Projeto de Redes Ciente de Gargalos para Otimização do Treinamento de Modelos Distribuídos de IA

Vitor F. Zanotelli[1], Arthur T. Sampaio[1], Magnos Martinello[1], Jordi Ros-Giralt[2], Giovanni Comarela[1]

[1]Departamento de Informática – Universidade Federal do Espírito Santo (UFES)
Av. Fernando Ferrari, 514 – Goiabeiras, Vitória – ES, 29075-910 – Brasil

[2]Qualcomm Europe, Inc

fz.vitor@gmail.com, arthur.sampaio@edu.ufes.br, magnos@inf.ufes.br, jros@qti.qualcomm.com, gc@inf.ufes.br

Abstract. *The distributed training of AI models significantly increases the demand for efficient communication both within and across data centers. Grounded in Bottleneck Theory (BST), this work aims to design a bottleneck-aware network architecture that minimizes training time while utilizing the least possible bandwidth capacity. We propose a methodology for optimizing network configurations to ensure efficiency and eliminate resource waste. The resulting optimized design serves as a robust performance baseline prior to the adoption of any overprovisioning strategies.*

Resumo. *O treinamento distribuído de modelos de IA impõe uma demanda crescente por comunicação eficiente tanto dentro quanto entre datacenters. Com base na Teoria dos Gargalos (BST), este trabalho visa projetar uma rede ciente dos gargalos, buscando minimizar o tempo total de treinamento utilizando a menor largura de banda possível. Propomos uma metodologia para otimizar as configurações de rede, promovendo eficiência e evitando desperdício de recursos. Esse projeto de rede otimizado estabelece uma linha de base robusta de desempenho, a ser considerada antes da adoção de qualquer estratégia de superprovisionamento.*

1. Introdução

À medida que os modelos de deep learning se tornam mais complexos, cresce também a demanda por recursos computacionais para seu treinamento, com arquiteturas de última geração alcançando bilhões de parâmetros. Modelos recentes, como o DeepSeek R1 e V3, já ultrapassam os 600 bilhões de parâmetros [DeepSeek-AI et al. 2025], enquanto o Grok e o Llama 4 têm, respectivamente, 300 [xAI 2024] e 400 bilhões [Meta AI 2024]. Esse crescimento exponencial torna inviável o treinamento em dispositivos únicos. Como consequência, soluções distribuídas tanto verticais quanto horizontais, tornam-se fundamentais para viabilizar a escalabilidade exigida pelas cargas de trabalho em IA modernas.

Enfrentar esse problema exige um esforço multifatorial envolvendo hardware e software, incluindo a escolha cuidadosa da estratégia de paralelização, das unidades de processamento (GPU e TPU), da topologia da rede e de sua capacidade de banda. Selecionar uma estratégia de paralelismo — como o Paralelismo de Dados (DP) ou Paralelismo de Tensores (TP) — envolve diferentes compromissos de desempenho, escalabilidade

e uso de recursos, uma vez que cada abordagem apresenta vantagens e limitações específicas.

O Paralelismo de dados (DP) replica o modelo inteiro em múltiplos hosts, gerando tráfego de rede composto principalmente por pesos e gradientes para sincronização de parâmetros. O TP, por sua vez, divide os tensores individuais entre múltiplos dispositivos, permitindo que operações matriciais internas sejam paralelizadas em nível mais granular. O DP limita o tamanho do modelo à memória de um único host [Rajbhandari et al. 2020].

O tráfego de comunicação gerado durante o treinamento distribuído pode, por si só, constituir um gargalo crítico, comprometendo o desempenho global do sistema. Para atender às exigentes demandas de largura de banda e sincronização das cargas de trabalho, é necessário adotar uma abordagem fundamentada de provisionamento que seja, ao mesmo tempo, econômica e escalável. Neste contexto, este trabalho propõe o projeto de uma arquitetura de rede ciente dos gargalos, com base na Teoria dos Gargalos [Ros-Giralt et al. 2022] , visando à minimização do tempo total de treinamento utilizando a menor largura de banda possível. Propomos uma metodologia para otimização das configurações de rede, com foco na eficiência e na eliminação de desperdícios de recursos. O design resultante estabelece uma linha de base robusta de desempenho, servindo como referência antes da adoção de estratégias de superprovisionamento.

2. Design de Rede Ciente de Gargalos

Este trabalho utiliza a estrutura teórica adotada na Teoria das Estruturas de Gargalo [Ros-Giralt et al. 2022] (*Theory of Bottleneck Structures*). Definimos um modelo de rede como sendo composto por um conjunto de hosts interconectados por um conjunto de enlaces, de capacidade finita, com a comunicação entre os hosts realizada por meio de fluxos. Um fluxo entre dois hosts percorre um subconjunto de enlaces, definindo um caminho. A rede é regulada por um algoritmo de roteamento que estabelece os caminhos de todos os fluxos. Assumimos que os fluxos de rede transferem a mesma quantidade de dados entre os hosts, como ocorre nos seguintes cenários de padrões de tráfego relevantes: (1) troca de pesos e gradientes, usando operações de comunicação coletiva para DP (por exemplo, All-Gather e All-Reduce), e (2) troca de dados de treino.

Para uma determinada rede, uma solução de planejamento de capacidade é especificada por meio de um design. Um design de rede é definido como a atribuição de um valor de capacidade (em bits por segundo) para cada enlace. O design proposto pode ser implementado por meio de diversas estratégias, como reserva de largura de banda em cada enlace ou utilização de técnicas de fatiamento de rede (network slicing) dentro de infraestruturas de rede já existentes. O objetivo da nossa metodologia é atribuir um design sem desperdício. Ou seja, um design no qual a capacidade atribuída a cada enlace seja totalmente utilizada durante a transmissão de dados. Um design sem desperdícios garante que qualquer melhoria no tempo de finalização dos fluxos ou na taxa de transferência exigiria o aumento da capacidade da rede [Ros-Giralt et al. 2021]. Esse design atinge o melhor desempenho para um custo fixo, sendo útil para projetistas de redes como um ponto de partida das necessidades de capacidade antes de considerar outros parâmetros usuais como falhas de enlace, redundância e requisitos de latência.

Para especificar a comunicação entre hosts, define-se um padrão de tráfego como um mapeamento de pares de hosts e a quantidade de bits sendo transmitida do primeiro

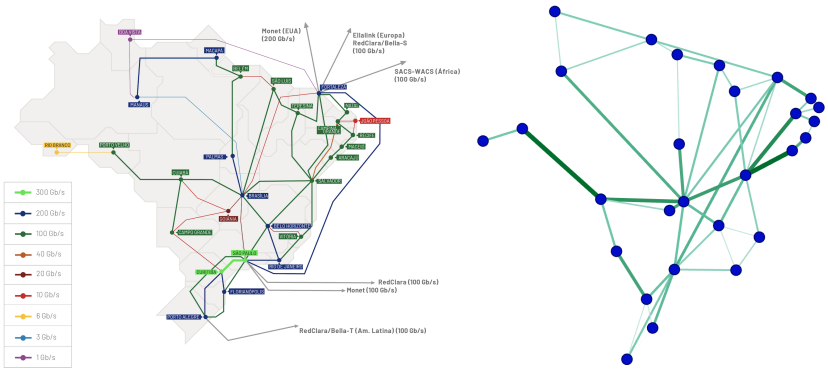

Figura 1. (Esquerda) Topologia atual da Rede Ipê (Fonte: Rede Nacional de Ensino e Pesquisa). (Direita) Alocação de capacidades para a Rede Ipê seguindo o Design Proporcional.

Tabela 1. Tempo de execução e capacidade total utilizada por topologia para uma rodada de transferência de dados.

Design de Rede	Capacidade Total Utilizada (Mbit)	Tempo (s)
Proporcional	2493	8.0
Uniforme (Mínimo)	530	76.3
Uniforme (Médio)	2493	16.2
Uniforme (Máximo)	5054	8.0

para o segundo. Um padrão de tráfego é dito *livre de interferência* se cada fluxo f percorre algum enlace l que não é percorrido por nenhum fluxo que transmita mais bits do que f (ver [Ros-Giralt et al. 2021]). Conforme nossa suposição anterior, no caso de comunicações coletivas aplicadas a estratégias de paralelização como DP e TP, o tráfego é livre de interferência porque todos os fluxos enviam a mesma quantidade de dados.

Para alcançar um design sem desperdícios, é utilizada uma classe de designs conhecida como *designs proporcionais* [Ros-Giralt et al. 2021]. Um design é proporcional se a capacidade de cada enlace for proporcional, com algum fator de escala α, à soma dos tamanhos de todos os fluxos que o percorrem. Para garantir que o design proporcional seja sem desperdícios, utilizamos a propriedade dos padrões de tráfego livres de interferência. Em [Ros-Giralt et al. 2021], os autores provam que, quando um padrão de tráfego é livre de interferência, um design proporcional garante que não há desperdício de largura de banda e que todos os fluxos completam a transmissão simultaneamente. Um design proporcional único passa então a existir, e ao selecionar cuidadosamente o fator de escala, é possível ajustar o nível de desempenho desejado [Ros-Giralt et al. 2021].

2.1. Caso de uso: Rede Ipê

Como exemplo, considere um experimento que consiste no treinamento de um modelo de linguagem com aproximadamente um bilhão de parâmetros utilizando Paralelismo de Dados. Para a topologia, é utilizada a Rede Ipê[1] com um host em cada PoP, a topologia da rede pode ser vista na Figura 1 (esquerda). O algoritmo de roteamento prioriza o caminho mais curto e, quando há múltiplos caminhos com o mesmo comprimento, seleciona aquele com a menor contagem de fluxos atual para alcançar um melhor balanceamento de carga. A quantidade de dados trocados para esse tamanho de modelo está na ordem de 1 GB por

[1]Rede Ipê: `https://www.rnp.br/sistema-rnp/infraestrutura-para-pesquisa/`

fluxo, e utilizamos um fator de escalonamento de $\alpha = 1$. Para obter o resultado de tempo de completude de fluxos é utilizada a ferramenta G2 [Ros-Giralt et al. 2019].

O design proporcional pode ser visto na Figura 1 (direita), onde a espessura de cada enlace é proporcional a banda alocada. A Tabela 1 apresenta a comparação entre a capacidade total utilizada por todos os enlaces e o tempo para a transferência de dados entre o design proporcional e o uniforme, onde todos os enlaces utilizam um valor fixo de banda. O valor fixo utilizado é respectivamente: o mínimo do design proporcional, o valor médio e o maior valor. Enquanto que para alcançar o mesmo desempenho, em tempo, a alocação uniforme utiliza o dobro de banda, no design médio, utilizando a mesma quantidade de banda, dobra-se o tempo para finalizar a transferência. Utilizar o design mínimo reduz o uso de banda para 20%, mas o tempo total para transferência dos dados aumenta em quase 10 vezes. Ajustando o fator de escala α para 20% em um design proporcional utilizaria a mesma quantidade de banda total do design mínimo e ainda assim teria um desempenho maior, terminando em 40 segundos (aproximadamente metade do tempo do caso uniforme). Os resultados evidenciam como a alocação ciente de gargalos permite utilizar a capacidade alocada de forma eficiente, a Figura 1 permite visualizar quais enlaces priorizar para esse experimento.

3. Conclusão e Trabalhos Futuros

Este trabalho auxilia no provisionamento de redes para atender às demandas de cargas de trabalho de IA. Ao fornecer um ponto de partida quantitativo, o trabalho permite que engenheiros de rede projetem infraestruturas com utilização garantida de largura de banda e taxa de transferência a um custo fixo — antes de considerar quaisquer estratégias de superprovisionamento.

Referências

DeepSeek-AI, Liu, A., and et al, B. F. (2025). Deepseek-v3 technical report.

Meta AI (2024). Introducing llama 4: Advancing open, multimodal intelligence. Disponível em: `https://ai.meta.com/blog/llama-4-multimodal-intelligence/`. Acesso em: 01 ago. 2025.

Rajbhandari, S., Rasley, J., Ruwase, O., and He, Y. (2020). Zero: memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press.

Ros-Giralt, J., Amsel, N., Yellamraju, S., Ezick, J., Lethin, R., Jiang, Y., Feng, A., and Tassiulas, L. (2022). A quantitative theory of bottleneck structures for data networks.

Ros-Giralt, J., Amsel, N., Yellamraju, S., Ezick, J., Lethin, R., Jiang, Y., Feng, A., Tassiulas, L., Wu, Z., Teh, M. Y., and Bergman, K. (2021). Designing data center networks using bottleneck structures. SIGCOMM '21.

Ros-Giralt, J., Yellamraju, S., Bohara, A., Lethin, R., Li, J., Lin, Y., Tan, Y., Veeraraghavan, M., Jiang, Y., and Tassiulas, L. (2019). G2: A network optimization framework for high-precision analysis of bottleneck and flow performance. In *INDIS '19*.

xAI (2024). Grok is now open source. Disponível em: `https://x.ai/news/grok-os`. Acesso em: 01 ago. 2025.