Balanced partitioning of grids and related graphs
A theoretical study of data distribution in parallel finite element model simulations

Author(s): Feldmann, Andreas E.
Publication Date: 2012

Permalink: https://doi.org/10.3929/ethz-a-007321964

Rights / License: In Copyright - Non-Commercial Use Permitted
Balanced Partitioning of Grids and Related Graphs
A Theoretical Study of Data Distribution in Parallel Finite Element Model Simulations

A dissertation submitted to

ETH Zurich

for the degree of Doctor of Sciences

presented by

Andreas Emil Feldmann
Diplom Informatiker, RWTH Aachen
born April 19, 1982,
citizen of Germany and Sweden

accepted on the recommendation of

Prof. Dr. Peter Widmayer, ETH Zurich
examiner

Prof. Dr. Christos H. Papadimitriou, UC Berkeley
co-examiner

2012
Abstract

This thesis considers the $k$-BALANCED PARTITIONING problem, which is defined as follows. Find the minimum number of edges in a graph that, when cut, partition the vertices into $k$ (almost) equally sized sets. Amongst others, the problem derives its importance from the need to distribute data within a parallel-computing architecture. In this setting we are particularly interested in 2D finite element model (FEM) simulations. We therefore model the input as a regular quadrilateral tiling of the plane. More precisely, we focus on solid grid graphs. These are finite connected subgraphs of the infinite 2D grid without holes. However we also consider other graph classes. In particular, trees often give surprising conclusions to the problem on grid graphs.

For the special case when $k = 2$ (the BISECTION problem) we show that an optimal solution can be found in $O(n^4)$ time for solid grid graphs. However the resulting runtime is unsatisfactory for practical purposes. Therefore we also show that near-optimal solutions can be found in $O(n^{1.5})$ time. This is achieved by analysing the cut shapes in solid grid graphs. We prove that simple shapes corresponding to straight lines and right-angled corners suffice in order to find near-optimal cuts.

We are able to further harness these structural insights for the general case when $k$ takes arbitrary values. This results in a fast bicriteria approximation algorithm which runs in $O(n^{1.5} \log k)$ time. The number of edges it cuts is at most $O(\log k)$ times the optimum and the sets are at most a factor of 2 from equal-sized. For practical purposes however, the latter deviation is unattractive. In particular in parallel-computing it means a slowdown by a factor of 2. Therefore we also consider algorithms that compute sets that are arbitrarily close to equal-sized. For trees we provide a PTAS. Furthermore, we provide a bicriteria approximation algorithm for general graphs. Here the number of edges cut is at most $O(\log n)$ times the optimum. This result is obtained by harnessing results on hierarchical decompositions of graphs into trees, together with our PTAS.
Next we show that when $k$ can be arbitrary, considering bicriteria approximations, as above, is necessary. For this we provide the following hardness results. When equal-sized sets are desired for solid grid graphs, it is NP-hard to approximate the number of cut edges within $n^c$, for any constant $c < 1/2$. For trees we can even prove this for any constant $c < 1$. We set up a general reduction framework in order to generate these results. The framework identifies some sufficient conditions for the considered graph class which make the problem hard. For trees the conditions are met by relying on high vertex degrees. We therefore also consider constant degree trees. For these the problem is APX-hard. This means that for trees the complexity grows with the maximum degree.

Note that one of the provided bicriteria approximation algorithms for arbitrary $k$ is fast but yields an unsatisfactory set-size ratio. The other algorithm is slow but achieves high-quality set sizes. This is because its runtime increases exponentially when the limit on the set-sizes becomes more stringent. We justify the achieved tradeoff by showing that it is necessary. For both grids and trees we prove that, unless $P=NP$, no fully polynomial time algorithms exist that compute sets which are arbitrarily close to equal-sized. This is true even if the number of cut edges is allowed to deviate further from the optimum the more stringent the limit on the set sizes.
Zusammenfassung

Wir untersuchen das $k$-BALANCIERTE PARTITIONIERUNGS Problem, in dem eine kleinste Menge von Kanten eines Graphen gefunden werden soll, die, wenn sie geschnitten werden, die Knoten in $k$ (fast) gleich große Mengen partitionieren. Dieses Problem leitet seine Bedeutung u. a. von der Notwendigkeit her, Daten auf einem Parallelrechner zu verteilen. Da uns insbesondere Simulationen von zweidimensionalen Finite-Elemente-Methoden (FEMs) interessieren, modellieren wir die Eingabe als gleichmäßige viereckige Parkettierung der Ebene. Um genauer zu sein, konzentrieren wir uns auf solide Gittergraphen, die endliche verbundene Teilgraphen des unendlichen zweidimensionalen Gitters ohne Löcher sind. Allerdings betrachten wir auch andere Graphen. Insbesondere liefern Bäume überraschende Einsichten in das Problem auf Gittergraphen.

Im speziellen Fall wenn $k = 2$ (das HALBIERUNGS Problem) zeigen wir, dass eine optimale Lösung in $O(n^4)$ Zeit für solide Gittergraphen gefunden werden kann. Da allerdings die Laufzeit unzufriedenstellend für praktische Anwendungen ist, zeigen wir auch, dass nahezu optimale Lösungen in $O(n^{1.5})$ Zeit gefunden werden können. Dies wird durch eine Analyse der Schnittformen in soliden Gittergraphen erreicht. Wir beweisen, dass einfache Schnittformen wie gerade Linien oder rechte Winkel ausreichen, um annähernd optimale Schnitte zu finden.


Es ist auffallend, dass einer der aufgezeigten Approximationsalgorithmen für beliebige $k$ schnell ist, aber keine zufriedenstellende Mengengrössen produziert, während der andere langsamer ist, aber hochwertige Grössen berechnen kann. Dies gilt, da die Laufzeit des letzteren Verfahrens exponentiell wächst, wenn die Begrenzung der Grössen strikter wird. Wir rechtfertigen den erlangten Kompromiss, indem wir zeigen, dass er notwendig ist. Sowohl für Gitter als auch für Bäume beweisen wir, dass kein Algorithmus existiert, dessen Laufzeit voll polynomiell ist, und Mengen produziert, die beliebig nahe an die gewünschte Zielgrösse herankommen. Dies ist sogar dann wahr, wenn die Anzahl der geschnittenen Kanten weiter vom Optimum abweichen darf, je strikter die Begrenzung der Mengengrössen wird.