

# Algorytmiczne aspekty współczesnych sieci komputerowych

Maciej Pacut  
Streszczenie pracy doktorskiej

Optymalizacja wykorzystania łączy sieciowych w centrach obliczeniowych jest kluczowa do przeprowadzenia efektywnych wielkoskalowych obliczeń, gdyż pojedynczym zadaniem obliczeniowym zajmują się liczne jednostki obliczeniowe, które współpracują wymieniając duże ilości danych przez sieć. Centra obliczeniowe udostępniają swoje zasoby obliczeniowe w formie maszyn wirtualnych połączonych siecią, razem zwanych siecią wirtualną. W celu efektywnego wykorzystania sieci, maszyny wirtualne współpracujące w obrębie jednego zadania powinny znajdować się blisko siebie i blisko danych, które przetwarzają. Aby zagwarantować efektywne wykorzystanie łączy (wysoką przepustowość i małe opóźnienia) dla wielu jednocześnie operujących sieci wirtualnych, konieczne są uprzednie rezerwacje przepustowości na łączach. Zarządzanie rezerwacjami jest problemem algorytmicznym o wysokim znaczeniu praktycznym.

Z algorytmicznego punktu widzenia, rezerwacja łączy w centrum obliczeniowym odpowiada zadaniu osadzenia (ang. embedding) grafu komunikacji maszyn wirtualnych w grafie topologii sieciowej centrum obliczeniowego [GKK<sup>+</sup>01]. Pożądane jest znalezienie takiego osadzenia, które umiejscawia często komunikujące się maszyny wirtualne blisko siebie. Z powodu dynamicznej i trudnej do przewidzenia natury obliczeń, statyczne umiejscowienie maszyn wirtualnych może prowadzić do nieefektywnego użycia sieci. Nowoczesne centra obliczeniowe dostarczają narzędzia pomagające sprostać temu problemowi: maszyna wirtualna może zostać przeniesiona (ang. migration) na inną fizyczną maszynę. To pozwala na dostosowanie osadzenia do zmieniających się w czasie potrzeb komunikacyjnych sieci wirtualnych. W niniejszej pracy zakładamy, że wzorzec komunikacji nie jest znany z wyprzedzeniem, a algorytm dokonuje rezerwacji na bieżąco. Z teoretycznego punktu widzenia, przedstawiony problem jest uogólnieniem problemu stronicowania.

W drugiej części pracy rozważam problem przekierowania pakietów (ang. packet forwarding), który jest fundamentalnym problemem występującym w sieciach komputerowych. Gdy urządzenie sieciowe odpowiedzialne za przesyłanie pakietów (tj. router) otrzymuje pakiet, jego zadaniem jest przekazać go do odpowiedniego portu prowadzącego do sąsiadującej sieci. W celu właściwego wyboru portu, router przechowuje tablicę przekierowania (ang. forwarding table), która składa się z reguł przypisujących adresy docelowe pakietów do właściwych portów. Urządzenia sieciowe mają ograniczone zasoby pamięciowe i często spotykana jest sytuacja, gdzie rozmiar tablicy przekierowania przekracza dostępną pamięć.

Przedmiotem badań jest metoda, której zastosowanie pozwala opóźnić kosztowną rozbudowę pamięci istniejących urządzeń sieciowych. Metoda polega na przechowaniu jedynie pewnego podzbioru reguł w urządzeniu sieciowym, a pełnej tablicy przekierowania na urządzeniu pomocniczym wyposażonym w obszerniejszą, choć wolniejszą pamięć [KARW16, KCGR09, Liu01, LLW15, SUF<sup>+</sup>12]. Tak sformułowane zadanie przypomina problem stronicowania, rozszerzony o hierarchiczne zależności między wpisami tablicy przekierowania.

Oba postawione problemy badam metodą analizy konkurencyjności [BE98], która jest odpowiednia do badania problemów, które są dynamiczne z natury. W pracy przedstawiam deterministyczne algorytmy online o zadowalającym współczynniku konkurencyjności oraz konstruuje nietrywialne dolne granice dla tych problemów.

# Bibliografia

- [BE98] Allan Borodin and Ran El-Yaniv. Online computation and competitive analysis. *Cambridge University Press*, 1998.
- [GKK<sup>+</sup>01] Anupam Gupta, Jon Kleinberg, Amit Kumar, Rajeev Rastogi, and Bulent Yener. Provisioning a virtual private network. *Proc. ACM Symposium on Theory of Computing (STOC)*, pages 389–398, 2001.
- [KARW16] Naga Katta, Omid Alipourfard, Jennifer Rexford, and David Walker. Cacheflow: Dependency-aware rule-caching for software-defined networks. In *Proc. ACM Symposium on SDN Research (SOSR)*, 2016.
- [KCGR09] Changhoon Kim, Matthew Caesar, Alexandre Gerber, and Jennifer Rexford. Revisiting route caching: The world should be flat. In *Proc. 10th Int. Conf. on Passive and Active Network Measurement (PAM)*, pages 3–12, 2009.
- [Liu01] Huan Liu. Routing prefix caching in network processor design. In *Proc. 10th Int. Conf. on Computer Communications and Networks (ICCCN)*, pages 18–23, 2001.
- [LLW15] Yaoqing Liu, Vince Lehman, and Lan Wang. Efficient FIB caching using minimal non-overlapping prefixes. *Computer Networks*, 83:85–99, 2015.
- [ST85] Daniel Sleator and Robert Tarjan. Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28(2):202–208, 1985.
- [SUF<sup>+</sup>12] Nadi Sarrar, Steve Uhlig, Anja Feldmann, Rob Sherwood, and Xin Huang. Leveraging Zipf’s law for traffic offloading. *ACM SIGCOMM Computer Communication Review*, 42(1):16–22, 2012.