HAL
open science

Génération de colonnes pour le problème de routage à délai variable

Nicolas Huin, Jeremie Leguay, Sébastien Martin

To cite this version:

Nicolas Huin, Jeremie Leguay, Sébastien Martin. Génération de colonnes pour le problème de routage à délai variable. ALGOTEL 2020 – 22èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Sep 2020, Lyon, France. hal-02875446

HAL Id: hal-02875446

https://hal.science/hal-02875446v1

Submitted on 19 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Génération de colonnes pour le problème de routage à délai variable

Nicolas Huin et Jérémie Leguay et Sébastien Martin

Huawei Technologies, France Research Center.

Avec la promesse d'un réseau plus performant, la 5G va amplifier l'apparition de services avec des besoins plus stricts pour la latence de bout en bout. Satisfaire des contraintes de délais plus strictes nécessite d'intégrer des modèles d'évolution du délai en fonction de la charge plus réalistes. De nombreuses approches permettent de modéliser le délai d'un réseau dans les cas stochastiques ou déterministes, mais peu de travaux ont intégré ces modèles dans l'optimisation globale du routage. Dans cet article, nous proposons une nouvelle méthode de décomposition qui permet d'inclure ces modèles et nous montrons que nous pouvons satisfaire des contraintes plus fortes qu'avec des modèles simplistes.

Mots-clefs : Programmation linéaire, modèle de délai, génération de colonnes, méthodes de décomposition, 5G.

1 Introduction

The 5th generation of networks is paving the road for latency-sensitive network services to enable a wide-range of time critical applications like factory automation, connected vehicles, and smart grids. All these new services have their own latency requirements; some tolerate a large end-to-end latency while others have tight requirements. Satisfying them all is crucial and requires accurate models of the delay in a network.

While modeling the delay induced by a queuing network is a well studied problem, only a few works integrates advanced latency models in an optimization global routing problem. Fortz *et al* considered the piecewise linear unsplittable multicommodity flow problem [FT00, FGJM17] where the cost of links is inversely proportional to the link utilization, but this model does not explicitly capture the end-to-end latency of each flow. Ben Ameur *et al* [BAO06] considered the Kleinrock function [Kle07] and gave a convex relaxation to compute a lower bound of the (fractional) routing problem with minimum linear cost. Truffot *et al* [TDM10] proposed a linear relaxation of a non-linear and non-convex problem, formulated also using the Kleinrock function, to minimize the maximum end-to-end latency. They solved the integer problem using a branch-and-price algorithm. These works cannot take into account more advanced delay model.

In this paper, we propose a new decomposition model, based on column generation, to solve the multicommodity flow problem with non-constant delay. Formulating the problem usually requires the use of non-linear inequalities: we push non-linearities into the pricing problem which allows us to consider constant models, piece-wise linear models, non-linear models from queuing theory or network calculus and even machine learning oracles. In this paper, we formulate the problem using the classical M/M/1 queuing model, where packet arrivals follow a Poisson process and the service rate of links is exponentially distributed.

2 Problem statement and model

We represent the network as a graph $G = (V,E)$. Each link $e \in E$ is characterized by its cost C_e, its propagation delay λ_e and its bandwidth capacity B_e. Each node $v \in V$ has a bandwidth capacity $B_v = \min \sum_{e\in\delta^+(v)} B_e, \sum_{e\in\delta^-(v)} B_e$ and a delay function $f_v : \mathbb{R}^+ \to \mathbb{R}^+$. We consider a node level delay, which is typically used when all packets have to pass through a unique data pipeline in the devices, but we can easily extend the model to function at link level.

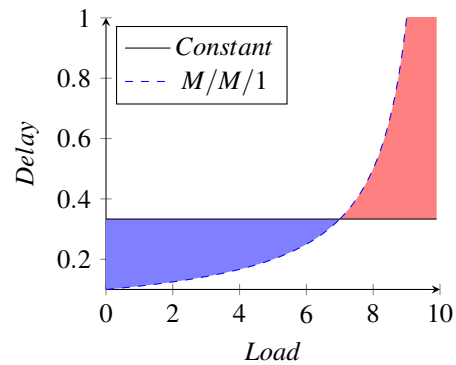

Figure 1: Different delay models as a function of the load. The blue zone (resp. red zone) represents the overestimated (resp. underestimated) delay by the constant model over the M/M/1 model.

The Variable Delay Multi-Commodity Flow (VD-MCF) problem consists in routing a set of requests in the network respecting the link capacity constraints and requests delay constraints, while minimizing the maximum delay experienced by a request. Each request $k \in K$ is characterized by its source s_k, its destination t_k, its bandwidth requirement D_k and its end-to-end latency requirement Λ_k.

Multiple functions can be used to model the delay as a function of the load. Figure 1 shows two possible functions: the M/M/1 delay function and the constant delay function. While the M/M/1 function fairly represents the average delay spent in a queue, the constant delay is simpler and can be easily integrated in global optimization models. Operators usually set the delay to the one observed at a 70% load, but this simple rule of thumbs leads to two major inconveniences: the delay at load below 70% is over-estimated — especially at lower loads — and the delay at loads above 70% is vastly under-estimated, leading to consider the capacity to be 70% of the original one to have a valid model.

Focusing on the M/M/1 model, we formulate the problem modeled as follows. For each link $e \in E$ and each demand $k \in K$ we consider a binary variable y_{ek} equal to 1 if the request k is routed on the link $e \in E$ and 0 otherwise. The variable z gives the maximum ratio of delay a request experiences. The average queuing time at node $v \in V$ for demand $k \in K$ is given by $x_v^k = \frac{1}{B_v - l_v}$ where the $l_v = \sum_{e \in \delta^+(v)} \sum_{k' \in K} D_{k'} y_{ek'}$ is the load on the node.

The VD-MCF problem is then formulated as follows:

$$\min z \tag{1a}$$

$$\alpha_{vk}: \quad \sum_{e \in \delta^+(v)} y_{ek} - \sum_{e \in \delta^-(v)} y_{ek} = \begin{cases} 1 \text{ if } v = s_k \\ -1 \text{ if } v = t_k \\ 0 \text{ otherwise} \end{cases} \quad \forall v \in V, \forall k \in K \tag{1b}$$

$$\alpha_e: \quad \sum_{k \in K} D_k y_{ek} \leq B_e \quad \forall e \in E, \tag{1c}$$

$$\beta_k: \quad \sum_{v \in V} \sum_{e \in \delta^+(v)} (x_v^k + \lambda_e) y_{ek} \leq z \Lambda_k \quad \forall k \in K, \tag{1d}$$

Inequalities α_{vk} are the flow conservation constraints, inequalities α_e are the link capacity constraints and inequalities β_k are the delay constraints. Unfortunately, these last constraints are not linear and thus the current model cannot be solved by linear solvers without linearizing them.

To do so, we introduce, for each pair of request kk' and each link $e = uv$, the variable $z_{ue}^{kk'}$ equals to x_u^k if $y_{ek'} = 1$ and 0 otherwise. The inequalities (1d) can then be replaced by:

$$\beta_k : \quad \sum_{v\in V}\sum_{e\in\delta^+(v)}\sum_{k'\in K} z_{ve}^{kk'} + \sum_{v\in V}\sum_{e\in\delta^+(v)} \lambda_e y_{ek} \leq z\Lambda_k \qquad \forall k \in K, \tag{2a}$$

$$B_v x_v^k - \sum_{e'\in\delta^+(v)}\sum_{k'\in K} D_{k'} z_{ve}^{kk'} = 1 \qquad \forall k \in K, \forall u \in V \tag{2b}$$

$$z_{ve}^{kk'} \leq x_v^k \qquad \forall k,k' \in K, \forall v \in V, \forall e' \in \delta^+(v) \tag{2c}$$

$$z_{ve}^{kk'} \leq M y_{ek'} \qquad \forall k,k' \in K, \forall v \in V, \forall e' \in \delta^+(v) \tag{2d}$$

$$x_v^k - (1 - y_{e'k'})M \leq z_{ve}^{kk'} \qquad \forall k,k' \in K, \forall v \in V, \forall e' \in \delta^+(v) \tag{2e}$$

This linearization comes with the heavy price of using a " big M " in inequalities (2d) and (2e) and $|K|^2|E||V|$ additional variables and $O(|K|^2|E||V|)$ additional constraints. We thus propose a more compact formulation that can efficiently be solved with column generation. For each node $v \in V$ and for each subset $K' \subseteq K$ a binary variable $x_v^{K'}$ equals to 1 if the set of demand K' traverses the node v and 0 otherwise. $\Lambda_v^{K'}$ corresponds to the delay on v when it is processing the demands in K'. We can replace inequalities (1d) by

$$\beta_k : \quad \sum_{v\in V}\sum_{K'\subseteq K: k\in K'} \lambda_u^{K'} x_v^{K'} + \sum_{e\in E} \lambda_e y_{ek} \leq z\Lambda_k \qquad \forall k \in K, \tag{3}$$

Variables $x_v^{K'}$ are in exponential number but we rely on a decomposition method that can solve linear programs with huge number of variables by generating them using a pricing problem. The pricing problem consists of finding new columns with a negative reduced costs. If no such column exists, the solution provided by the current column set is optimal. In our case, the pricing problem consists of finding a subset of demands for each node. Let $\bar{x}_k$ be a binary variable equal to 1 if the demand k is selected. The objective function (i.e., the reduced cost) of the pricing problem is to min $\beta_v - \sum_{k\in K}(\beta_{uk}^1 + \beta_{uk}^2)\bar{x}_k - \sum_{k\in K}\beta_k y_{K'}$ where $y_{K'} = \frac{1}{B_v - \sum_{k\in K'} D_k \bar{x}_k}$ and β_v, β_{uk}^1, β_{uk}^2, β_k are the dual values of the corresponding inequalities. This problem can be linearized using the same technique as for the original problem.

3 Numerical results

To compare our model solved with column generation with M/M/1 delay or a constant delay, we generated 10 random IP-RAN networks composed of a 10-nodes core connected to two domains of 5 nodes with 10 requests. The latency and bandwidth requirements are generated at random so that a feasible routing can be found. For the delay and bandwidth requirements, we consider three scenarios: loose, tight and ultra tight.

We first compare three models in these scenarios: the constant delay, the constant delay with 70% capacity and the M/M/1. We define two multiplicative ratios, θ and ρ, that we apply to scale respectively all delay and all bandwidth requirements of the requests. The lower θ is, the tighter the delay requirement the model can accept; the higher the ρ is, the higher bandwidth requirement the model can accept.

Using the compact linearized ILP model (inequalities (1a) to (1c) and (2a) to (2e)), we see, in Figure 2, that the M/M/1 model can tighten the demand requirements slightly more than the two constant models while it can support up to 770% increase in bandwidth requirements, compared to more than 500% for the *constant (70%)* model in Tight and Loose scenarios. On ultra tight scenarios, the M/M/1 model can find solutions, albeit with smaller ρ and θ, while the *constant (70%)* model cannot find any solution.

Surprisingly, the constant model outperforms the M/M/1 model for Ultra Tight bandwidth scenarios and can even compete with it on Loose and Tight delay requirements. But when we compute the "true" delay of these solutions, they violate some delay requirements due to the under-estimation of the delay in the 70 to 100% load range, making them unusable in practice.

We then compare in Table 1 the optimal value obtained with the linearized ILP, the bound provided by the linear relaxation of the linearized ILP (by relaxing the integrality constraints) and the bound provided by the Column Generation (CG) algorithm. Our decomposition shines by providing better bounds than the compact formulation. The difference grows even more as the requirements gets tighter and the CG algorithm bounds are still close to the optimal ILP solutions.

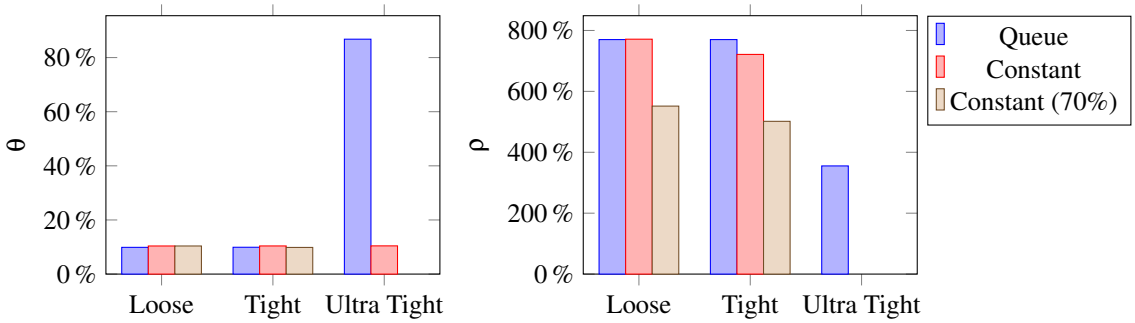

Figure 2: Comparison of the possible tightening of the delay (left) and bandwidth requirements (right) between the constant and M/M/1 models for different scenarios of delay requirements and link bandwidth capacity.

Bandwidth Requirements	Delay Requirements	CG	ILP	LP
Loose	Loose	9.65	9.65	9.56
	Tight	12.38	12.38	12.26
	Ultra Tight	99.43	99.44	98.48
Tight	Loose	9.70	9.70	9.59
	Tight	12.44	12.45	12.30
	Ultra Tight	99.88	99.95	98.83
Ultra Tight	Loose	77.86	77.87	77.71
	Tight	99.87	99.87	99.68

Table 1: Comparison of z for the ILP model, LP relaxation and the Column Generation model for different bandwidth and delay requirements. Ultra Tight bandwidth requirements are not shown with Ultra Tight delay requirements because they are infeasible.

4 Conclusion

We have introduced a new decomposition model for solving the Variable Delay Multi-commodity Flow (VD-MCF) problem. After exposing the limitation of the constant delay model, we show that our column generation algorithm can compete with the ILP formulation. Finally, our CG model can also integrates any kind of delay model in the pricing problem; we could replace the simple M/M/1 model to a more complex one such as a machine learning oracle, which would be impossible using the original model.

References

[BAO06] Walid Ben-Ameur and Adam Ouorou. Mathematical models of the delay constrained routing problem. *Algorithmic Operations Research*, 1(2), 2006.

[FGJM17] Bernard Fortz, Luis Gouveia, and Martim Joyce-Moniz. Models for the piecewise linear unsplittable multicommodity flow problems. *European Journal of Operational Research*, 261(1):30–42, 2017.

[FT00] Bernard Fortz and Mikkel Thorup. Internet traffic engineering by optimizing ospf weights. In *Proc.IEEE INFOCOM*, 2000.

[Kle07] Leonard Kleinrock. *Communication nets: Stochastic message flow and delay*. Courier Corporation, 2007.

[TDM10] Jérôme Truffot, Christophe Duhamel, and Philippe Mahey. k-Splittable delay constrained routing problem: A branch-and-price approach. *Networks*, 55(1):33–45, 2010.