

Ściąganie danych i rozpraszanie obliczeń.

Michał Hanćkowiak, specjalność Projektowanie Algorytmów

1. Charakterystyka obszaru badawczego

Niektóre algorytmy powinny działać w wersji "rozproszonej" w tym sensie, że ich dane wejściowe znajdują się w węzłach sieci, np. Internetu. Przykładem może być uczenie maszynowe, gdy dane do uczenia pobierane są na wielu węzłach sieci równocześnie (może tam być np. czujnik temperatury). Rozpraszanie obliczeń/ algorytmów można przeprowadzać na następujące sposoby:

1) Przez centralizację obliczeń. Można ściągać dane do jednego węzła (data gathering, convergecast) i wszelkie obliczenia wykonywać lokalnie na tym węźle. Przy tym podejściu niezbędne są algorytmy, które w sposób wysoce optymalny ściągają dane do wierzchołka przy różnych założeniach co do grafu komunikacyjnego i dopuszczalnej długości komunikatów.

2) Bez centralizacji. Obliczenia zostają w miarę możliwości rozdzielone na wiele węzłów sieci. Wolumen przesyłanych danych jest dzięki temu znacznie mniejszy niż w rozwiązaniu 1). Może to być rozwiązanie znacznie bardziej efektywne niż oparte na centralizacji, ale równocześnie otrzymujemy algorytmy mniej uniwersalne, które muszą być opracowywane osobno dla każdego problemu obliczeniowego. W ramach projektu badamy istniejące lub opracowujemy nowe algorytmy rozproszone dla obu podejść.

2. Motywacja

Algorytmy opracowane w ramach tego projektu być może będą miały szansę pojawić się w przyszłości w protokołach sieciowych, dzięki czemu aplikacje rozproszone będą działać sprawniej. Możemy sobie wyobrazić protokoły sieciowe wyspecjalizowane do wspierania obliczeń konkretnego rodzaju np. do obliczeń występujących w uczeniu maszynowym. Przydatne w praktyce mogą się okazać także protokoły sprawnie ściągające do węzła dane z wielu źródeł.

3. Obecny poziom badań i możliwości finansowania

Prace dotyczące rozpraszania obliczeń i ściągania danych: [1] [2] [3] [4] [5] [6].

4. Tematyka badawcza

+ Algorytmy rozproszone ściągające dane do węzła, z wielu źródeł, za pomocą ścieżki lub drzewa.

+ Ściąganie danych do węzła, gdy sieć składa się z gwiazdy i grafu dwudzielnego.

+ Obliczanie sum liczb w zbiorach rodziny zbiorów w sieci (dość uniwersalny problem).

+ Reprezentowanie i uczenie Perceptronu z jedną warstwą ukrytą, przy założeniu, że warstwa wejściowa jest rozproszona (każda element na innym węźle).

5. Wymagania odnośnie członków projektu

Studenci powinni mieć ogólną wiedzę o matematyce dyskretnej i o algorytmach sekwencyjnych. Dobrze, jeśli mają przygotowanie z algorytmów rozproszonych (np. w ramach zajęć "Algorytmy rozproszone" oferowanych na UAM WMI).

6. Literatura

[1] J. A. Garay , S. Kuttn , D. Peleg, "A SUBLINEAR TIME DISTRIBUTED ALGORITHM FOR MINIMUM-WEIGHT SPANNING TREES ",

[2] W. Shang, P. Wan, Xiaodong Hu, "Approximation algorithm for minimal convergecast time problem in wireless sensor networks",

[3] L. Kaur, J. Malhotra, "Comparison of Wise Route and Flooding Network Type of Convergecast Routing in Wireless Sensor Network",

[4] S. Chakraborty, S. Karmakar, "Adaptive Convergecast by Distributed Topology Switching ",

[5] L. Barenboim, G. Oren, "Fast Distributed Backup Placement in Sparse and Dense Networks",

[6] Y. Chang, "Efficient Distributed Decomposition and Routing Algorithms in Minor-Free Networks and Their Applications".