependência da estrutura espectral
<i>GraphTransformer</i>	estrutura local + atenção global	dependências fim a fim	custo e codificação estrutural
GNN + RL	GNN codifica estado; agente escolhe ação	TEAL e otimização	recompensa e restrições não são triviais

ais indutivos podem aceitar novas estruturas, enquanto modelos fortemente ligados à base espectral ou a identificadores específicos tendem a ser menos transferíveis. A decomposição deve seguir o fenômeno, não apenas limites computacionais. Cortar um grafo por região pode eliminar um caminho inter-regional; amostrar vizinhos em um *hub* pode omitir justamente a demanda responsável pela violação. Estratégias de amostragem precisam preservar classes raras, caminhos críticos e relações de serviço. Em operação, também é preciso distinguir inferência incremental de retreinamento completo.

- **Dados fora da distribuição de treinamento (*Out Of Distribution* – OOD):** Um modelo treinado em determinadas topologias, cargas, políticas e ambientes pode perder precisão quando essas distribuições mudam. Dados sintéticos e NDTs ampliam a cobertura de cenários, mas não eliminam lacunas. O domínio de validade, a incerteza e o *drift* de dados precisam acompanhar a inferência.
- **Mudanças fora da distribuição (*Out of Distribution* – OOD):** podem ocorrer sem alteração da topologia como novos padrões de aplicação, políticas de fila, hardware, versões de software ou perfis de mobilidade já modificam a relação entre entrada e saída. Detectar que a amostra está fora do domínio de treinamento é tão importante quanto produzir a previsão. Em aplicações críticas, a resposta adequada pode ser recusar a recomendação, aumentar a margem de segurança ou recorrer a um modelo determinístico.
- **Explicabilidade, causalidade e restrições:** As GNNs aprendem dependências estatísticas; isso não prova causalidade. Pesos de atenção ou métodos de explicação

podem indicar fatores influentes, sem fornecer automaticamente uma justificativa operacional. Da mesma forma, uma saída neural não garante capacidade, política ou segurança. Restrições fortes (*Hard constraints*) precisam ser incorporadas por parametrização, projeção, otimização restrita, Método dos Multiplicadores de Direção Alternada (*Alternating Direction Method of Multipliers* – ADMM), uso de “solvers” tradicionais (*solver-in-the-loop*) ou validação externa.

4.5.3. Mapa para os Estudos de Caso

A escolha de uma GNN começa pela pergunta operacional: quais entidades existem, que relações condicionam a saída, como o estado evolui e que tipo de resultado será produzido. Não há arquitetura universal. Predição de carga requer representar espaço e tempo; predição de desempenho exige capturar compartilhamento de enlaces, caminhos e filas; decisões de engenharia de tráfego acrescentam ações, objetivos e restrições.

Quatro perguntas ajudam a disciplinar o projeto. Primeiro, qual variável precisa ser prevista ou decidida. A segunda, que entidades e relações carregam a informação necessária. A terceira, em que escala temporal e espacial o fenômeno ocorre. Finalmente, em quarto, em quais mudanças o modelo deverá generalizar. Somente depois dessas respostas faz sentido escolher uma GCN, MPNN, GAT, ST-GNN, modelo heterogêneo ou *Graph Transformer*.

A validação deve acompanhar a escolha. Um modelo de atraso precisa ser testado em diferentes cargas, caminhos e topologias; uma predição, em horizontes e eventos distintos; um modelo de RAN, em ambientes e condições de canal variados. O melhor resultado médio não é suficiente quando o uso operacional exige confiança, limites de erro e comportamento previsível nos extremos.

4.5.4. Exemplo: Indicadores-chave de Desempenho Representado por Grafos

Considere a previsão de atraso fim-a-fim para um conjunto de demandas IP. O indicador-chave de desempenho (KPI) é associado ao caminho, mas resulta da interação entre tráfego, enlaces, capacidade, filas e políticas. Representar apenas roteadores e enlaces preserva a topologia, porém não explicita que vários caminhos compartilham os mesmos recursos. Um grafo caminho-enlace torna essa dependência direta: cada caminho se conecta aos enlaces que utiliza.

Os atributos iniciais precisam separar as informações observadas do contexto. Caminhos podem carregar demanda, classe, origem, destino e comprimento; enlaces, capacidade, utilização, atraso de propagação e tipo; filas, disciplina, prioridade e tamanho. Incluir mais atributos não é automaticamente melhor: cada variável precisa estar disponível na inferência e permanecer semanticamente estável entre treinamento e operação.

O *message passing* pode alternar duas etapas. Primeiro, caminhos informam sua demanda aos enlaces. Os enlaces agregam as contribuições, atualizando uma representação da competição por capacidade. Depois, os enlaces retornam aos caminhos estados que refletem congestionamento e disponibilidade. Se filas forem entidades próprias, uma etapa adicional representa a associação entre classe, fila e enlace. Um decodificador aplicado ao *embedding* do caminho produz atraso, *jitter* ou perda.

Tabela 4.4. Sequência mínima para projetar e avaliar uma GNN aplicada a redes.

Etapa	Pergunta	Decisão de modelagem	Risco a testar
1. Tarefa	qual KPI ou ação?	saída por nó, enlace, caminho ou grafo	rótulo incompatível com o uso
2. Representação	quem influencia quem?	entidades, relações e atributos	grafo omite dependência crítica
3. Arquitetura	local, temporal ou global?	MPNN, GAT, ST-GNN, heterogênea ou híbrida	complexidade sem ganho útil
4. Generalização	o que mudará em produção?	split por tempo, topologia, escala e domínio	vazamento ou teste insuficiente
5. Operação	como confiar e restringir?	incerteza, OOD, solver e validação	ação inviável ou não explicada

O conjunto de treinamento deve conter diversidade de cargas, roteamentos, políticas e topologias compatível com o uso pretendido. A validação precisa separar novas matrizes na mesma topologia de novas topologias. Também deve examinar caminhos longos, enlaces próximos da saturação e classes minoritárias, pois esses cenários concentram risco de *over-squashing*, fora da distribuição de treinamento (*Out Of Distribution* – OOD) e erro operacional.

Finalmente, a previsão precisa chegar acompanhada de domínio de validade e mecanismo de verificação. O NDT pode comparar a estimativa com simulação, medição ou regras analíticas, recusar cenários fora da distribuição e impedir que uma decisão seja tratada como viável apenas porque o erro médio do modelo é baixo. Esse exemplo sintetiza o processo completo: definir a variável, escolher o grafo, construir as mensagens, treinar, testar generalização e integrar a inferência com validação.

A próxima seção aplica essa base em três etapas complementares. A SGRN utiliza estrutura espacial e dinâmica temporal para prever a demanda. A família RouteNet utiliza *message passing* entre entidades da rede para estimar o desempenho em cenários *what-if*. O TEAL combina uma representação centrada em fluxos com aprendizado por reforço e refinamento de viabilidade para calcular alocações candidatas. A GNN, em cada caso, ocupa uma função diferente no *Network Digital Twin*.

4.6. Aplicações de GNNs em NDTs para Redes 5G/6G

Os requisitos associados ao 5G-Advanced e às futuras redes 6G não se concentram em um único domínio. Eles surgem da combinação entre RAN dinâmica, funções virtualizadas do núcleo de rede fixa e/ou móvel (*Core*), aplicações distribuídas de computação de borda *edge*, fatiamento de rede (*network slicing*) e serviços com perfis muito distintos de capacidade, latência, confiabilidade e densidade. Entretanto, a experiência fim-a-fim depende também do transporte IP que interliga esses componentes. É nessa infraestrutura compartilhada que os requisitos dos serviços se convertem em volumes origem-destino, ocupação de enlaces e filas, caminhos, atraso, *jitter*, perda e risco de violação de SLA.

Os requisitos percebidos pelo usuário são fim a fim, mas os recursos que os sus-

tentam pertencem a domínios diferentes. Na RAN, o desempenho depende de cobertura, interferência, mobilidade, alocação de espectro, potência e carga das células. No núcleo de redes, depende da seleção e do dimensionamento de funções, do tratamento das sessões, das políticas e da sinalização. Nas bordas, o posicionamento da aplicação e a disponibilidade de CPU, memória e armazenamento passam a compartilhar o orçamento de latência com a própria rede. No transporte, caminhos, capacidades, filas, classes de serviço e mecanismos de proteção precisam prover conectividade coerente com o serviço.

O fatiamento de rede torna essa interdependência explícita. O 3GPP define uma fatia como uma rede lógica apoiada por funções e recursos físicos ou virtualizados. Sua gestão deve traduzir mudanças de capacidade em alterações correspondentes nas sub-redes de RAN e núcleo de rede, bem como em requisitos para a rede de transporte. A especificação técnica 3GPP TS 28.530 determina, inclusive, que o sistema de gestão seja capaz de calcular variações de capacidade nas sub-redes e comunicar ao domínio de transporte os requisitos associados à mudança [3GPP, 2025a]. Portanto, uma fatia de rede não elimina o planejamento de capacidade; ela o torna multidomínio e orientado por perfis de serviço, conforme mostrado na Figura 4.7. O mesmo ocorre com a latência. Colocar uma aplicação mais próxima do usuário por meio de MEC pode reduzir o percurso até o processamento, mas não remove os atrasos da interface aérea, da (*gNodeB* – *gNB*), da rede de transporte, da Função do Plano de Usuário (*User Plane Function* – *UPF*) e da própria aplicação. O ETSI GR MEC 024 mostra que a garantia de latência exige coordenação entre o fatiamento e a infraestrutura de borda. Evoluções posteriores do 3GPP tornam a seleção de *UPF* e de servidores de aplicação sensível ao atraso do caminho, aproximando decisões de posicionamento, roteamento e gestão de sessão [ETSI, 2019a, 3GPP, 2026b].

Arquitetura Fim a Fim com Fatiamento de Rede

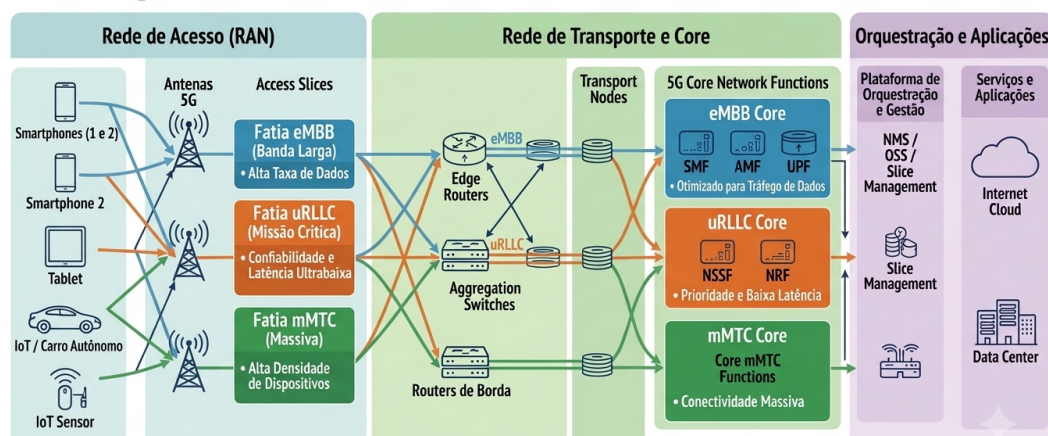


Figura 4.7. Arquitetura de Rede Multidomínio do 5G-Advanced e 6G.

Redes não terrestres ampliam a heterogeneidade. Diferentes órbitas, cobertura descontínua, enlaces com características variáveis e mecanismos de *store-and-forward* quebram a hipótese de conectividade uniforme. No Release 18, por exemplo, o 3GPP introduz controle de QoS consciente da categoria de *backhaul* por satélite e reconhece que o

atraso e a largura de banda podem variar ao longo do tempo. Em vez de uma rede com um único regime operacional, o planejamento passa a lidar com múltiplos regimes, cada qual com propriedades de propagação, disponibilidade e capacidade diferentes [3GPP, 2025b]. A tabela 4.5 exemplificam desafios em alguns domínios de rede, bem como variáveis de engenharia impactadas e consequência na operação das redes.

Tabela 4.5. Exemplos de desafios multidomínio e sua tradução em variáveis de engenharia.

Desafio	Domínios	Variáveis de engenharia	Consequência operacional
Latência e confiabilidade fim-a-fim	RAN, transporte, núcleo de rede e borda	PDB, <i>jitter</i> , perda, disponibilidade	Orçamento precisa ser decomposto e monitorado entre domínios.
Mobilidade e distribuição espacial	RAN, borda e transporte	Carga por célula/PoP, <i>handover</i> , ingressos	Demanda desloca-se e altera caminhos e locais de processamento.
Fatiamento de rede e elasticidade	RAN, Núcleo de Rede transporte e nuvem	Capacidade, isolamento, perfil de serviço	Recursos precisam ser criados, ajustados e verificados sem degradar outras fatias.
Multiacesso e NTN	RAN, Núcleo de Rede e transporte	Atraso de propagação, disponibilidade, direcionamento de tráfego	Caminhos com propriedades distintas exigem seleção e adaptação contínuas.
Energia e sustentabilidade	RAN, Borda e nuvem	Consumo, capacidade ativa, desempenho	Economia de energia passa a competir com cobertura, capacidade e SLA.
Falhas e eventos abruptos	Todos	Capacidade residual, tempo de recuperação, risco	A validade da decisão depende da rapidez na detecção, análise e atuação.

A mobilidade, o reposicionamento de funções de rede, a seleção de servidores de borda e as mudanças na origem do conteúdo alteram continuamente a distribuição espacial do tráfego. Ao mesmo tempo, fatias de rede (*slices*) com requisitos diferentes competem por recursos físicos comuns, e eventos de falha ou concentração de demanda podem deslocar rapidamente os gargalos. Por essa razão, a previsão de demanda, planejamento de capacidade e engenharia de tráfego no transporte IP constituem um recorte técnico particularmente útil: permitem observar, em uma mesma infraestrutura, a cadeia que vai da antecipação da demanda à estimativa de desempenho e à seleção de uma ação de engenharia.

O panorama multidomínio apresentado a seguir preserva a amplitude do problema: exemplos em RAN, *slicing*, Core e *edge* mostram que a representação em grafos pode apoiar tarefas distintas, como gerenciamento de recursos de rádio, classificação de falhas e alocações (*placement*). O aprofundamento posterior no transporte IP não pretende reduzir 5G/6G a esse domínio. Ele fornece uma ligação técnica no qual é possível comparar, com maior precisão, três funções complementares de um NDT: prever a evolução da demanda, estimar as consequências de uma configuração e produzir uma alocação candidata sob restrições, conforme mostrado na Tabela 4.6. O panorama multidomínio apresentado a seguir

Tabela 4.6. Efeitos no Transporte IP a partir dos Requisitos do 5G Avançado e 6G.

Cenário ou requisito	Efeito no transporte IP	Resposta aprofundada discutida na seção
eMBB e comunicações imersivas	Crescimento de volume, concentração espacial e mudanças nos principais pares Origem-Destino (OD)	Previsão de Demanda (<i>Forecasting</i>) e Planejamento de Capacidade (<i>Capacity Planning</i>)
URLLC e serviços críticos	Restrições de atraso, <i>jitter</i> , perda, proteção e rápida recomposição	Modelagem <i>what-if</i> e Engenharia de Tráfego (<i>Traffic Engineering – TE</i>)
mMTC e comunicação massiva	Grande número de fluxos, rajadas e maior incerteza sobre a demanda agregada	Observabilidade e Previsão de demanda (<i>forecasting</i>)
Fatiamento de rede (<i>Network slicing</i>)	Capacidade compartilhada entre serviços com SLAs e prioridades diferentes	Modelagem de desempenho e TE restrita
<i>Edge/MEC</i> , mobilidade e Core virtualizado	Mudança dinâmica de origens, destinos, cadeias de serviço (<i>service chains</i>) e caminhos	Previsão Origem-Destino (OD), avaliação de cenários e TE

preserva a amplitude do problema: exemplos em RAN, *slicing*, Core e *edge* mostram que a representação em grafos pode apoiar tarefas distintas, como gerenciamento de recursos de rádio, classificação de falhas e alocações (*placement*). O aprofundamento posterior no transporte IP não pretende reduzir 5G/6G a esse domínio. Ele fornece uma ligação técnica na qual é possível comparar, com maior precisão, três funções complementares de um NDT: prever a evolução da demanda, estimar as consequências de uma configuração e produzir uma alocação candidata sob restrições. Nesse recorte, a Rede Neural Recorrente Convolucional Espaço-Temporal em Grafo (*Spatiotemporal Graph Convolutional Recurrent Network – SGCRN*) antecipa a matriz origem-destino observada; a família RouteNet estima o comportamento e os KPIs da rede sob determinada demanda e configuração; e o TEAL calcula as razões de divisão de fluxos de tráfego (*split ratios*) candidatos à engenharia de tráfego. O NDT permanece responsável por organizar dados e cenários, coordenar os modelos, validar restrições e riscos e controlar qualquer atuação sobre a rede física.

4.6.1. Modelos Especializados em um NDT

Um NDT precisa desempenhar funções distintas ao longo do ciclo de planejamento e operação. Ele precisa representar o estado atual da rede, antecipar sua evolução, avaliar cenários alternativos e apoiar a seleção de ações. Essas funções não são necessariamente executadas por um único modelo. Em uma arquitetura modular, diferentes modelos podem ser combinados conforme o caso de uso, o domínio da rede, a granula-

ridade exigida e o grau de fidelidade necessário. As GNNs entram nesse contexto como motores analíticos especializados, e não como substitutos da arquitetura completa do gêmeo digital.

Nesta seção, três famílias ilustram papéis complementares. A Rede Neural Recorrente Convolucional Espaço-Temporal em Grafo (*Spatiotemporal Graph Convolutional Recurrent Network* – SGCRN) estima a matriz de tráfego observada no próximo intervalo; a família RouteNet recebe a topologia, a demanda e a configuração para estimar consequências de desempenho; e o TEAL calcula uma alocação candidata de tráfego, expressa por (*split ratios*). Essa separação evita confundir a previsão de demanda, a modelagem de comportamento e a decisão de engenharia. Ela também não implica uma sequência de etapas obrigatória e linear: conforme a aplicação, o NDT pode acionar apenas um desses modelos, executá-los em ordens diferentes ou combiná-los com simuladores, *solvers* e regras operacionais.

O NDT permanece responsável por tarefas que os modelos não resolvem isoladamente: aquisição e sincronização de dados, controle de versões, composição entre domínios, validação de restrições, análise de risco, autorização, atuação e *rollback*. Dessa forma, a contribuição das GNNs deve ser vista como parte de uma arquitetura governada. Antes do aprofundamento em *forecasting*, RouteNet e TEAL, convém mostrar que o mesmo princípio relacional se aplica a problemas muito distintos em redes 5G e 6G, da interferência de rádio à classificação de falhas no núcleo de rede fixa/móvel (*Core*) e à alocação de tarefas nas bordas. A figura 4.8 como os modelos SGCRN, TEAL e RouteNet se complementam na representação de um Gêmeo Digital de Redes (NDT).

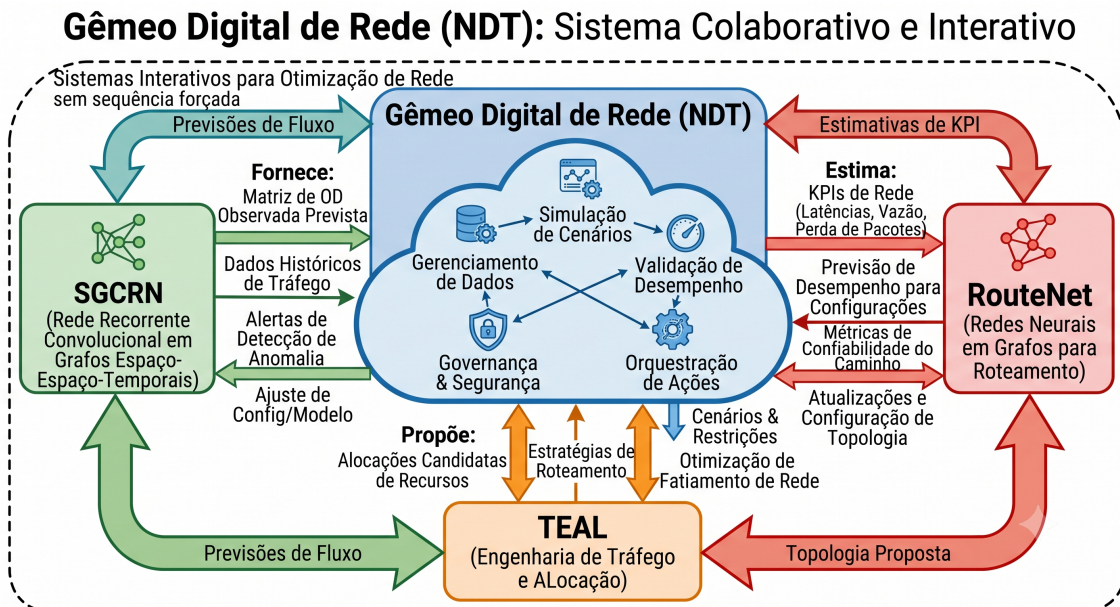


Figura 4.8. Papéis complementares de SGCRN, RouteNet e TEAL em um NDT.

4.6.2. Panorama Multidomínio de Aplicações de GNNs em Redes 5G/6G

O significado de um grafo depende do problema de engenharia. Em transporte IP, nós e arestas podem representar pontos de presença (*Point of Presence* – PoPs) e enlaces físicos; em rádio, podem representar pares transceptor-receptor e relações de interferência, em fatiamento de redes (*slicing*), estações-base e vizinhança geográfica; no Core, componentes funcionais e dependências, no *edge*, tarefas, ingressos e recursos computacionais. Portanto, “aplicar uma GNN à rede” não descreve uma técnica única. A escolha das entidades, das relações e da tarefa de aprendizado determina que informações o modelo pode capturar e que tipo de saída pode produzir.

Alguns exemplos de trabalhos relacionados de GNNs em Redes 5G/6G são listadas a seguir:

RAN (Interferência e Gerenciamento de Recursos): Um dos exemplos mais diretos é o gerenciamento de recursos de rádio. Na Rede Convolucional em Grafos de Canal Sem Fio (*Wireless Channel Graph Convolutional Network* – WCGCN), cada nó representa um par transmissor-receptor, enquanto as arestas representam relações de comunicação e interferência conforme mostrado na figura 4.3. Os atributos incluem o canal direto, o peso do usuário, o ruído e, quando disponíveis, as informações de estado do canal dos enlaces interferentes. A arquitetura é treinada de forma não supervisionada, usando a própria utilidade de comunicação como função-objetivo, e é avaliada em tarefas de controle de potência, maximização da taxa ponderada e conformação do feixe de rádio (*beamforming*). Não se trata, portanto, de um modelo para associação usuário-célula ou para escalonamento fino, que permanecem fora do escopo experimental do trabalho [Shen et al., 2021].

O resultado mais expressivo em termos de escala foi obtido no *beamforming*: a WCGCN processou 1.000 pares transmissor-receptor em aproximadamente 6 ms. O compartilhamento de parâmetros e a equivariância a permutações permitiram treinar o modelo com 50 pares e aplicá-lo a redes maiores. Essa transferência, contudo, não deve ser confundida com generalização irrestrita. Quando a densidade se afastou substancialmente das condições de treinamento, o desempenho se degradou, chegando a uma perda de aproximadamente 5,7% no cenário mais denso avaliado. Além disso, a execução distribuída em equipamentos de rádio reais não foi demonstrada no estudo.

Uma variação voltada a subredes industriais 6G, denominada controle de potência baseado em Redes Neurais de Grafos (*Power Control Graph Neural Network with heterogeneous input and Distributed execution* – PCGNN-hD), reduz a necessidade de coletar a matriz completa de informações de estado dos canais (*Channel State Information* – CSI) durante a inferência. Cada subrede utiliza o ganho do enlace desejado e as distâncias geométricas para representar os interferentes; o *full CSI* permanece necessário no treinamento para calcular a função de perda. Em 50.000 *snapshots* simulados, o modelo obteve 33% mais eficiência espectral do que a estratégia de potência máxima. Além disso, ficou cerca de 3% abaixo de uma versão baseada em *full CSI* e utilizou, em média, 50% menos potência transmitida que o Erro Quadrático Médio Mínimo Ponderado (*Weighted Minimum Mean Square Error* – WMMSE) [Abode et al., 2022].

Fatiamento de Rede (*Slicing*) (Atenção e Decisão Multiagente): No gerenciamento de *slicing*, o grafo pode representar cooperação entre células, e não interferência física.

No modelo Redes Neurais de Atenção em Grafos com Aprendizado por Reforço Multiagente (*Graph Attention Networks - Multi-Agent Reinforcement Learning – GAT-MARL*), cada uma das estações-base atua como um agente. A *Graph Attention Network* agrega as demandas anteriores das estações vizinhas, definidas por proximidade geográfica, e produz uma representação do estado local. Essa representação alimenta políticas *Deep Q-Network* (DQN) ou *Advantage Actor-Critic* (A2C), que escolhem uma alocação discreta de banda entre três fatias de rede (*slices*). O decisão inter-*slice* é atualizada a cada segundo, enquanto o agendamento dentro de cada *slice* permanece baseado em *round-robin* em intervalos de 0,5 ms [Shao et al., 2021].

A inclusão da GAT elevou a utilidade dos agentes em aproximadamente 4% a 7% em relação às versões DQN e A2C sem grafo, dependendo da granularidade da banda. Nos experimentos, a taxa de satisfação do *slice* URLLC permaneceu em torno de 0,8 e 0,9, enquanto as fatias de rede (*slices*) menos restritivas se aproximaram de 1. Esses resultados mostram que informações de vizinhança podem melhorar decisões de alocação, mas não constituem garantia matemática de SLA, isolamento fim-a-fim ou transferência para outras topologias. Os coeficientes de atenção indicam a influência relativa dos vizinhos na construção do estado; contudo não explicam integralmente a decisão final nem estabelecem causalidade.

Núcleo de Rede (Core) (Classificação de Falhas em Dados Desbalanceados): No núcleo de rede móvel (*Core 5G/5G Advanced*), o modelo de aprendizado de máquina que combina a Transformada de Fourier em Grafos e uma Rede Neural de Passagem de Mensagens (*Graph Fourier Transform - Message Passing Neural Network – GFT-MPNN*) foi proposta para classificação multiclasse de falhas em um conjunto de telemetria com 4.121 com funcionalidade (*features*) e 16 classes: uma classe normal e 15 tipos de falha. A proposta combina uma transformação espectral no grafo com *message passing* para explorar relações entre componentes e métricas. No conjunto real de teste do domínio (*Domain*) C do desafio ITU/KDDI, o modelo alcançou 98,05% de acurácia, *weighted F1* de 0,98 e *macro F1* de 0,97 [Isah et al., 2024]. Esses números devem ser interpretados como desempenho de classificação no *dataset* avaliado. Os domínios simulado e real foram treinados e testados separadamente. O estudo não demonstrou transferência direta do simulador para a rede real. Como os rótulos contêm o componente e o tipo de falha, o modelo pode produzir uma classificação localizada, mas não reconstrói a cadeia causal que originou o evento. Logo, trata-se de classificação de falhas rotuladas, não de *Root Cause Analysis*. A contribuição da GFT para o tratamento do desbalanceamento também deve ser atribuída aos autores, pois a evidência de ablação é limitada.

Borda (Edge) (Alocação de tarefas e Limites de uma Heurística Aprendida): Na borda, uma Rede Convolucional em Grafos (*Graph Convolutional Network – GCN*) foi treinada para imitar soluções de um problema de programação inteira associado à alocação (*placement*) de tarefas de metaverso. O modelo escolhe até dois nós de computação em rede (*In-Network Computing*) ou encaminha a tarefa para um servidor de borda, considerando o custo de rede, a capacidade de processamento e o tempo limite para execução da tarefa. Na versão v1 do artigo, treinada com 100.000 soluções produzidas pelo *solver* Gurobi, a GCN alcançou 97% de acurácia e F1 de 0,96, com custo aproximadamente 2% acima do ótimo [Rashid et al., 2024]. A métrica deve ser versionada, pois uma revisão

posterior do trabalho reporta 92% de acurácia. Esse caso é particularmente instrutivo porque impede uma conclusão simplista sobre aceleração. No *setup* da versão v1, o *solver* Gurobi foi aproximadamente 100 vezes mais rápido que os modelos de aprendizado. A contribuição da GCN foi, portanto, obter soluções de boa qualidade por imitação, sem superar o *solver* em tempo de cálculo. Além disso, o controlador continua verificando os limites de tempo e capacidade, aplicando *fallback* na borda quando necessário.

Os exemplos anteriores mostram que as GNNs podem apoiar regressão temporal, otimização de recursos, decisão sequencial, classificação e imitação de *solvers*. O denominador comum não é uma arquitetura neural específica, mas o uso explícito das relações do problema. Ao mesmo tempo, todos os estudos permanecem em simulação, avaliação *offline*, *datasets* reais ou *testbeds* controlados; nenhum demonstra operação autônoma em malha fechada em uma rede comercial. O NDT é justamente a camada que deve contextualizar esses modelos, validar suas saídas e controlar sua aplicação.

Tabela 4.7. Síntese das aplicações multidomínio de GNNs.

Domínio	Modelo	Grafo principal	Saída	Limitação central
RAN	WCGCN	Pares rádio e relações de interferência/CSI	Potência ou <i>beamforming</i>	Degradação fora da densidade treinada; sem <i>deployment</i> real
RAN 6G	PCGNN-hD	Subredes e distâncias de interferência	Potência por subrede	<i>Full CSI</i> no treino; validação apenas em simulação
Slicing	GAT-MARL	Estações-base e vizinhança geográfica	Banda por <i>slice</i>	Sem garantia de SLA, isolamento E2E ou nova topologia
Core	GFT-MPNN	Componentes, <i>features</i> e dependências	Classe e componente da falha	Sem RCA e sem transferência <i>sim-to-real</i> demonstrada
Edge	GCN placement	Tarefas, ingressos e recursos computacionais	Executor ou <i>offload</i>	Solver foi mais rápido; restrições exigem validação externa

4.6.3. SGCRN: previsão da matriz de tráfego observada

O planejamento e a operação de uma rede precisam estimar como a demanda evoluirá antes de decidir se a resposta adequada será desviar o tráfego, alterar uma política ou expandir a capacidade. Na SGCRN, esse problema é formulado como previsão de curto prazo da matriz de tráfego origem-destino. Para uma rede com n pontos de presença, a matriz X_t possui dimensão $n \times n$, e cada elemento representa o volume observado entre uma origem e um destino no intervalo t . Dada uma sequência de h matrizes históricas, o modelo estima a matriz do próximo intervalo ($\hat{X}_t$). Dessa forma, tem-se que $[X_{t-h}, \dots, X_{t-1}] \rightarrow \hat{X}_t$, onde h define o comprimento da janela de entrada, e não um horizonte futuro de h passos [Zhao et al., 2020]. Planejar capacidade exige estimar quanto tráfego chegará, onde ele surgirá, que recursos consumirá e com que nível de serviço deverá ser atendido. O método tradicional combina séries históricas, crescimento esperado,

sazonalidade, percentis, margens de segurança e cenários de falha. Essa abordagem permanece válida, mas sua precisão diminui quando a demanda se torna não estacionária, especialmente móvel e condicionada por decisões de aplicações, plataformas de conteúdo e orquestradores.

Serviços imersivos, transferência de modelos de IA, sensoriamento, uplink intensivo, mobilidade e edge computing alteram não apenas o volume, mas a direção, a duração e o local de concentração do tráfego. A criação, modificação ou remoção de slices acrescenta outra dimensão: parte da capacidade pode estar fisicamente disponível e, ainda assim, indisponível para uma classe específica por causa de isolamento, políticas ou reservas. Em redes não terrestres (*Non Terrestrial Networks* – NTN), a cobertura e a capacidade podem variar com o tempo; em cenários de desastre, grandes populações de usuários podem migrar abruptamente para redes hospedeiras. Assim, a previsão da demanda (*forecast*) deixa de ser apenas extrapolação de uma série agregada e passa a depender do contexto da rede e do serviço. Mesmo nas redes IP tradicionais, a demanda já era observada de forma incompleta. Contadores de interface indicam carga por enlace, mas não revelam diretamente a matriz origem-destino. Protocolos como NetFlow ou IPFIX fornecem visibilidade de fluxos, porém podem ser amostrados, parciais ou coletados apenas em determinados pontos. A inferência de matrizes de tráfego por técnicas de tomografia, como Tomogravity, surgiu justamente da necessidade de combinar cargas de enlace e restrições de roteamento para estimar variáveis que não são diretamente mensuradas [Zhang et al., 2003a]. Em ambientes 5G/6G, essa dificuldade é ampliada por funções virtuais, múltiplos acessos, mobilidade, estados de rádio e recursos computacionais. Outro problema é distinguir tráfego medido de demanda oferecida. O volume efetivamente transportado pode ser menor que o volume que usuários e aplicações tentaram submeter à rede. Congestionamento, perda, policiamento (*policing*), conformação (*shaping*), controle de admissão (*admission control*), TCP backoff e adaptação de taxa de bit (*adaptive bitrate*) limitam o tráfego observado. Uma interface saturada pode apresentar uma curva aparentemente estável e induzir a conclusão de que o crescimento cessou, quando a própria capacidade está reprimindo a demanda. Após uma expansão, parte dessa demanda pode reaparecer como crescimento inesperado. A inferência contrafactual do volume que existiria sem a restrição continua sendo um problema aberto; por isso, o forecast deve representar incerteza e não depender apenas do histórico transportado.

O sobreprovisionamento (*overprovisioning*) é uma resposta racional a essas incertezas. A folga de capacidade em enlaces (*Headroom*) protege a rede contra erros de previsão, falhas, picos e atrasos no processo de expansão. Ele não pode ser tratado simplesmente como desperdício. Entretanto, quando a folga de capacidade se torna o principal mecanismo para compensar baixa confiança em dados, modelos e decisões, o operador paga por uma incerteza que permanece sem ser explicitamente quantificada. Novos objetivos de eficiência energética e sustentabilidade tornam esse equilíbrio ainda mais delicado: desligar recursos para economizar energia reduz a margem disponível para eventos inesperados, enquanto manter todos os recursos ativos eleva custo e consumo.

Consequentemente, o capacity planning e forecasting precisam funcionar em conjunto. A previsão deve alimentar cenários de capacidade, falha, crescimento, elasticidade de slices e deslocamento de aplicações. O planejamento, por sua vez, precisa indicar onde a incerteza da previsão é operacionalmente aceitável e onde exige margem adicio-

nal, medição mais precisa ou alternativas de engenharia de tráfego. A saída desejada não é um único número de crescimento, mas um conjunto de cenários com confiança, riscos e consequências explícitas.

A necessidade de uma arquitetura espaço-temporal decorre de duas propriedades do tráfego. A primeira é temporal: os volumes Origem-Destino (OD) apresentam sazonalidade, tendência e transições não lineares entre períodos de baixa e alta utilização. A segunda é estrutural: os pontos de presença pertencem a uma rede, e o comportamento observado em uma região está relacionado ao estado das regiões conectadas. Modelos puramente temporais processam cada série ou a matriz vetorizada sem representar explicitamente essa conectividade; convoluções bidimensionais, por sua vez, tratam a matriz como uma grade regular, embora a rede física não possua a geometria de uma imagem.

A SGCN representa a rede como um grafo $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \mathbf{A})$, no qual $\mathbf{V}$ contém PoPs ou roteadores, $\mathbf{E}$ contém os enlaces físicos e $\mathbf{A}$ é a matriz de adjacências. Nos experimentos, essa matriz permanece fixa. Cada representação do grafo em um único instante de tempo específico (*snapshot*) X_t passa por uma convolução no grafo que combina as funcionalidades (*features*) dos nós conforme sua vizinhança topológica. Essa formulação introduz o conhecimento estrutural da rede no modelo, mas exige uma cautela: a adjacência física não representa diretamente o compartilhamento de todos os caminhos de roteamento. Dois fluxos OD podem competir pelo mesmo recurso sem que essa relação esteja explicitada como uma entidade separada, como ocorre posteriormente na família RouteNet.

Na Equação 5, $\tilde{A}$ inclui autoconexões, $\tilde{D}$ normaliza a contribuição da vizinhança e W_s contém os parâmetros aprendidos. O resultado Y_t é uma representação espacial do *snapshot*. O objetivo didático da equação não é derivar a convolução espectral, mas evidenciar que a informação de tráfego é filtrada pela estrutura do grafo antes de ser submetida ao componente temporal. Cada intervalo histórico produz uma representação espacial e a sequência dessas representações alimenta uma GRU (*Gated Recurrent Unit*).

$$Y_t = \delta \left(\tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2} X_t W_s \right). \quad (5)$$

A GRU atualiza um estado oculto que combina memória do passado e informação presente por meio das portas de *reset* e *update*. Em comparação com uma LSTM (*Long Short-Term Memory*), a GRU utiliza menos mecanismos de controle e menos parâmetros, mantendo a capacidade de modelar dependências temporais de curto e médio alcance. No contexto da SGCN, o estado oculto não deve ser interpretado como uma explicação causal da evolução do tráfego. O modelo aprende correlações temporais condicionadas ao histórico e à topologia; a precedência cronológica não demonstra que uma variação em determinado PoP tenha causado a variação observada em outro.

O último estado da GRU é transformado por uma camada totalmente conectada para gerar $\hat{X}_t$. O treinamento minimiza o erro quadrático entre a matriz prevista e a observada, e a saída é, portanto, uma estimativa pontual determinística. O artigo divide os dados cronologicamente em 75% para treinamento, 5% para validação e 20% para teste. Essa separação preserva a ordem temporal, mas não acrescenta ao modelo intervalos de confiança, quantis ou probabilidade de excedência, elementos que seriam necessários para

traduzir a previsão em margem de engenharia ou risco de investimento.

Como exemplo considere uma topologia simplificada de **3 PoPs** ($n = 3$) interconectados em linha ($P_1 \leftrightarrow P_2 \leftrightarrow P_3$).

Dados de Entrada

1. **Matriz de Adjacência $\tilde{A}$ (com auto-conexões):**

$$\tilde{A} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

2. **Matriz Normalizada de Graus ($\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}$):** Calculando os graus ($\tilde{D} = \text{diag}(2, 3, 2)$), a matriz de propagação normalizada é dada por:

$$\hat{A}_{\text{norm}} \approx \begin{bmatrix} 0,50 & 0,41 & 0 \\ 0,41 & 0,33 & 0,41 \\ 0 & 0,41 & 0,50 \end{bmatrix}$$

3. **Matriz de Tráfego Observada no Instante t (X_t , em Gbps):**

$$X_t = \begin{bmatrix} 0 & 10,0 & 5,0 \\ 8,0 & 0 & 12,0 \\ 4,0 & 15,0 & 0 \end{bmatrix}$$

4. **Matriz de Pesos Aprendidos (W_s – assumindo $F = 2$ filtros):**

$$W_s = \begin{bmatrix} 0,2 & -0,1 \\ 0,5 & 0,3 \\ 0,1 & 0,4 \end{bmatrix}$$

Passo a Passo dos Cálculos

Passo 1: Agregar Informação Espacial dos Vizinhos ($\hat{A}_{\text{norm}} \cdot X_t$) Multiplica-se a estrutura topológica pela matriz de tráfego para que cada PoP receba a influência do volume de tráfego dos seus vizinhos diretos:

$$X_{\text{espacial}} = \hat{A}_{\text{norm}} \cdot X_t = \begin{bmatrix} 3,28 & 5,00 & 7,42 \\ 4,28 & 10,25 & 6,01 \\ 3,28 & 11,65 & 4,92 \end{bmatrix}$$

Passo 2: Projeção de Atributos ($X_{\text{espacial}} \cdot W_s$) Aplica-se a matriz de transformação treinável W_s :

$$Y_{\text{linear}} = X_{\text{espacial}} \cdot W_s = \begin{bmatrix} 3,90 & 4,14 \\ 6,58 & 5,05 \\ 6,98 & 5,14 \end{bmatrix}$$

Passo 3: Aplicação da Função de Ativação ($\sigma = \text{ReLU}$) Como todos os valores de Y_{linear} são positivos, a ativação mantém os valores originais:

$$Y_t = \begin{bmatrix} 3,90 & 4,14 \\ 6,58 & 5,05 \\ 6,98 & 5,14 \end{bmatrix}$$

No trabalho relacionado [Zhao et al., 2020] a avaliação utiliza quatro topologias: Abilene, Nobel, GÉANT e Germany50. As três primeiras matrizes têm 12, 17 e 22 nós; Germany50 possui 50. Os *snapshots* são agregados em intervalos de 5 minutos, exceto na GÉANT, cuja granularidade é de 15 minutos. Na GÉANT, a SGCRN atingiu um R^2 médio de $0,9642 \pm 0,0001$, superando os *baselines* avaliados. Na Abilene, Nobel e Germany50, os valores foram 0,8220, 0,6922 e 0,5215, respectivamente. O próprio contraste entre conjunto de dados (*datasets*) mostra que a precisão não é uma propriedade universal do modelo: ela depende do volume de dados, da estrutura da rede e da previsibilidade do sinal. O tempo de teste foi descrito pelos autores como muito inferior a um segundo, mas sem um valor numérico tabulado que permita generalizar a latência de inferência.

Para a engenharia de tráfego, a matriz prevista pode ser projetada sobre o roteamento atual ou sobre configurações candidatas. Esse passo permite antecipar enlaces com risco de saturação e preparar uma resposta antes que o pico se materialize. A SGCRN, entretanto, não calcula a utilização dos enlaces, o atraso nem a perda. Essas grandezas dependem da interação entre demanda, capacidades, caminhos, filas e políticas. A função do modelo é fornecer um possível estado futuro da demanda; a avaliação das consequências cabe a um modelo de desempenho, a um simulador ou a um *solver*.

A contribuição para o planejamento de capacidade é mais indireta. Uma previsão mais à frente (*one-step-ahead*) é útil para decisões táticas e para detectar padrões recorrentes, mas não resolve o planejamento de meses ou anos. A expansão de capacidade exige combinar crescimento estrutural, cenários econômicos, falhas, incerteza, custos, prazos de implantação e critérios de risco. A SGCRN original não modela esses fatores. Ela pode alimentar um fluxo mais amplo, mas não substitui um método de predição de longo prazo nem a avaliação de cenários de investimento.

Outra limitação é a diferença entre o tráfego observado e a demanda oferecida. Sob congestionamento, perdas, TCP *backoff*, *shaping*, *policing* ou controle de admissão podem impedir que toda a intenção de tráfego apareça nos contadores. O modelo aprende o sinal transportado disponível no histórico e tende a reproduzir o teto observado. Em um enlace saturado, uma previsão precisa do tráfego medido pode ainda subestimar a capacidade que seria consumida se o gargalo fosse removido. Por isso, a saída não deve ser descrita como “demanda real” nem como estimativa de demanda reprimida.

Finalmente, o estudo não avalia topologia variável, dados faltantes, *drift* estrutural ou eventos raros. Como a matriz A permanece fixa, não há evidência experimental para afirmar como o modelo reagiria a uma expansão, falha ou mudança persistente de roteamento. Também não se pode concluir, apenas pela arquitetura, que toda alteração exigiria, obrigatoriamente, retreinamento completo. A formulação segura é mais restrita: a robustez às mudanças de topologia não foi demonstrada. Esses limites motivam o uso de mecanismos externos de detecção de *drift*, imputação, quantificação de incerteza e va-

lidação de domínio no NDT. Mesmo com uma boa previsão de $\hat{X}_t$, permanece a pergunta central: *como a rede se comportará sob essa demanda e uma configuração determinada?*

4.6.4. Família RouteNet: Modelagem de Desempenho da Rede

Responder à pergunta anterior por meio de simulação pode exigir a geração e o processamento de grandes volumes de pacotes. Quanto maiores forem o cenário, o horizonte e a carga, maior tende a ser o custo computacional. Modelos analíticos são mais rápidos, mas dependem de hipóteses que nem sempre refletem o tráfego, as filas e os escalonadores reais. A família RouteNet explora um ponto intermediário: aprende um modelo substituto, (*surrogate*), que recebe topologia, roteamento, demanda e configuração dos recursos e estima KPIs de desempenho sem reproduzir cada evento de pacote.

Dentro de um NDT, o RouteNet funciona como um modelo funcional para análise de cenários hipotéticos (*what-if*). O gêmeo digital organiza os dados e a configuração do cenário; o RouteNet estima suas consequências; um mecanismo de decisão compara alternativas; e controladores ou processos operacionais aplicam apenas as configurações autorizadas. Esse papel não transforma o modelo em um NDT completo nem elimina a necessidade de simuladores. Simuladores permanecem importantes para produzir *ground truth*, investigar fenômenos em granularidade de pacote e validar situações fora do domínio de treinamento.

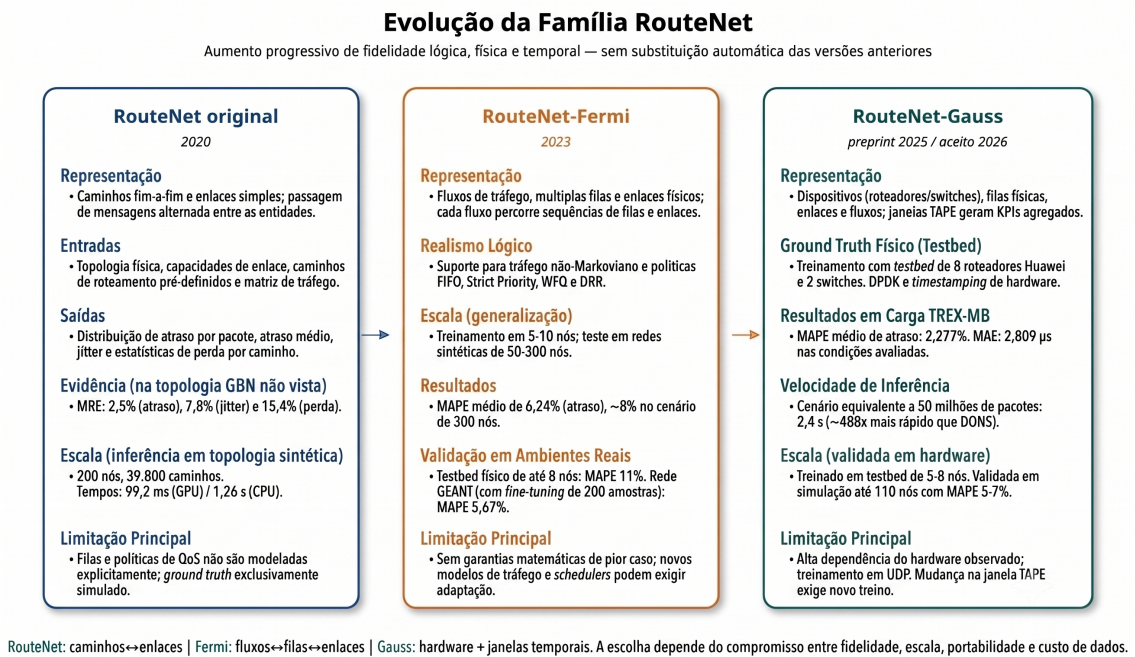


Figura 4.9. Evolução representacional da família RouteNet: caminhos e enlaces no RouteNet original; fluxos, filas e enlaces no RouteNet-Fermi; dispositivos, filas, enlaces e fluxos, com janelas TAPE, no RouteNet-Gauss. As versões ampliam a fidelidade lógica, física e temporal, mas não se substituem automaticamente.

4.6.4.1. RouteNet Original: Dependência entre Caminhos e Enlaces

O RouteNet original parte de duas entidades: caminhos origem-destino e enlaces. Cada caminho é representado como uma sequência ordenada de enlaces, preservando o fato de que a posição do recurso no percurso influencia o estado acumulado. A dependência entre as entidades é circular. O desempenho de um caminho depende dos enlaces que ele atravessa; o estado de um enlace depende do conjunto de caminhos que concorre por sua capacidade. Essa relação é difícil de representar por uma rede neural convencional com entrada de dimensão fixa, mas se encaixa naturalmente em *message passing* com parâmetros compartilhados [Rusek et al., 2020].

Em cada iteração, uma rede recorrente percorre os enlaces na ordem de cada caminho e atualiza o estado oculto desse caminho. As mensagens produzidas pelos caminhos são então agregadas em cada enlace, atualizando o estado do recurso compartilhado. O novo estado dos enlaces alimenta outra atualização dos caminhos, e o processo é repetido por um número definido de iterações. A agregação nos enlaces é invariante à ordem dos caminhos concorrentes. A atualização do caminho, em contraste, preserva a sequência dos enlaces. Essa combinação permite representar simultaneamente o compartilhamento de recursos e a ordem topológica.

As entradas incluem topologia, capacidade dos enlaces, caminhos definidos pelo roteamento e matriz de tráfego. O modelo produz, para cada caminho, uma distribuição de atraso por pacote e uma estimativa de perda. Dessa distribuição, derivam-se o atraso médio e o *jitter*. A unidade de saída é agregada por caminho ou por par origem-destino (OD). O RouteNet não reproduz o estado de cada pacote, não reconstrói a fila em cada instante e não fornece garantia de pior caso. Sua utilidade é avaliar rapidamente uma configuração no domínio aprendido.

O treinamento foi realizado com amostras geradas no OMNeT++ para as topologias NSF, Geant2 e Germany50. A avaliação incluiu também a GBN, de 17 nós, não vista no treinamento. Nessa topologia, o *Mean Relative Error* foi de 2,5% para atraso, 7,8% para *jitter* e 15,4% para perda. O valor de 15,4% deve ser associado à perda e constitui o pior erro entre esses KPIs; não é o erro de atraso. Em um experimento de inferência com uma topologia sintética de 200 nós e 39.800 caminhos, o modelo levou 99,2ms em GPU e 1,26s em CPU. Esses resultados demonstram transferência para uma topologia não vista e capacidade de processar cenários maiores, mas não autorizam a generalização para qualquer topologia, roteamento ou regime de tráfego. A GBN tem porte semelhante ao das topologias usadas no treino, e os dados permanecem em um ambiente simulado com modelos simplificados. A validade depende das interações da rede-alvo permanecerem suficientemente próximas às distribuições englobadas pelos dados. Além disso, filas, classes de serviço e políticas de escalonamento não aparecem como entidades explícitas. A necessidade de representar esses elementos leva ao outro modelo da família: o RouteNet-Fermi.

4.6.4.2. RouteNet-Fermi: Fluxos, Filas e Políticas de QoS

O RouteNet-Fermi reformula o grafo para lidar com tráfego mais realista, múltiplas filas e diferentes políticas de atendimento. Em vez de modelar somente caminhos e enlaces, ele representa fluxos, filas e enlaces. Cada fluxo percorre uma sequência de pares fila-enlace. Assim, duas classes podem compartilhar o mesmo enlace e, ao mesmo tempo, receber tratamento diferente por filas ou escalonadores distintos. Essa decomposição introduz no grafo parte do conhecimento que, em modelos de teoria de filas, aparece por meio de hipóteses analíticas [Ferriol-Galmés et al., 2023].

O *message passing* atualiza o estado de cada fluxo com base na sequência de recursos percorridos. As mensagens dos fluxos são agregadas nas filas concorrentes, e as filas de saída contribuem para a atualização do estado dos enlaces. Como no RouteNet original, a arquitetura combina operações invariantes à permutação nos recursos compartilhados com atualizações que preservam a ordem do percurso. A diferença é que o caminho lógico se torna uma sequência de estados de fila e de enlace, aproximando a representação do mecanismo que produz espera, *jitter* e perda.

A arquitetura foi avaliada com tráfego não Markoviano e políticas como FIFO, *Strict Priority*, *Weighted Fair Queuing* e *Deficit Round Robin*. O objetivo não é ensinar o funcionamento de cada escalonador, mas mostrar que a política de serviço altera as interações entre fluxos e, portanto, precisa fazer parte do cenário. O compartilhamento de pesos permite que o mesmo conjunto de funções neurais seja aplicado a números diferentes de fluxos, filas e enlaces, condição necessária para testar redes maiores que as vistas no treinamento.

O atraso fim-a-fim não é ignorado nem simplesmente lido como um rótulo global. Fermi o reconstrói a partir de contribuições associadas à ocupação e ao serviço das filas ao longo do percurso. Essa decomposição favorece variáveis mais relacionadas aos mecanismos físicos e menos dependentes do tamanho absoluto de uma topologia específica. Ainda assim, a saída continua sendo uma previsão aprendida. Ela não estabelece um limite determinístico de pior caso, não substitui o *Network Calculus* e não dispensa validação quando o SLA exige garantias estritas.

A evidência de generalização de escala foi produzida ao treinar em topologias de 5 a 10 nós e avaliar redes sintéticas de 50 a 300 nós. Em um conjunto de 1.000 amostras, o Erro Percentual Absoluto Médio (*Mean Absolute Percentage Error* – MAPE) médio de atraso foi de 6,24%; no cenário de 300 nós, o pior resultado ficou próximo de 8%. A formulação correta é “generalização de escala demonstrada nas condições experimentais”, e não “invariância de escala”. O modelo compartilha parâmetros e pode ser aplicado a grafos maiores, mas a precisão continua dependente dos modelos de tráfego, políticas, cargas e configurações utilizados no treinamento.

O trabalho também separa avaliações com fontes de dados distintas. Em um *testbed* físico de até oito nós, o MAPE reportado foi de aproximadamente 11%. Em uma avaliação com topologia e *traces* reais, após *fine-tuning* com 200 amostras da GÉANT, o MAPE foi de 5,67%. Esses resultados não devem ser combinados como se pertencessem ao mesmo experimento. A base principal de treinamento e escala permanece simulada. O *testbed* e os *traces* reais servem como validações adicionais e mostram que a adaptação

com dados do domínio-alvo pode ser necessária.

A inferência em CPU cresceu de 48,03ms em 10 nós para 613,3ms em 110 nós nas condições reportadas. O crescimento é substancialmente menor que o custo de simulação pacote a pacote, mas não elimina a dependência do hardware, da implementação e do número de entidades no grafo. Novos modelos de tráfego, *schedulers* não contemplados e equipamentos com comportamento diferente podem exigir novos dados ou ajustes no modelo. Fermi amplia o realismo lógico e a transferência de escala. A diferença entre simulação e comportamento físico do equipamento, porém, permanece. É essa lacuna que o RouteNet-Gauss procura reduzir.

4.6.4.3. RouteNet-Gauss: Hardware e Dinâmica Temporal

O RouteNet-Gauss [Güemes-Palau et al., 2026] desloca a fonte principal de dados reais (*ground truth*) de um simulador para um *testbed* com hardware comercial. A motivação é o *reality gap*: mesmo um simulador detalhado pode divergir de *buffers*, implementações de filas, temporização e *pipelines* de encaminhamento de equipamentos reais. O modelo não é implantado numa rede de produção. Em vez disso, ele utiliza o *testbed* como gerador acelerado de dados e como referência física para aprender o comportamento observado.

O ambiente físico é formado por oito roteadores e dois *switches*, com geração de tráfego, captura baseada em Kit de Desenvolvimento de Plano de Dados (*Data Plane Development Kit* – DPDK) e registro de tempo (*timestamping*) de hardware. A decomposição do grafo inclui dispositivos, filas, enlaces e fluxos. Essa expansão procura representar não apenas a competição lógica por capacidade, mas também o efeito de elementos físicos e de sua organização interna. Como consequência, a fidelidade tende a aumentar no parque semelhante ao utilizado para gerar os dados, ao custo de maior dependência do hardware e da configuração observados.

Outra contribuição é a Estimação de Desempenho Temporal Agregada (*Temporal Aggregated Performance Estimation* – TAPE). Em vez de produzir uma única média ao longo de toda a duração do cenário, o TAPE divide a observação em janelas e gera uma sequência de KPIs por fluxo e por intervalo. Essa granularidade permite representar variações não estacionárias que desapareceriam na média global. O recurso não equivale a reconstrução pacote a pacote e não é arbitrariamente reconfigurável após o treinamento: alterar a duração das janelas modifica o alvo e requer novo treinamento do modelo.

Nos cenários sintéticos com gerador de tráfego *open source* com multi-rajadas (TREX-MB), o RouteNet-Gauss reportou MAPE médio de atraso de 2,277% e MAE de 2,809 μ s, enquanto o OMNeT++ apresentou 56,122% e 67,807 μ s nas mesmas condições. O contraste evidencia a vantagem de aprender com *ground truth* do *testbed* para aquele conjunto de cenários; ssa forte dependência do testbed físico onde foi treinado. Em um cenário correspondente a 50 milhões de pacotes, a inferência levou 2,4s, aproximadamente 488 vezes mais rápida que o DeepOnet (*DONS*). O aumento na velocidade depende do número de pacotes, da topologia, da quantidade de janelas e da implementação comparada.

A avaliação de escala também exige separação cuidadosa. O modelo foi treinado em topologias de 5 a 8 nós e aplicado a amostras de redes simuladas com até 110 nós, mantendo MAPE entre aproximadamente 5% e 7%. Não existiu um *testbed* físico de 110 roteadores. O experimento demonstra que a arquitetura relacional pode ser transferida para grafos maiores em dados simulados, enquanto a fidelidade física foi estabelecida no *testbed* menor. Confundir essas duas dimensões faria parecer que o hardware em grande escala foi validado diretamente, o que o artigo não reporta.

As limitações decorrem justamente da proximidade com o hardware. Equipamentos de outro fornecedor, políticas internas distintas, protocolos não cobertos e regimes de tráfego muito diferentes podem exigir novos dados ou um ajuste fino (*fine-tuning*). O treinamento reportado concentra-se em UDP, sendo que incorporar TCP requer representar suas dinâmicas de controle e interação com perdas e filas. Nos *traces* RWPT (*Real-World Packet Traces*), a vantagem também não é uniforme: o OMNeT++ pode ser superior em *jitter* e em algumas medidas de correlação ou percentis. O RouteNet-Gauss reduz a lacuna da realidade (*reality gap*) nos cenários avaliados, mas não torna o simulador obsoleto nem transforma um *testbed* específico em um modelo universal.

A evolução RouteNet-Fermi/RouteNet-Gauss pode ser resumida como um aumento progressivo de fidelidade. O modelo original demonstra a dependência entre caminhos e enlaces, O RouteNet-Fermi acrescenta fluxos, filas, classes e políticas de serviço. Já o RouteNet-Gauss acrescenta hardware e variação temporal baseada em janelas. Cada passo resolve uma lacuna e introduz novos custos: mais dados, maior especificidade do domínio, mais entidades no grafo e maior necessidade de validação. A escolha entre as versões depende do problema e do grau de fidelidade requerido, não de uma hierarquia em que a versão mais recente substitui automaticamente todas as anteriores.

No ciclo do NDT, uma matriz observada ou prevista define a demanda de um cenário. A família RouteNet estima como topologia, roteamento, filas e configuração afetam os KPIs. Um mecanismo de otimização pode consultar o modelo repetidamente para comparar alternativas. A decisão, contudo, não é produzida pela família RouteNet: ela precisa vir de um *solver*, de uma política aprendida ou de regras de engenharia. O próximo passo da seção aborda justamente esse papel prescritivo por meio do TEAL, que calcula *split ratios* candidatos e usa otimização explícita para reduzir violações de capacidade.

4.6.5. TEAL: Alocação Candidata para Engenharia de Tráfego

A família RouteNet responde a uma pergunta avaliativa: como uma determinada combinação de topologia, roteamento e demanda afeta os KPIs da rede? O TEAL aborda uma pergunta diferente e prescritiva: como distribuir a demanda entre caminhos candidatos com rapidez suficiente para acompanhar a evolução de uma WAN de grande escala? A saída não é uma previsão de atraso, *jitter* ou perda, mas um conjunto de *split ratios* que define que fração de cada demanda deve seguir por cada caminho. O objetivo padrão do trabalho é maximizar o volume total atendido sem exceder as capacidades dos enlaces, embora versões separadamente treinadas também sejam avaliadas para minimizar a máxima utilização ou incorporar penalização por atraso [Xu et al., 2023].

$$0 \leq F_d(p) \leq 1, \quad \sum_{p \in P_d} F_d(p) \leq 1, \quad (6)$$

$$\sum_{d, p: e \in p} D_d F_d(p) \leq C_e, \quad \forall e \in E. \quad (7)$$

Representação de Demandas, Caminhos e Gargalos: Uma GNN construída apenas sobre roteadores não expressa diretamente a relação entre uma demanda, seus caminhos alternativos e os enlaces que esses caminhos compartilham. O TEAL introduz a FlowGNN, um grafo bipartido em que *EdgeNodes* representam enlaces e *PathNodes* representam uma demanda sobre um caminho candidato. Existe uma conexão quando o caminho atravessa o enlace. A capacidade inicializa o estado do *EdgeNode*, enquanto o volume da demanda inicializa os quatro *PathNodes* associados ao par origem-destino.

A arquitetura alterna camadas de GNN e camadas densas. As primeiras propagam informação sobre competição por capacidade entre caminhos que compartilham enlaces; as segundas coordenam os caminhos pertencentes à mesma demanda. Na configuração experimental, seis pares de camadas produzem um *embedding* de seis componentes para cada *PathNode*. Esse *embedding* resume a posição de um caminho no problema de alocação: demanda transportada, recursos compartilhados e potenciais gargalos. A FlowGNN, portanto, não modela filas nem QoS detalhada como o RouteNet-Fermi. Sua finalidade é estruturar o problema de capacidade para a etapa decisória.

Decisão Multiagente e Razões de Divisão: Uma política neural monolítica teria de receber e produzir milhões de valores. Para reduzir essa dimensão, o TEAL trata cada demanda como um agente lógico. Cada agente observa somente os *embeddings* dos seus quatro caminhos e usa uma política compartilhada para produzir quatro frações. O treinamento é centralizado e otimiza uma recompensa global. A execução também ocorre no controlador, embora a decomposição permita processar as demandas em paralelo. Assim, o termo multiagente descreve a organização computacional do problema, não os agentes distribuídos nos roteadores da WAN.

O algoritmo de treinamento é uma adaptação de Gradientes de Política Multiagente Contrafactual (*Counterfactual Multi-Agent Policy Gradients* – COMA). Para atribuir crédito, calcula-se como o resultado global mudaria se a ação de uma demanda fosse alterada enquanto as demais permanecessem fixas. Esse contrafactual é local ao mecanismo de aprendizado e não deve ser confundido com a análise *what-if* completa de um NDT. A política inicial usa uma função *softmax* e distribui integralmente a demanda entre os quatro caminhos. Essa ação neural, entretanto, ainda pode sobrecarregar os enlaces compartilhados, como visto na Figura 4.10.

Refinamento por ADMM: O papel do Método dos Multiplicadores de Direção Alternada (*Alternating Direction Method of Multipliers* – ADMM) é preservar a separação entre uma solução rápida aprendida e as restrições explícitas do problema. O método introduz variáveis auxiliares, folgas e um lagrangiano aumentado para decompor a otimização em subproblemas paralelizáveis por demanda, caminho ou enlace. A alocação neural funciona como *warm start*: em vez de iniciar em um ponto aleatório e executar

GÊMEO DIGITAL TEAL: ARQUITETURA SIMPLIFICADA

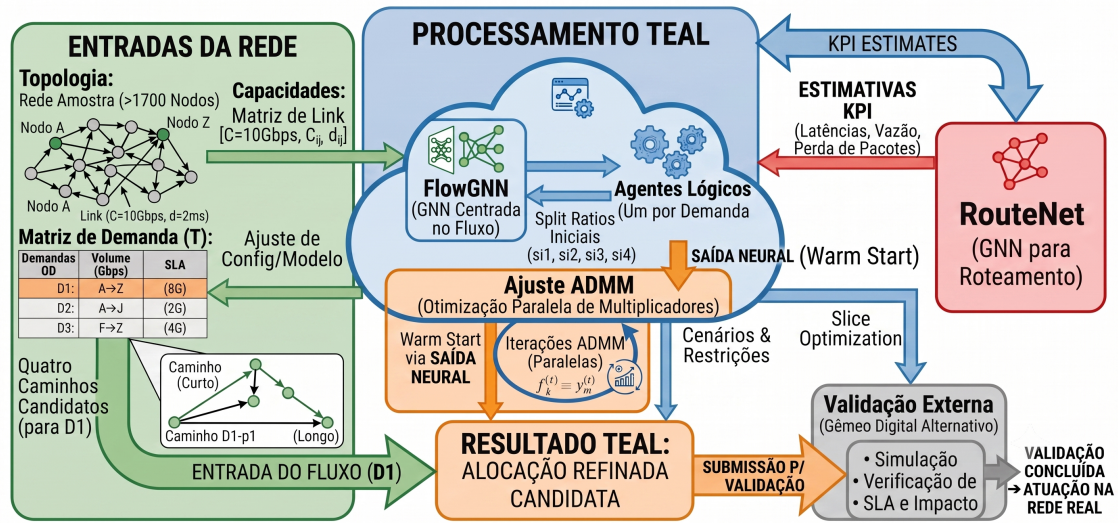


Figura 4.10. Arquitetura do TEAL.

muitas iterações, o refinador começa próximo de uma região de boa qualidade. Nos experimentos, o TEAL utiliza duas iterações em redes com menos de 100 nós e cinco nas demais. Essa combinação não autoriza afirmar que a GNN gera uma solução viável e que o ADMM elimina todas as violações. A ablação mostra que remover o refinamento reduz a demanda satisfeita em aproximadamente 2% a 2,5%, evidenciando sua contribuição, mas não estabelece uma garantia universal de factibilidade após poucas iterações. O resultado final continua sendo uma solução candidata, que deve ser verificada pelo sistema de controle. O ADMM tampouco é uma técnica de aprendizado: é o componente de otimização explícita que ancora a política neural nas restrições de capacidade.

Escala, Dados e Qualidade da Solução: No exemplo de avaliação da Microsoft, foram utilizados 20 dias de matrizes coletadas da *Software-Driven Wide Area Network (SWAN)*, uma WAN inter-*datacenter* da Microsoft, em intervalos de cinco minutos. Os *traces* são mapeados para topologias públicas e privadas, incluindo Operadoras tais como B4, UsCarrier, Kdl e ASN; 700 intervalos são usados no treinamento, 100 na validação e 200 no teste. A presença de dados coletados em produção não implica a implantação do TEAL na SWAN: os experimentos são avaliações *offline*. Entre os comparadores estão a programação linear total (*Linear Programming* – LP-all) com Gurobi, LP-top, NCFlow, POP e Engenharia de Tráfego baseada em Valor em Risco (*Traffic Engineering Applying Value at Risk* – TEAVAR), uma variante de engenharia de tráfego probabilística orientada ao risco de falhas adaptada ao objetivo de fluxo atendido.

Na Kdl, com 754 nós e 1.790 enlacs, o tempo médio foi 0,95 s. Na ASN, com 1.739 nós e 8.558 enlacs, foi 0,97 s, variando entre 0,89 e 1,08 s. Nessa última topologia, o TEAL foi 394 vezes mais rápido que o NCFlow, 625 vezes mais rápido que o POP e 197 vezes mais rápido que o LP-top. A demanda satisfeita atingiu 89,4% na métrica

online, que incorpora o atraso de cálculo. Essas acelerações (*speedups*) pertencem a linha base (*baselines*) e ao hardware do experimento, não devendo ser condensados em uma aceleração universal de 197 a 625 vezes.

Quando se remove a penalização do tempo de cálculo e se compara apenas a qualidade da solução, o resultado também precisa ser contextualizado. Na Kdl, o TEAL ficou 4,8% abaixo do LP-all, que representa a solução ótima obtida pelo *solver*, e apenas 0,7% abaixo do melhor esquema viável LP-top. O ganho central é, portanto, um compromisso entre tempo e qualidade: o método abre mão de parte da otimalidade para produzir uma alocação em aproximadamente um segundo. A expressão mais adequada é tempo quase real para o ciclo de TE estudado, e não tempo real com garantia ótima.

Generalização, Retreinamento e Validação Operacional: O modelo é treinado separadamente para cada topologia e objetivo. Ele aceita matrizes inéditas na mesma topologia e pode tratar uma falha transitória definindo a capacidade do enlace como zero, sem retreinamento imediato. Mudanças permanentes, como a adição de nós ou enlaces, exigiram retreinamento de 6 a 10 horas. O treinamento inicial de uma topologia grande levou aproximadamente uma semana. O artigo não demonstra transferência para uma topologia inteiramente não vista, para outro conjunto estrutural de caminhos ou para um objetivo novo sem novo treinamento.

Também há limites de robustez temporal. Sob uma flutuação sintética extrema de 20 vezes, o desempenho ficou 2,3% abaixo do LP-top, e os autores sugerem executar em paralelo um método alternativo capaz de assumir a decisão quando a entrada estiver fora da distribuição. Em operação, esse mecanismo precisa ser complementado por detecção de desvios (*drift*), avaliação de confiança, limites para mudanças, análise de grupos de enlace que compartilham risco (*Shared Risk Link Group* – SRLGs), políticas de disponibilidade e de retorno ao estado anterior (*rollback*). O TEAL acelera a otimização, não eliminando a engenharia de segurança necessária para fechar o ciclo (*loop*).

Dentro de um NDT, o TEAL ocupa a função prescritiva. A matriz utilizada pode vir de observação corrente ou de um *forecast* como a SGCRN. Modelos de desempenho, como o RouteNet, podem avaliar os KPIs de uma alocação candidata e o controlador decide se a solução pode ser aplicada. Essa composição é uma possibilidade arquitetural, não um *pipeline* obrigatório nem uma solução integrada originalmente proposta pelos autores. Trabalhos posteriores, como MAGNETO, GROM, TELGEN e LO-TE, exploraram alternativas para políticas distribuídas, generalização topológica, alinhamento com algoritmos clássicos e refinamento por *solvers*, mostrando que a combinação entre aprendizado e otimização permanece uma área ativa [Bernárdez et al., 2023, Ding et al., 2024, Zhou et al., 2025, Liu et al., 2025].

4.7. Desafios Abertos e Tecnologias Habilitadoras

Os modelos examinados na seção anterior demonstram que as GNNs e os métodos híbridos de aprendizado e otimização podem antecipar matrizes de tráfego, estimar indicadores de desempenho e calcular alocações candidatas em escalas relevantes. Essa capacidade experimental, porém, não equivale automaticamente à prontidão operacional. Entre uma previsão produzida em laboratório e uma mudança segura na rede física

existe uma cadeia de dependências: dados precisam ser coletados com semântica e tempo adequados; estados não observáveis precisam ser inferidos com incerteza conhecida; modelos devem preservar desempenho fora do ambiente de treinamento; decisões precisam respeitar restrições rígidas; e toda atuação deve ser monitorada, explicada, autorizada e reversível.

Esta seção trata dessas condições de operacionalização. O foco não é reapresentar a rede neural recorrente convolucional espaço-temporal em grafos (SGCRN), RouteNet ou TEAL, mas explicitar o que ainda limita seu uso como componentes dos gêmeos digitais de rede (NDT). A tese central é que a utilidade de um NDT não depende apenas da qualidade do modelo analítico. Ela depende da coerência entre os dados, a representação, a atualização temporal, o domínio de validade, os mecanismos de controle e a governança. Um modelo rápido e preciso pode produzir uma recomendação inadequada se os dados estiverem atrasados, se o estado relevante for parcialmente observável ou se a rede tiver mudado de forma não representada durante o treinamento.

4.7.1. Dados, Telemetria e Sincronização

A base de um NDT é um conjunto de dados suficientemente amplo e coerente para a decisão que o gêmeo digital pretende apoiar. Esse conjunto pode incluir topologia, inventário, configurações, capacidades, matrizes de tráfego, contadores, eventos, estado de filas, políticas, indicadores de serviço e contexto do domínio. A ITU-T Y.3090 e o ETSI GR ZSM 015 (*Group Report for Zero-touch network and Service Management*) tratam o domínio de dados como componente estrutural do gêmeo digital: os dados alimentam modelos, permitem comparar o estado digital com sua contraparte física e sustentam análises e recomendações [ITU-T, 2022b, ETSI, 2022]. A exigência correta, entretanto, não é reproduzir indiscriminadamente todo o estado da rede, mas sim dispor de observabilidade suficiente e rastreável para o caso de uso, dentro de um atraso e de um custo aceitáveis.

A RFC 9232 ajuda a evitar uma visão estreita da telemetria como mera leitura de contadores. Ela descreve um fluxo de processamento (*pipeline*) que envolve geração, exportação, coleta, análise e consumo de dados provenientes dos planos de gestão, controle e encaminhamento. A sondagem (*polling*) periódica, notificações orientadas a eventos e *streaming* baseado em modelos oferecem compromissos diferentes. A sondagem é adequada para variáveis lentas e para inventários, mas pode ocultar fenômenos que ocorrem em intervalos de tempo inferiores ao da coleta. As notificações reduzem a latência para eventos discretos. Já os *streamings* permitem acompanhar séries de alta frequência, ao custo de um maior volume, de um maior processamento e de uma maior complexidade de correlação. Nenhum mecanismo elimina os demais; a arquitetura deve combinar fontes conforme a dinâmica do fenômeno observado [IETF, 2022a].

A instrumentação também possui custo e pode produzir efeito de observação. Coletas agressivas consomem CPU, memória, largura de banda e capacidade dos coletores. Em situações extremas, o tráfego de telemetria pode competir com o tráfego que se pretende proteger. Por isso, filtros, agregação, assinaturas seletivas, isolamento e políticas de retenção são parte do desenho do NDT. A frequência máxima tecnicamente viável nem sempre é a frequência operacionalmente desejável. O planejamento de longo prazo pode trabalhar com agregações de minutos ou horas, enquanto a detecção de degradação e a

garantia (*assurance*) de acordos de nível de serviço (*Service Level Agreement* – SLA) podem exigir atualizações sensíveis ao tempo e tratamento orientado a eventos.

A sincronização apresenta o mesmo caráter relativo. Um NDT possui atraso inerente entre a ocorrência na rede, a exportação da medição, o processamento e a atualização do estado digital. Esse atraso deve ser conhecido e compatível com a ação considerada. Correlacionar uma mudança de roteamento com perda ou aumento de latência exige relógios e janelas temporais consistentes. Já uma análise de expansão de capacidade tolera defasagens maiores. Assim, “tempo real” não deve ser tratado como propriedade binária. O requisito útil é a atualidade suficiente para que o modelo, o cenário e a decisão se refiram ao mesmo estado operacional.

4.7.2. Observabilidade Incompleta e Estado Oculto

Mesmo com uma arquitetura moderna de telemetria, grande parte do estado necessário ao planejamento e à engenharia de tráfego não é observada diretamente. Contadores de interface informam carga por enlace, registros IPFIX descrevem amostras ou agregações de fluxos nos pontos instrumentados, dados de roteamento explicam parte do caminho provável e sondas ativas observam o desempenho entre pares selecionados. Cada fonte responde a uma pergunta distinta. A matriz de tráfego origem-destino, fundamental para avaliar cenários, raramente aparece com completude e granularidade uniforme em toda a rede. O problema passa, portanto, de uma simples coleta para uma inferência de estado.

Considere o vetor $\mathbf{y}$ de cargas observadas nos enlaces, a matriz de roteamento $\mathbf{A}$ e o vetor $\mathbf{x}$ de demandas Origem-Destino (OD). Em forma simplificada, a relação pode ser escrita como $\mathbf{y} = \mathbf{Ax}$. Em redes reais, o número de pares OD costuma ser muito maior do que o número de enlaces medidos. O sistema é subdeterminado: várias matrizes $\mathbf{x}$ podem produzir as mesmas cargas $\mathbf{y}$. A ausência de solução única não é uma falha do algoritmo, mas uma propriedade do problema inverso.

O TomoGravity, proposto por Zhang et al., combina as restrições impostas pelas cargas e pelo roteamento com uma estrutura de gravidade para a demanda. A técnica foi importante porque tornou viável estimar matrizes de grande porte a partir de dados disponíveis em Operadoras de Serviço de Telecomunicações. Entretanto, a solução permanece condicionada às hipóteses do modelo. As mudanças estruturais na composição da demanda podem introduzir viés e a consistência com as cargas não garante que cada par OD tenha sido estimado corretamente. O resultado deve ser usado como estimativa com incerteza, e não como reconstrução exata de um estado oculto [Zhang et al., 2003b].

Soule et al. mostram que a dicotomia entre medir tudo e inferir tudo é inadequada. Medições diretas de uma parcela dos fluxos podem calibrar o modelo e reduzir o erro, enquanto a inferência completa as regiões não instrumentadas. O benefício depende dos fluxos medidos, do padrão de tráfego, da estabilidade da matriz de roteamento e do método empregado, não existindo um percentual universal de medição adicional que garanta um erro determinado. Para o NDT, a consequência arquitetural é clara, uma vez que as estimativas devem conter proveniência, janela temporal, método, cobertura e um indicador de confiança. Os dados faltantes não desaparecem porque foram imputados; tornam-se parte da incerteza do estado digital [Soule et al., 2005].

A reconciliação entre fontes heterogêneas é, assim, uma função central. Divergências entre contadores, fluxos, políticas, estado de rotas e sondas podem refletir amostragem, perda de coleta, diferença de relógio, encapsulamento, tráfego de controle ou uma alteração real na rede. Antes de alimentar um mecanismo prescritivo, o NDT precisa detectar essas inconsistências, estimar sua magnitude e, quando necessário, recusar ou degradar a confiança da análise. A operação segura não exige certeza absoluta, mas exige que o sistema saiba quando sua visão é insuficiente [IETF, 2009, IETF, 2022a, Soule et al., 2005].

4.7.3. Demanda Latente e Reprimida

Há uma segunda forma de estado oculto que não é resolvida apenas com uma matriz OD melhor estimada. Contadores e registros de fluxo descrevem o tráfego transportado nos pontos de observação. Em condições normais, esse sinal pode se aproximar da carga oferecida. Sob congestionamento, porém, perdas, filas, *backoff* do TCP, modelagem de tráfego (*shaping*), policiamento (*policing*) ou controle de admissão limitam o volume que efetivamente atravessa a rede. O tráfego medido pode estabilizar justamente porque a infraestrutura já não consegue entregar mais. O planejamento da capacidade apenas a partir desse platô corre o risco de interpretar a saturação como ausência de crescimento.

É útil distinguir três conceitos. A demanda oferecida é o volume que as fontes tentariam enviar sob as condições da aplicação. O tráfego transportado é a parcela que percorre a rede e aparece nos contadores e registros. Já a demanda reprimida é a parcela da intenção de tráfego que deixa de se materializar ou de ser entregue por causa de restrições da rede ou de políticas. A SGCRN, por exemplo, aprende a próxima matriz observada a partir do histórico disponível. Ela não recebe rótulos sobre o tráfego que teria surgido na ausência do gargalo e, portanto, não deve ser descrita como estimadora de “demanda real” [Zhao et al., 2020].

A inferência de demanda reprimida é essencialmente contrafactual: quanto tráfego teria sido oferecido se a capacidade, a política ou a condição de congestionamento fosse diferente? Responder exige combinar múltiplos sinais, como perda, tempo de ida e volta (*Round Trip Time* – RTT), retransmissões, estado de filas, respostas da aplicação, limitações configuradas e mudanças posteriores à remoção do gargalo. Mesmo assim, a relação entre esses sinais e a intenção do usuário depende do protocolo e do comportamento da aplicação. O corpus utilizado nesta seção não contém um método quantitativo geral e validado para reconstruir essa demanda; o tema deve ser apresentado como um desafio aberto, e não como uma capacidade já disponível do NDT.

Para o planejamento da capacidade, a implicação é direta. Uma predição baseada apenas no tráfego transportado pode subestimar a capacidade necessária nos pontos já saturados. Para a engenharia de tráfego, a demanda reprimida também altera a avaliação de alternativas. A liberação de um caminho pode gerar tráfego adicional e alterar o equilíbrio previsto. Um NDT operacional deveria, portanto, distinguir medições, estimativas e cenários contrafactuais, evitando tratar todos como o mesmo tipo de dado. Essa separação prepara os desafios seguintes: saber se um modelo permanece válido diante de novas topologias e hardware, se consegue operar na janela temporal requerida e como suas recomendações são materializadas com segurança.

4.7.4. Fidelidade, Generalização e Simulado/Real

A generalização de um modelo de rede não é uma propriedade única. Um modelo pode prever novos “*snapshots*” na mesma topologia e falhar quando o número de nós muda. Um modelo pode ainda aceitar grafos maiores e perder precisão diante de outro hardware ou pode manter desempenho com novas matrizes de tráfego, mas não com protocolos e políticas ausentes do treinamento. Por isso, toda afirmação de generalização precisa declarar o eixo transferido: dados, escala, topologia, distribuição de tráfego, hardware, protocolo ou domínio.

Tabela 4.8. Dimensões distintas de generalização.

Dimensão	Pergunta operacional	Evidência necessária
Novos dados	O modelo funciona em novos períodos da mesma rede?	Teste cronológico e detecção de <i>drift</i> .
Escala	Funciona com mais nós, fluxos ou filas?	Grafos maiores com distribuição controlada.
Topologia	Aceita conectividade não vista?	Topologias distintas, não apenas mais nós.
Hardware/protocolo	Transfere entre equipamentos, schedulers, UDP/TCP?	<i>Testbed</i> e adaptação ao domínio-alvo.
Domínio	Transfere de simulação para rede física ou entre operadores?	Teste cruzado, <i>shadow mode</i> e validação.

A família RouteNet oferece exemplos úteis da distinção de dimensões mostrada na tabela 4.8 O RouteNet-Fermi foi treinado em topologias pequenas e avaliado em redes sintéticas de até 300 nós, alcançando erro percentual absoluto médio de 6,24% no conjunto de escala. O resultado demonstra que o compartilhamento de parâmetros e a representação relacional permitem aplicar o modelo a grafos maiores nas condições experimentais avaliadas. Não demonstra invariância em relação a qualquer topologia, tráfego ou política. Em outra avaliação, usando tráfego real da GÉANT, o modelo alcançou MAPE de 5,67% somente após ajuste fino com 200 amostras do domínio-alvo. A necessidade de adaptação é parte do resultado, não uma falha a ser omitida [Ferriol-Galmés et al., 2023].

O RouteNet-Gauss separa ainda mais claramente a fidelidade física e a escala. Seu *ground truth* físico foi produzido em um *testbed* com até oito roteadores comerciais, aproximando o modelo de *buffers*, filas e temporização do hardware observado. A avaliação com redes de até 110 nós, porém, foi realizada em amostras simuladas. Assim, o estudo fornece duas evidências complementares: maior fidelidade no *testbed* menor e transferência de escala em grafos simulados maiores. Combiná-las como se existisse um *testbed* físico de 110 roteadores seria uma extrapolação indevida [Güemes-Palau et al., 2026].

A mesma cautela vale para a rede de acesso por rádio, núcleo de rede fixa e móvel e fatiamento de rede. Treinar e testar em um canal simulado, usar *traces* coletados de uma rede, executar um modelo em *testbed* e atuar em uma rede comercial representam níveis de evidência diferentes. Dados reais reduzem parte da diferença de distribuição, mas não comprovam que o modelo controlou a infraestrutura em produção. Além disso, o aprendizado por reforço introduz um risco específico: explorar ações online em um sistema ativo

pode levar a estados degradados antes que a política convirja. O NDT deve funcionar como "sandbox" para avaliação *offline* e, na transição operacional, apoiar *shadow mode*, implementação gradual de alterações (*canary tests*), limites de autoridade e comparação com a decisão corrente.

Ajuste fino, adaptação de domínio e treinamento em múltiplos ambientes reduzem a lacuna da realidade (*reality gap*), mas não a eliminam. Mudanças de fornecedor, versão de software, agendador, protocolo ou perfil de tráfego podem alterar as relações aprendidas. Um mecanismo operacional precisa detectar quando a entrada está fora do domínio conhecido, reduzir a confiança e impedir que a saída seja tratada como uma recomendação equivalente às condições validadas. O objetivo não é exigir generalização universal, e sim delimitar e monitorar o domínio no qual o modelo permanece confiável.

4.7.5. Escala, Tempo Real e Custo Computacional

A escala de um NDT não pode ser avaliada por um único número de latência. Treinamento, inferência neural, simulação e otimização resolvem tarefas diferentes e ocupam janelas temporais distintas. O treinamento pode ocorrer ao longo de horas ou dias, a inferência pode ser executada a cada novo cenário, um refinador matemático pode consumir tempo adicional e a simulação detalhada pode servir apenas para gerar dados ou validar casos selecionados. Comparar esses tempos sem informar o hardware, a topologia, o número de entidades e o *baseline* dá a impressão de ganhos universais, quando há resultados específicos.

O RouteNet-Gauss ilustra o ganho e o limite de um modelo substituto (*surrogate*). No cenário equivalente a 50 milhões de pacotes, o modelo realizou a inferência em 2,4 s, aproximadamente 488 vezes mais rápido que o DeepOnet (*DONS*). Esse *speedup* pertence ao cenário, à implementação e ao simulador usados como *baseline*, não constituindo uma constante da arquitetura. Na avaliação de escala simulada, o tempo aumentou em torno de 5,4 ms por nó adicional. O dado mostra que o custo continua dependente do tamanho do grafo, embora cresça muito menos que uma simulação pacote-a-pacote nas condições estadas [Güemes-Palau et al., 2026].

O TEAL fornece um exemplo prescritivo em larga escala na WAN (*Wide Area Network*). Nos experimentos reportados, o *pipeline* calculou alocações em média em 0,95 s na topologia Kdl, com 754 nós e 1.790 enlases, e em 0,97 s na ASN, com 1.739 nós e 8.558 enlases. Na ASN, as acelerações (*speedups*) variaram conforme o comparador: 394 vezes em relação ao NCFlow, 625 vezes ao POP e 197 vezes ao LP-top; o LP-all chegou a aproximadamente 5,5 horas. Esses valores sustentam a possibilidade de acelerar a decisão *offline* em topologias grandes, mas não devem ser convertidos em garantia de execução subsegundo em qualquer rede ou hardware [Xu et al., 2023].

A inferência rápida também não elimina o custo de manutenção do modelo. O TEAL é treinado por topologia e objetivo. O treinamento inicial pode levar cerca de uma semana, enquanto os retreinamentos reportados permanecem na ordem de 6 a 10 horas. Mudanças permanentes de enlases, candidatos de caminho ou distribuição de tráfego podem exigir nova preparação. A distinção é operacionalmente importante: uma política pode responder rapidamente a matrizes inéditas na mesma topologia e, ao mesmo tempo, tornar-se inadequada após uma alteração estrutural que a coloque fora do domínio trei-

nado.

Aceleração por GPU, matrizes esparsas, processamento distribuído e divisão borda-nuvem (*edge-cloud*) podem ampliar a escala, mas introduzem compromissos em termos de custo, energia, disponibilidade e transferência de dados. O projeto deve partir da frequência real de decisão: o planejamento de capacidade (*Capacity Planning*) não precisa compartilhar a mesma infraestrutura de baixa latência usada por garantia (*assurance*); a TE táctica pode privilegiar inferência rápida e validação restrita; o treinamento pode permanecer centralizado e assíncrono. A meta não é executar todos os componentes em tempo real, mas posicionar cada etapa na janela em que seu resultado ainda seja útil e seguro. O desafio seguinte é transformar essa recomendação em estado operacional, sem violar restrições nem perder a capacidade de reversão.

4.7.6. Atuação por Controladores, Orquestradores e Roteamento por Segmentos

Um modelo analítico não altera a rede apenas porque produziu uma previsão ou uma solução de otimização. Entre a saída do modelo e a mudança de estado da infraestrutura existe uma fronteira operacional que precisa ser explícita. SGCRN, RouteNet, TEAL e outros componentes podem estimar demanda, avaliar cenários ou recomendar alocações. A aplicação, porém, depende de validação, autorização e de mecanismos capazes de traduzir a decisão em políticas e configurações. Essa separação reduz o risco de tratar uma inferência estatística como um comando diretamente executável e preserva a responsabilidade do sistema de controle [ETSI, 2022, IETF, 2024].

A primeira etapa é validar a recomendação. O NDT pode verificar capacidades, políticas, restrições de SLA, resiliência e consistência com o estado conhecido da rede. Quando a saída neural não satisfaz integralmente restrições rígidas, ela pode ser projetada para uma região admissível ou refinada por um *solver*. No TEAL, por exemplo, a solução aprendida funciona como um “*warm start*” para o ADMM, reduzindo violações de capacidade e melhorando a qualidade da alocação. O refinamento não deve ser descrito como uma garantia universal de factibilidade ou de ótimo global, ele é uma camada explícita de correção antes da atuação [Xu et al., 2023].

Após a validação, controladores, orquestradores ou um elemento de computação de caminho (*Path Computation Element* – PCE) materializam a decisão. A RFC 8664 define extensões do PCEP (*Path Computation Element Communication Protocol*) para que uma sequência de segmentos calculada por um PCE seja comunicada ao PCC (*Path Computation Client*) e convertida na representação de encaminhamento correspondente. O PCEP é, portanto, um mecanismo de controle entre o PCE e o PCC. Ele não deve ser confundido com a distribuição de políticas de roteamento por segmentos (*SR Policies*), por BGP (*Border Gateway Protocol*) ou por configuração direta; são caminhos configurados distintamente ainda que possam participar da mesma arquitetura operacional [IETF, 2019].

A RFC 9256 organiza uma *SR Policy* em torno do roteador inicial (*headend*), da cor que identifica os enlaces (*color*) e do roteador final (*endpoint*). A política contém caminhos candidatos (*candidate paths*), e cada caminho candidato pode referenciar uma ou mais listas de segmentos (*segment lists*). O *headend* seleciona o caminho válido conforme preferência, disponibilidade e política local e aplica a lista de segmentos no plano

de dados de roteamento por segmentos (*data plane SR*). Essa abstração é adequada para executar uma recomendação de TE porque mantém a inteligência de seleção na borda do domínio e evita exigir estado por fluxo em todos os nós do núcleo. Contudo, o *Segment Routing* e as *SR Políticas* não produzem a decisão: elas fornecem um substrato programável para materializá-la [IETF, 2022b, IETF, 2018].

A cadeia completa não termina na instalação. Telemetria e garantia de serviço (*service assurance*) precisam verificar se a alteração produziu os efeitos previstos, se surgiram degradações não previstas e se o estado observado permanece compatível com o cenário validado. A reversão ao estado anterior (*rollback*) deve ser planejada antes da mudança, com o estado anterior conhecido, política de reversão, critérios de acionamento e autoridade definida. Sem esse fechamento, o NDT funciona apenas como um laboratório de análise. O NDT pode participar de um *closed loop* governado, cujo grau de autonomia depende do risco e do escopo operacional [IETF, 2023, ETSI, 2022].



Figura 4.11. Cadeia de atuação governada em um NDT. A cadeia representa uma composição possível entre telemetria, estado do NDT, análise, validação, controle, atuação, garantia e reversão ao estado anterior. As etapas e o grau de automação variam conforme o domínio e o risco operacional.

4.7.7. Explicabilidade, Segurança e Governança

Explicabilidade, confiança, validação e governança respondem a perguntas diferentes. A explicabilidade procura tornar compreensível por que o modelo produziu determinada saída. A confiança procura estimar quando a previsão pode estar fora do domínio conhecido. Já a validação verifica restrições técnicas e políticas. A segurança protege dados, modelos e interfaces. Por fim, a governança define quem pode autorizar, aplicar, auditar e reverter uma mudança. Agrupar essas funções sob o rótulo genérico de “IA confiável” oculta controles que precisam ser implementados separadamente. O trabalho

sobre conhecimento potencializado (*knowledge-powered*) XAI para automação rumo ao 6G propõe combinar resultados aprendidos com conhecimento do domínio para produzir justificativas mais úteis ao operador. Em redes, uma explicação operacional precisa relacionar a recomendação a entidades verificáveis (demanda, caminho, enlace, capacidade, fila, política e SLA) e não apenas exibir pesos internos da rede neural. Pesos de atenção (*Attention weights*) e incorporações (*embeddings*) podem indicar quais relações influenciaram a representação, mas não demonstram causalidade, correção ou segurança. A explicação é uma evidência para revisão; não é uma prova formal da decisão [Wu et al., 2022].

A confiança da previsão também não deve ser confundida com explicação. Um modelo pode fornecer uma justificativa plausível e ainda operar fora da distribuição de treinamento. O NDT precisa observar *drift*, cobertura dos dados, consistência temporal e distância em relação às condições validadas. Quando a confiança cai, a arquitetura pode reduzir o grau de autoridade, solicitar uma nova análise, acionar um *solver* determinístico ou manter a configuração atual. O cerne desta seção não sustenta a apresentação detalhada de um único método de quantificação de incerteza. Por isso, a confiança aparece como um requisito operacional e não como uma capacidade já resolvida.

A RFC 9417 organiza a garantia de serviço (*service assurance*) por meio de grafos que representam serviços, subserviços, dependências, sintomas e estados de saúde. Essa decomposição permite correlacionar uma degradação fim-a-fim com componentes associados e acompanhar o efeito das mudanças na configuração. Ela favorece a localização e a triagem, mas não garante a causa raiz (*Root Cause Analysis*) de forma automática. O mesmo sintoma pode resultar de causas diferentes. Sem sincronização temporal, sem a versão da topologia e sem o contexto de mudança, a correlação pode apontar para o estado errado [IETF, 2023].

A segurança começa pela integridade do *pipeline* de dados. Telemetria falsa, atrasada ou retirada de contexto pode induzir uma recomendação aparentemente coerente e operacionalmente perigosa. Autenticação, autorização, proteção das interfaces, proveniência, controle de acesso aos modelos e registro de alterações são requisitos do *closed loop*. Ataques adversos específicos contra as GNNs e mecanismos de envenenamento (*poisoning*) permanecem relevantes, mas exigem tratamento bibliográfico próprio. Nesta seção, a discussão permanece no nível arquitetural de proteção de dados, modelos e canais de atuação.

A governança transforma esses mecanismos de segurança em regras de operação. O ETSI ZSM diferencia progressão de automação para autonomia e permite distribuir *closed loops* entre domínios e funções de gestão. Essa distribuição exige a delimitação de objetivos, de escopo, de precedência e de autoridade para evitar que dois *loops* atuem sobre o mesmo recurso com metas incompatíveis. As operações de baixo risco podem ser aprovadas automaticamente dentro de barreiras de segurança (*guardrails*) e mudanças de maior impacto podem exigir a intervenção humana (*human-in-the-loop*). A presença de aprovação humana não é sinônimo de governança completa, assim como a autonomia não significa ausência de responsabilidade ou de rastreabilidade.

Uma política de governança deve registrar a versão do modelo, os dados utilizados, o estado da rede, a recomendação, as validações executadas, a autoridade responsável, a mudança aplicada e o resultado observado. O *shadow mode* e a estratégia de

mitigação de riscos (*canary deployment*) permitem a introdução gradual, comparando a recomendação à operação atual e limitando o alcance inicial. A reversão ao estado anterior (*rollback*) completa o ciclo ao tornar reversível uma decisão que se mostrou inadequada. Esses mecanismos não eliminam falhas, mas reduzem a exposição e produzem evidências para a melhoria contínua do gêmeo digital e de seus modelos. A Tabela 4.9 ilustra as condições de habilitação da adoção de NDTs e GNNs.

Tabela 4.9. Condições para operacionalização de NDTs e GNNs.

Desafio	Impacto	Habilitador	Limite em aberto
Dados e sincronização	Twin desatualizado ou inconsistente	Telemetria seletiva, proveniência e janelas adequadas	Custo e correlação multi-domínio
Estado oculto	Decisões baseadas em visão parcial	Medição, inferência e confiança	Problema inverso e viés
Demanda reprimida	Subdimensionamento em pontos saturados	Sinais indiretos e análise contrafactual	Ausência de estimador geral validado
Generalização	Queda de precisão fora do treino	<i>Fine-tuning</i> , <i>domain adaptation</i> , <i>shadow/canary</i>	<i>Cross-vendor</i> , <i>cross-protocol</i> e <i>sim-to-real</i>
Escala e tempo	Resultado chega tarde ou custa demais	<i>Surrogates</i> , aceleração e separação de janelas	Retreinamento após mudanças estruturais
Atuação	Recomendação viola restrições ou produz impacto	Validação, PCE/PCEP, SR Policy, <i>assurance</i> e <i>rollback</i>	<i>Closed loops</i> seguros em produção
XAI e governança	Baixa confiança, risco e falta de auditoria	Explicação operacional, <i>guardrails</i> , autoridade e logs	Causalidade, segurança adversarial e conflitos entre loops

Os desafios desta seção mostram que a adoção de NDTs e GNNs depende menos de uma arquitetura neural isolada que da engenharia do sistema completo. Dados precisam ser suficientes e temporalmente coerentes; estimativas devem carregar incerteza; generalização deve ser declarada por dimensão; treinamento, inferência e atuação precisam caber em janelas operacionais distintas; e recomendações devem atravessar validação, controle, garantia de serviço (*assurance*) e reversão ao estado anterior (*rollback*). A transição de modelos experimentais para operação confiável é, portanto, um problema multidisciplinar que combina redes, inferência, otimização, sistemas distribuídos, segurança e governança. Persistem desafios em demanda reprimida, portabilidade, adaptação, coordenação entre *closed loops* e segurança, incertezas que o NDT deve tornar observáveis e governáveis.

4.8. Relação com Redes Autônomas

As seções anteriores apresentaram modelos capazes de prever demanda, estimar desempenho, recomendar alocações e validar ações antes de sua aplicação. Esses componentes tornam a automação mais informada, mas não constituem, isoladamente, uma rede autônoma (*Autonomous Networking*). As redes autônomas emergem da integração entre

observabilidade, modelos, decisão, atuação e garantia *assurance*, sob a influência de objetivos, limites de autoridade e mecanismos de coordenação. Esta seção apresenta o desfecho do minicurso, situando a Rede Neural Recorrente Convolucional Espaço-Temporal em Grafos (*Spatiotemporal Graph Convolutional Recurrent Network* – SGCRN), RouteNet, TEAL e o Gêmeo Digital de Rede (*Network Digital Twin* – NDT) em duas referências complementares: o ETSI ZSM, voltado à arquitetura e aos habilitadores (*enablers*), de ciclos fechados (*closed loops*), e o TM Forum, voltado à progressão e à avaliação da maturidade por cenário.

4.8.1. Automação, Automação de Ciclo Fechado e Autonomia

A automação descreve a execução de tarefas por mecanismos previamente definidos, como *scripts*, políticas ou fluxos orquestrados. Um ciclo fechado (*closed loop*) introduz realimentação: observa o sistema, analisa seu estado, decide uma resposta, executa a ação e verifica o resultado. Essa estrutura pode preservar um objetivo sem intervenção em cada passo, mas ainda requer a operação dentro de regras, modelos e limites estabelecidos externamente. Portanto, a automação em *closed-loop automation* não implica autonomia plena nem ausência de supervisão humana. O grupo Gerenciamento de Rede e Serviços sem intervenção humana ou Gerenciamento de redes Zero-Touch do ETSI (*Group Specification for Zero-touch network and Service Management*), ETSI GS ZSM 009-1, prevê governança do ciclo de vida, políticas, logs, pausa da execução, delegação e escalonamento, evidenciando que laços automatizados continuam sujeitos a controle operacional [ETSI, 2021]. Já o ETSI GR ZSM 021 caracteriza redes autônomas como plataformas capazes de perceber o ambiente, aprender e adaptar seu comportamento com pouca ou nenhuma entrada humana. O documento discute “autoconsciência” (*self-awareness*) e “tomada de decisão” (*choice-making*) como capacidades adicionais para evoluir de respostas predominantemente reativas para comportamento proativo. Essas funções não devem ser tratadas como um interruptor binário entre automação e autonomia, mas como parte de uma progressão: quanto maior a incerteza, a variedade de objetivos e a necessidade de adaptação, maior a exigência de conhecimento, aprendizagem, avaliação de risco e capacidade de escalar a decisão para outra entidade ou para um operador [ETSI, 2026].

4.8.2. Integração dos Componentes no Ciclo Fechado

Neste minicurso, o ciclo funcional pode ser expresso como observar, analisar, decidir, executar e verificar. A telemetria e o estado mantido pelo NDT apoiam a observação. A SGCRN antecipa a matriz OD (origem-destino) observada; o RouteNet e simuladores estimam consequências e os KPIs; o TEAL, os *solvers* e as regras produzem alternativas ou alocações candidatas. Os controladores, PCEs e orquestradores traduzem decisões autorizadas em políticas ou configurações, enquanto a garantia de serviços (*service assurance*) compara o resultado observado com o objetivo e pode recomendar manutenção, ajuste ou reversão (*rollback*). Cada componente executa uma função limitada, sendo que nenhum delas equivale à arquitetura completa das redes autônomas.

Essa relação entre as ferramentas não forma um *pipeline* obrigatório. O RouteNet pode avaliar alternativas antes ou depois de um otimizador. O NDT pode limitar-se à análise ou incluir funções que conduzam a decisões sobre a atuação. Por sua vez, uma re-

comendação pode ser rejeitada por um verificador. Por fim, um *closed loop* pode terminar em informação para um operador, sem modificar a rede. A composição depende do caso de uso, do risco, do domínio e da autoridade concedida. A autonomia operacional resulta menos da quantidade de IA empregada do que da capacidade de coordenar essas funções, preservar objetivos e conter efeitos indesejados. As relações são modulares e o nível de autonomia é avaliado por cenário conforme mostrado na Figura 4.12.

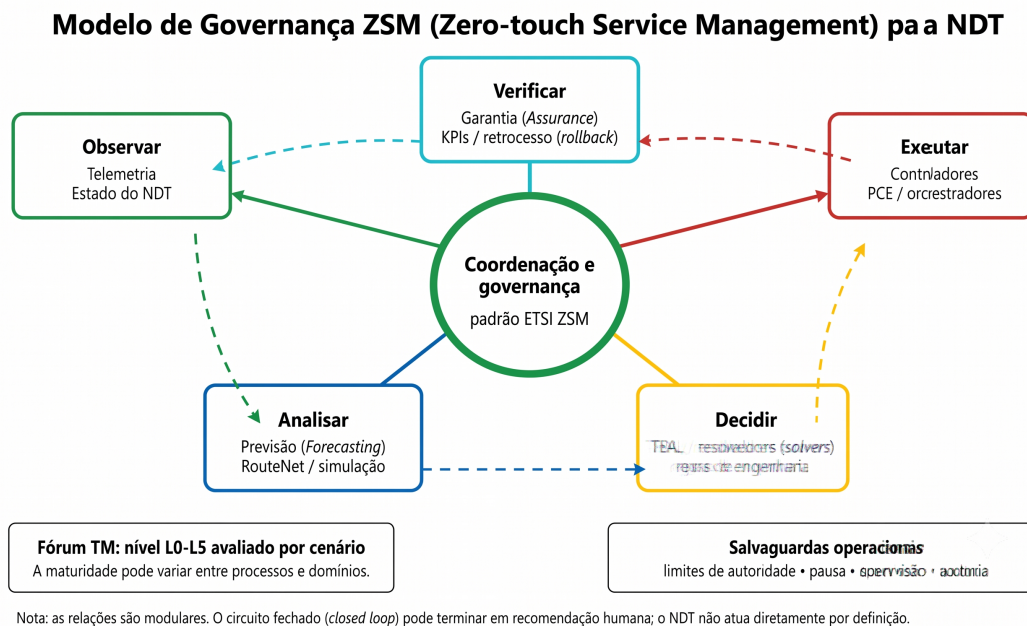


Figura 4.12. Composição funcional das redes autônomas (*Autonomous Networking*)

4.8.3. ETSI ZSM: Arquitetura, NDT e Coordenação de Closed Loops

O ZSM organiza funções de gestão como serviços de gerenciamento (*management services*) oferecidos em domínios de gerenciamento (*management domains*) e no domínio de gestão de serviço fim a fim. Essa separação de gerenciamento de domínios permite que dados sejam coletados em um domínio, e análise de dados (*analytics*) seja executada em outro, e que ações ocorram onde os recursos são controlados pelo gerenciador. O NDT pode participar como serviço analítico, fornecendo previsão e avaliação de cenários, ou integrar funções adicionais que conduzam a decisões em direção à atuação. O ETSI GR ZSM 015 denomina essas perspectivas como *Analyzing NDT* e *Controlling NDT*. A distinção evita assumir que todo gêmeo digital é um controlador e, ao mesmo tempo, reconhece arquiteturas nas quais o gêmeo participa de um *closed loop* mais amplo [ETSI, 2019b, ETSI, 2022, ETSI, 2024].

Quando vários *closed loops* coexistem, o desafio deixa de ser apenas executar cada um corretamente. O *Closed Loop Governance* do ETSI GS ZSM 009–1 gerencia modelos, objetivos, políticas, ciclo de vida, estado e pontos de pausa dos *loops*. O *Closed Loop Coordination* identifica interações e apoia a coordenação pré e pós-execução, a avaliação

do impacto, a delegação, o escalonamento e a resolução de conflitos. Esses *enablers* permitem distribuir autonomia entre domínios sem eliminar responsabilidade. O *framework* fornece os mecanismos arquiteturais. O grau de aprendizagem, adaptação e autoridade continua específico de cada caso de uso [ETSI, 2021].

4.8.4. TM Forum: Níveis de Autonomia

O TM Forum complementa a visão arquitetural com uma metodologia de maturidade. Os níveis de redes autônomas (*Autonomous Network Levels*) variam de L0 a L5 e medem a capacidade autônoma em um cenário definido, como gerência de falhas (*fault management*) na RAN ou otimização de qualidade numa rede IP. A avaliação considera processos, subprocessos, tarefas e evidências da divisão de trabalho entre humanos e máquinas. Assim, uma organização pode apresentar maturidade elevada em um processo e permanecer fortemente manual em outro. Não é metodologicamente correto atribuir um único nível indiferenciado à rede inteira [TM Forum, 2023].

Os níveis também não representam simplesmente “mais IA”. Um cenário pode empregar modelos avançados e ainda depender de aprovação manual, de configuração externa ou de *assurance* incompleto. Em sentido inverso, a automação determinística bem governada pode elevar a maturidade de tarefas específicas sem recorrer a aprendizado profundo. O IG1218F orienta a passagem da estratégia à implementação, enquanto o IG1252 padroniza a avaliação do nível. Para o minicurso, a consequência é prática: o SGCNRN, o RouteNet e o TEAL devem ser avaliados pelo papel que desempenham em um cenário operacional completo, e não pelo desempenho isolado do modelo [TM Forum, 2025, TM Forum, 2023].

4.8.5. Governança e Implantação Progressiva

As Redes Baseadas em Intenção (*Intent-Based Networking*) acrescentam uma camada declarativa de objetivos. A RFC 9315 diferencia a intenção (*intent*) de conceitos relacionados: a intenção expressa resultados desejados, enquanto políticas, modelos e configurações participam de sua realização. Traduzir um objetivo em mudança de rede exige decomposição, validação, cumprimento da intenção *fulfillment* e *assurance*. A RFC 9417 reforça que a saúde de um serviço depende de uma visão holística de subserviços, funções e recursos. Dessa forma, verificar o resultado não é uma etapa acessória; é parte constitutiva do *closed loop* [Clemm et al., 2022, IETF, 2023].

A implantação deve progredir conforme o valor e o risco do cenário. Limites de autoridade, logs, pontos de pausa, supervisão proporcional ao impacto e *rollback* permitem ampliar a autonomia sem perder o controle. O modo sombra (*Shadow mode*) e a atualização progressiva (*canary deployment*) são estratégias possíveis para observar recomendações e restringir o alcance inicial, mas não constituem requisitos universais. A autonomia plena permanece uma direção de evolução, não uma propriedade demonstrada pelos estudos apresentados. O resultado mais defensável é a autonomia graduada: componentes especializados integrados por uma arquitetura que sabe quando agir, quando verificar, quando escalar e quando não executar.

4.9. Conclusões

As redes 5G-Advanced e as futuras redes 6G ampliam a heterogeneidade, o dinamismo, a escala e o rigor dos requisitos de desempenho impostos à infraestrutura de telecomunicações. Serviços de banda larga avançada, comunicações críticas, conectividade massiva, aplicações distribuídas nas bordas, fatiamento de redes, virtualização e mobilidade fazem com que capacidade, atraso, *jitter*, perda, disponibilidade e consumo de recursos precisem ser coordenados entre RAN, núcleo, borda/nuvem e transporte. Esses desafios não eliminam as práticas tradicionais de planejamento e operação, mas reduzem a margem de erro de abordagens baseadas apenas em séries históricas, estados agregados, simulações demoradas e ciclos predominantemente manuais.

Este minicurso apresentou os principais conceitos de Gêmeos Digitais de Rede (*Network Digital Twins* – NDTs) como uma arquitetura orientada a casos de uso para representar, sincronizar, experimentar e validar o comportamento de redes e serviços antes de uma eventual atuação na infraestrutura física. Um NDT não é uma GNN, um simulador ou uma réplica integral e permanentemente atualizada de toda a rede. Ele integra dados, modelos, mapeamentos, interfaces, gestão de ciclo de vida e aplicações, permitindo compor os gêmeos (*twins*) de domínio ou de serviço com a fidelidade, a granularidade e a frequência adequadas à decisão apoiada.

Também foram apresentados os conceitos das Redes Neurais de Grafos (*Graph Neural Networks* – GNNs), introduzidas como modelos capazes de explorar explicitamente a topologia e as relações entre entidades físicas, virtuais e de serviço. A partir de grafos formados por roteadores, enlaces, caminhos, filas, células, usuários, funções e recursos computacionais, o *message passing* permite aprender representações orientadas a tarefas de previsão, classificação, modelagem de desempenho e decisão. O panorama multidomínio inclui exemplos de gerenciamento de recursos de rádio, fatiamento, classificação de falhas no núcleo da rede e alocação de recursos nas bordas. A profundidade técnica, entretanto, a demonstração concentra-se no transporte IP, utilizado como eixo para demonstrar a cadeia entre previsão de demanda (*forecasting*), planejamento de capacidade (*Capacity Planning*), modelagem de desempenho e Engenharia de Tráfego (*Traffic Engineering*).

Três estudos de caso delimitam funções complementares. A Rede Neural Recorrente Convolutiva Espaço-Temporal em Grafos (*Spatiotemporal Graph Convolutional Recurrent Network* – SGCRN) combina estrutura espacial e dinâmica temporal para prever a próxima matriz de tráfego observada. A família RouteNet utiliza representações relacionais entre caminhos, fluxos, filas, enlaces e equipamentos para estimar atraso, *jitter*, perda e outros indicadores-chave de desempenho em análises do tipo *what-if*. O TEAL combina FlowGNN, aprendizado multiagente e refinamento pelo Método dos Multiplicadores de Direção Alternada (*Alternating Direction Method of Multipliers* – ADMM) para produzir alocações candidatas de tráfego em escala. Esses modelos não formam obrigatoriamente um *pipeline* e não constituem, isoladamente, um NDT ou uma solução autônoma completa.

O minicurso também examinou limitações que condicionam a adoção operacional: telemetria e sincronização, observabilidade incompleta, inferência de estado, demanda latente ou reprimida, fidelidade, generalização, simulação para realidade (*sim-to-*

real), custo computacional, explicabilidade, segurança e governança. A atuação requer validação de restrições, controladores ou orquestradores, mecanismos como elemento de computação de caminhos (*Path Computation Element – PCE / Path Computation Element Protocol – PCEP*) e políticas de roteamento por segmentos (*SR Policies*), garantia (*assurance*) contínua e reversão de estado (*rollback*). Por fim, ETSI ZSM e TM Forum são utilizados para posicionar esses componentes em ciclos fechados (*closed loops*) progressivamente mais autônomos, avaliados por cenário e submetidos a limites de autoridade. A mensagem central é que a evolução para redes autônomas resulta da integração governada entre conhecimento de redes, dados, modelos, otimização e automação, e não da substituição da engenharia por um algoritmo isolado.

Referências

- [3GPP, 2025a] 3GPP (2025a). Management and orchestration: Concepts, use cases and requirements. Relatório Técnico TS 28.530 V19.0.0, 3rd Generation Partnership Project.
- [3GPP, 2025b] 3GPP (2025b). Release 18 description: Summary of rel-18 work items. Relatório Técnico TR 21.918 V18.0.0, 3rd Generation Partnership Project.
- [3GPP, 2025c] 3GPP (2025c). Study on management aspects of network digital twin. Relatório Técnico TR 28.915 V19.0.0, 3rd Generation Partnership Project.
- [3GPP, 2026a] 3GPP (2026a). Management and orchestration; management aspects of network digital twins. Relatório Técnico TS 28.561 V20.1.0, 3rd Generation Partnership Project.
- [3GPP, 2026b] 3GPP (2026b). Release 19 description: Summary of rel-19 work items. Relatório Técnico TR 21.919 V19.0.0, 3rd Generation Partnership Project.
- [Abode et al., 2022] Abode, D., Adeogun, R. e Berardinelli, G. (2022). Power control for 6g industrial wireless subnetworks: A graph neural network approach.
- [Almasan et al., 2022] Almasan, P., Ferriol-Galmés, M., Paillisse, J., Suárez-Varela, J., Perino, D., Lopez, D., Pastor Perales, A. A., Harvey, P., Ciavaglia, L., Wong, L., Ram, V., Xiao, S., Shi, X., Cheng, X., Cabellos-Aparicio, A. e Barlet-Ros, P. (2022). Network digital twin: Context, enabling technologies and opportunities. *IEEE Communications Magazine*.
- [Alon e Yahav, 2021] Alon, U. e Yahav, E. (2021). On the bottleneck of graph neural networks and its practical implications. Em *International Conference on Learning Representations*.
- [Battaglia et al., 2018] Battaglia, P. W. et al. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.
- [Bernárdez et al., 2023] Bernárdez, G. et al. (2023). Magnneto: A graph neural network-based multi-agent system for traffic engineering.

- [Bogle et al., 2019] Bogle, J., Bhatia, N., Ghobadi, M., Menache, I., Bjørner, N., Valadarsky, A. e Schapira, M. (2019). Teavar: Striking the right utilization-availability balance in wan traffic engineering. Em *Proceedings of the ACM SIGCOMM 2019 Conference*.
- [Clemm et al., 2022] Clemm, A., Ciavaglia, L., Granville, L. Z. e Tantsura, J. (2022). Intent-based networking - concepts and definitions. Relatório Técnico RFC 9315, RFC Editor.
- [Ding et al., 2024] Ding, M., Guo, Y., Huang, Z., Lin, B. e Luo, H. (2024). Grom: A generalized routing optimization method with graph neural network and deep reinforcement learning. *Journal of Network and Computer Applications*, 229:103927.
- [ETSI, 2019a] ETSI (2019a). Multi-access edge computing (mec): Support for network slicing. Relatório Técnico ETSI GR MEC 024 V2.1.1, European Telecommunications Standards Institute.
- [ETSI, 2019b] ETSI (2019b). Zero-touch network and service management (zsm); reference architecture. Relatório Técnico ETSI GS ZSM 002 V1.1.1, European Telecommunications Standards Institute.
- [ETSI, 2021] ETSI (2021). Zero-touch network and service management (zsm); closed-loop automation; part 1: Enablers. Relatório Técnico GS ZSM 009-1 V1.1.1, ETSI.
- [ETSI, 2022] ETSI (2022). Zero-touch network and service management (zsm); network digital twin. Relatório Técnico ETSI GR ZSM 015 V1.1.1, European Telecommunications Standards Institute.
- [ETSI, 2024] ETSI (2024). Zero-touch network and service management (zsm); network digital twin for enhanced zero-touch network and service management. Relatório Técnico GS ZSM 018 V1.1.1, ETSI.
- [ETSI, 2026] ETSI (2026). Zero-touch network and service management (zsm); study of zsm framework from automation to autonomy. Relatório Técnico ETSI GR ZSM 021 V1.1.1, European Telecommunications Standards Institute.
- [Ferriol-Galmés et al., 2023] Ferriol-Galmés, M., Paillisse, J., Suárez-Varela, J., Rusek, K., Xiao, S., Shi, X., Cheng, X., Barlet-Ros, P. e Cabellos-Aparicio, A. (2023). Routenet-fermi: Network modeling with graph neural networks. *IEEE/ACM Transactions on Networking*, 31(6):3080–3095.
- [Gilmer et al., 2017] Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. e Dahl, G. E. (2017). Neural message passing for quantum chemistry. Em *Proceedings of the 34th International Conference on Machine Learning*, p. 1263–1272.
- [Güemes-Palau et al., 2026] Güemes-Palau, C., Ferriol-Galmés, M., Paillisse-Vilanova, J., López-Brescó, A., Barlet-Ros, P. e Cabellos-Aparicio, A. (2026). Routenet-gauss: Hardware-enhanced network modeling with machine learning. *IEEE Transactions on Networking*, 34:3840–3853. Preprint: arXiv:2501.08848.

- [Hamilton, 2020] Hamilton, W. L. (2020). *Graph Representation Learning*. Morgan & Claypool Publishers.
- [Hu et al., 2020] Hu, Z., Dong, Y., Wang, K. e Sun, Y. (2020). Heterogeneous graph transformer. Em *Proceedings of The Web Conference 2020*, p. 2704–2710.
- [IETF, 2009] IETF (2009). Architecture for ip flow information export. RFC 5470, RFC Editor.
- [IETF, 2018] IETF (2018). Segment routing architecture. RFC 8402, RFC Editor.
- [IETF, 2019] IETF (2019). Path computation element communication protocol (pcep) extensions for segment routing. RFC 8664, RFC Editor.
- [IETF, 2022a] IETF (2022a). Network telemetry framework. RFC 9232, RFC Editor.
- [IETF, 2022b] IETF (2022b). Segment routing policy architecture. RFC 9256, RFC Editor.
- [IETF, 2023] IETF (2023). Service assurance for intent-based networking architecture. RFC 9417, RFC Editor.
- [IETF, 2024] IETF (2024). Overview and principles of internet traffic engineering. RFC 9522, RFC Editor.
- [IRTF NMRG, 2026] IRTF NMRG (2026). Network digital twin (ndt): Concepts and reference architecture. Internet-Draft draft-irtf-nmrg-network-digital-twin-arch-13, Internet Research Task Force. Work in Progress.
- [Isah et al., 2024] Isah, A., Aliyu, I., Shim, J., Ryu, H. e Kim, J. (2024). Beyond 5g network failure classification for network digital twin using graph neural network.
- [ITU-R, 2015] ITU-R (2015). Imt vision: Framework and overall objectives of the future development of imt for 2020 and beyond. Relatório Técnico Recommendation ITU-R M.2083-0, International Telecommunication Union, Radiocommunication Sector.
- [ITU-R, 2023] ITU-R (2023). Framework and overall objectives of the future development of imt for 2030 and beyond. Relatório Técnico Recommendation ITU-R M.2160-0, International Telecommunication Union, Radiocommunication Sector.
- [ITU-T, 2022a] ITU-T (2022a). Digital twin network: Requirements and architecture. Relatório Técnico Recommendation Y.3090, International Telecommunication Union.
- [ITU-T, 2022b] ITU-T (2022b). Digital twin network: Requirements and architecture. Recommendation ITU-T Y.3090, International Telecommunication Union.
- [Kipf e Welling, 2017] Kipf, T. N. e Welling, M. (2017). Semi-supervised classification with graph convolutional networks. Em *International Conference on Learning Representations*.
- [Lin et al., 2023] Lin, X., Kundu, L., Dick, C., Obiodu, E. e Mostak, T. (2023). 6g digital twin networks: From theory to practice. *IEEE Communications Magazine*.

- [Liu et al., 2025] Liu, C., Deng, H., Aggarwal, V., Yang, Y. e Xu, M. (2025). Shooting large-scale traffic engineering by combining deep learning and optimization approach. *Proceedings of the ACM on Networking*, 3.
- [NGMN Alliance, 2023] NGMN Alliance (2023). 6g requirements and design considerations. Relatório Técnico Version 1.0, Next Generation Mobile Networks Alliance.
- [O-RAN nGRG, 2024a] O-RAN nGRG (2024a). Digital twin ran: Key enablers. Relatório Técnico RR-2024-09, O-RAN Alliance nGRG.
- [O-RAN nGRG, 2024b] O-RAN nGRG (2024b). Research report on digital twin ran use cases. Relatório Técnico RR-2024-07, O-RAN Alliance nGRG.
- [Rampášek et al., 2022] Rampášek, L., Galkin, M., Dwivedi, V. P., Luu, A. T., Wolf, G. e Beaini, D. (2022). Recipe for a general, powerful, scalable graph transformer. Em *Advances in Neural Information Processing Systems*, volume 35, p. 14501–14515.
- [Rashid et al., 2024] Rashid, S. M., Aliyu, I., Jeong, I.-K., Um, T.-W. e Kim, J. (2024). Graph neural network for in-network placement of real-time metaverse tasks in next-generation network. Version 1.
- [Rusek et al., 2020] Rusek, K., Suárez-Varela, J., Almasan, P., Barlet-Ros, P. e Cabellos-Aparicio, A. (2020). Routenet: Leveraging graph neural networks for network modeling and optimization in sdn. *IEEE Journal on Selected Areas in Communications*, 38(10):2260–2270.
- [Shao et al., 2021] Shao, Y., Li, R., Hu, B., Wu, Y., Zhao, Z. e Zhang, H. (2021). Graph attention network-based multi-agent reinforcement learning for slicing resource management in dense cellular network.
- [Shen et al., 2021] Shen, Y., Shi, Y., Zhang, J. e Letaief, K. B. (2021). Graph neural networks for scalable radio resource management: Architecture design and theoretical analysis. *IEEE Journal on Selected Areas in Communications*, 39(1):101–115.
- [Soule et al., 2005] Soule, A., Salamatian, K., Taft, N., Emilion, R. e Papagiannaki, K. (2005). Traffic matrices: Balancing measurements, inference and modeling. Em *Proceedings of ACM SIGMETRICS*.
- [TM Forum, 2023] TM Forum (2023). Autonomous network levels evaluation methodology v1.2.0. Relatório Técnico IG1252, TM Forum.
- [TM Forum, 2025] TM Forum (2025). Autonomous networks framework v2.0.0. Relatório Técnico IG1218F, TM Forum.
- [Veličković et al., 2018] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P. e Bengio, Y. (2018). Graph attention networks. Em *International Conference on Learning Representations*.
- [Wu et al., 2022] Wu, Y., Lin, G. e Ge, J. (2022). Knowledge-powered explainable artificial intelligence (xai) for network automation towards 6g.

- [Xu et al., 2023] Xu, Z., Yan, F. Y., Singh, R., Chiu, J. T., Rush, A. M. e Yu, M. (2023). Teal: Learning-accelerated optimization of wan traffic engineering. Em *Proceedings of ACM SIGCOMM 2023*, p. 378–393.
- [Yu et al., 2018] Yu, B., Yin, H. e Zhu, Z. (2018). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. Em *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, p. 3634–3640.
- [Zhang et al., 2003a] Zhang, Y., Roughan, M., Duffield, N. e Greenberg, A. (2003a). Fast accurate computation of large-scale ip traffic matrices from link loads. Em *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*.
- [Zhang et al., 2003b] Zhang, Y., Roughan, M., Duffield, N. e Greenberg, A. (2003b). Fast accurate computation of large-scale ip traffic matrices from link loads. Em *Proceedings of ACM SIGMETRICS*.
- [Zhao et al., 2020] Zhao, J., Qu, H., Zhao, J., Dai, H. e Jiang, D. (2020). Spatiotemporal graph convolutional recurrent networks for traffic matrix prediction. *Transactions on Emerging Telecommunications Technologies*, 31:e4056.
- [Zhou et al., 2025] Zhou, F., Liu, X., Yu, R. e Xue, G. (2025). Traffic engineering in large-scale networks with generalizable graph neural networks.