

Algorytmy grafowe w różnych modelach obliczeń rozproszonych i równoległych.

Kierownik projektu:

Krzysztof Nowicki

Wydział Matematyki i Informatyki

Uniwersytet Wrocławski

Modele obliczeń równoległych i rozproszonych

W obecnych czasach ilość przetwarzanych danych rośnie znacznie szybciej niż moc obliczeniowa pojedynczego procesora. Jednym ze sposobów na poradzenie sobie z tym problemem jest zwiększenie liczby procesorów które przetwarzają dane - to jednak wymaga pewnej koordynacji pomiędzy procesorami / maszynami, które wykonują obliczenia. W związku z tym proponuje się różne modele obliczeń rozproszonych i równoległych, które pozwalają na formalne opisanie algorytmów korzystających z wielu procesorów.

W przeciwieństwie do obliczeń sekwencyjnych, dla których model RAM jest powszechnie używany do modelowania obliczeń, nie istnieje jeden model, który dobrze opisuje wszystkie aspekty obliczeń rozproszonych i równoległych. Jest to spowodowane tym, że obliczenia równoległe i rozproszone mogą być zrealizowane na wiele różnych sposobów. W związku z tym jest wiele czynników wpływających na to, jakiego rodzaju techniki rozwiązywania problemów dają efektywne algorytmy: wpływ na to mają między innymi to jak realizowana jest komunikacja pomiędzy procesorami, jaka jest liczba procesorów względem rozmiaru danych oraz jak wiele wiadomości w jednostce czasu może wysłać pojedynczy procesor.

W moich badaniach głównie rozważam algorytmy synchroniczne (wykonywane w rundach), dla modeli w których każda para maszyn potencjalnie może się ze sobą skomunikować w jednej rundzie, niemniej liczba wysłanych / odbieranych wiadomości jest ograniczona. Obliczenia takie, nie dają się sklasyfikować jednoznacznie jako obliczenia równoległe lub rozproszone (według ogólnie akceptowanych definicji z lat 90-tych) i często uważa się zarówno za obliczenia równoległe jak i rozproszone. Za praktyczną realizację takich modeli są uważane np. MapReduce, Hadoop i Spark.

Problemy grafowe

W moich badaniach głównie zajmuję się problemami grafowymi, głównym powodem jest to, że za ich pomocą można opisać wiele problemów, które spotyka się w prawdziwym życiu.

Przykład: Jednym z najstarszych przykładów problemu, który został zamodelowany za pomocą problemu grafowego i rozwiązany za pomocą algorytmu ten problem rozwiązującego, jest problem zbudowania najtańszej sieci elektrycznej. Innymi słowy, 'wejściem' problemu był zbiór miast które mają zostać połączone wspólną siecią elektryczną. Aby opisać ten problem za pomocą języka grafów, powiemy, że każde miast jest wierzchołkiem grafu, a krawędzie pomiędzy wierzchołkami mają wagi równe cenom zbudowania połączenia elektrycznego pomiędzy miastami. Celem jest znalezienie Minimalnego Drzewa Spinającego tego grafu – takie drzewo, to najtańszy zbiór krawędzi, który spowoduje, że każde dwa wierzchołki są połączone. Odpowiada to zbiorowi najtańszych połączeń elektrycznych, który gwarantuje, że każde miasto jest podłączone do wspólnej sieci.

Ogólnie o problemach rozważanych w projekcie: Poza tym konkretnym przykładem istnieje wiele innych problemów które da się wyrażać jako problemy grafowe, w szczególności są też problemy takie, dla których przetwarzane grafy są olbrzymie, lub dane wejściowe są rozproszone geograficznie. W obu przypadkach niemożliwym jest zebranie pełnej informacji w jednym miejscu w krótkim czasie. W związku z tym rozważanie algorytmów równoległych i rozproszonych jest uzasadnione i to właśnie planuję robić w zakresie mojego projektu.

Moim celem jest projektowanie algorytmów dla problemów które mają wiele wspólnego ze wspomnianym już problemem znajdowania Minimalnego Drzewa Spinającego, jak również innych problemów uważanych za podstawowe problemy dziedziny algorytmów grafowych, np. różnych wariantów problemu znajdowania najkrótszych ścieżek w grafie.