ON APPROXIMATION ALGORITHMS FOR DETERMINING MINIMUM COST EMULATIONS

H.L. Bodlaender

RUU-CS-85-10
March 1985

Rijksuniversiteit Utrecht

Vakgroep informatica

Budapestlaan 6 3584 CD Utrecht
Corr. adres: Postbus 80.012 3508 TA Utrecht
Telefoon 030-53
The Netherlands

ON APPROXIMATION ALGORITHMS FOR DETERMINING MINIMUM COST EMULATIONS

H.L. Bodlaender

Technical Report RUU-CS-85-10
March 1985

Department of Computer Science
University of Utrecht
P.O. Box 80.012, 3508 TA Utrecht
the Netherlands

ON APPROXIMATION ALGORITHMS FOR DETERMINING MINIMUM COST EMULATIONS*

H.L. Bodlaender

Department of Computer Science, University of Utrecht
P.O. Box 80.012, 3508 TA Utrecht, the Netherlands

Abstract. Emulations are structure preserving simulations of large (processor-) networks on smaller networks. In this note the notion of computation cost of an emulation is introduced. It is shown that every polynomial time approximation algorithm for minimizing the computation cost of an emulation of a network G on a network H must have a worst case in which the resulting approximation differs at least a factor 2 from the optimal solution (unless P=NP). Relations between approximation algorithms that minimize the computation cost of emulations and approximation algorithms for BANDWIDTH, CLIQUE and BALANCED COMPLETE BIPARTITE SUBGRAPH are given, which indicate that it will be hard (if not impossible) to find polynomial time approximation algorithms that give "good" approximations of the minimum computation cost possible for emulations of a given graph G on a given graph H.

1. Introduction. Parallel algorithms are normally designed for execution on a suitable network of N processors, with N depending on the size of the problem to be solved. In practice N will be large and varying, whereas processor networks will be small and fixed. The resulting disparity between algorithm design and implementation must be resolved by simulating a network of some size N on a fixed and smaller size network of a similar or different kind, in a structure

---

*This work was supported by the Foundation for Computer Science (SION) of the Netherlands Organisation for the Advancement of Pure Research (ZWO).

preserving manner. In 1982, Fishburn and Finkel [3] proposed such a notion of simulation, termed emulation.

<u>Definition</u>. Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be networks of processors (graphs). We say that G can be emulated on H if there exists a function $f: V_G \rightarrow V_H$ such that for every edge $(g,g') \in E_G$: $f(g) = f(g')$ or $(f(g), f(g')) \in E_H$. The function f is called an emulation function or, in short, an emulation of G on H.

Let f be an emulation of G on H. Any processor $h \in V_H$ must actively emulate the processors $\in f^{-1}(h)$ in G. When $g \in f^{-1}(h)$ communicates information to a neighbouring processor g', then h must communicate the corresponding information "internally", when it emulates g' itself, or to a neighbouring processor $h' = f(g')$ in H otherwise. If all processors act synchronously in G, then the emulation will be slowed by a factor proportional to $\max_{h \in V_H} |f^{-1}(h)|$.

<u>Definition</u>. Let G,H and f be as above. The computation cost of f is $cc(f) = \max_{h \in V_H} |f^{-1}(h)|$.

<u>Definition</u>. Let G, H, f be as above. The emulation f is said to be (computationally) uniform iff for all $h,h' \in V_H$: $|f^{-1}(h)| = |f^{-1}(h')|$.

<u>Proposition 1.1</u>. Let G,H and f be as above.
f is uniform iff $cc(f) = |V_G|/|V_H|$.

In [1,2] we addressed the problem to determine whether a given connected graph G can be uniformly emulated on another given connected graph H. The problem was proved to be NP-complete, even if various additional, realistic constraints are posed upon G and H. A natural extension of the UNIFORM EMULATION problem is the following problem:

[MINIMUM COST EMULATION]

Instance: Connected graphs G,H , integer $c \in \mathbb{N}^+$.

Question: Is there an emulation f of G on H, with $cc(f) \leq c$?

We will abbreviate MINIMUM COST EMULATION as MCE.

MCE is also NP-complete: it contains UNIFORM EMULATION as a subproblem. However, it seems reasonable to look for approximation algorithms for this problem: an emulation with "relatively low cost" will enable us to simulate the given large network G on the given smaller network H "relatively efficiently".

Unfortunately there is some evidence that it will be hard to find polynomial time approximation algorithms for MCE that find "good" approximations for the computation cost of the emulation. By "good" approximations we mean e.g. approximations that give a computation cost that differs at most by a constant factor from the optimal solution.

In section 3 we show that every polynomial time approximation algorithm for MCE will give - in worst case - approximations that differ at least a factor 2 from the optimal solution (unless P=NP). In sections 4,5 and 6 we show that the existence of a "good" polynomial time approximation algorithm for MCE will imply the existence of "good" polynomial time approximation algorithms for BANDWIDTH, CLIQUE and BALANCED COMPLETE BIPARTITE SUBGRAPH. However, for the latter three problems no polynomial time approximation algorithms are known to exist that guarantee approximations that differ for instance by a constant or even a logarithmic (in the size of the problem) factor from the optimal solution. In 1973 Johnson [6] proved that for all of the polynomial time approximation algorithms for CLIQUE that had been suggested at the time, the worst case ratio between the approximation and the optimal value grows at least as fast as $O(n^{\varepsilon})$, where n is the problem size and $\varepsilon > 0$ depends on the algorithm. Presently no algorithm that gives better ratios is known.

2. Definitions and notations. The frame work in which we present the results is taken from [4, chapter 6]. An approximation problem Π

consists of the following parts:

(1) a set D_π of instances ($D_\pi \neq \emptyset$).

(2) for each instance $I \in D_\pi$ a finite set $S_\pi(I)$ of candidate solutions for I ; and

(3) a function m_π, that assigns to each instance $I \in D_\pi$ and each candidate solution $\sigma \in S_{\pi(I)}$ a positive rational number $m_\pi(I,\sigma)$, called the solution value for σ.

We call Π a minimization (maximization) problem, if we look for (approximations of) the minimal (maximal) value of $m_\pi(I,\sigma)$ for a given instance $I \in D_\pi$ and all candidate solutions $\sigma \in S_n(I)$. This minimal (maximal) value is denoted by $OPT_\pi(I)$. (The subscript Π is usually dropped when the problem is clear from the context.)

Let A be an approximation algorithm for a minimization problem Π, (like MINIMUM COST EMULATION), and let $I \in D_\pi$, and $A(I)$ be the approximation for $OPT_\pi(I)$, yielded by the algorithm A on input I.

<u>Definitions</u>. $R_A(I) = \dfrac{m_\pi(I,A(I))}{OPT_\pi(I)}$.

$R_A = \inf \{ r \geq 1 \mid \forall I \in D_\pi \; R_A(I) \leq r\} = \sup \{R_A(I) \mid I \in D_\pi\}$.

$R_A^\infty = \inf \{ r \geq 1 \mid \exists N \in \mathbb{N}^+ : \forall I \in D_\pi \; OPT_\pi(I) \geq N \Rightarrow R_A(I) \leq r\}$.

$R_{min}(\Pi) = \inf \{ r \geq 1 \mid$ there exist a polynomial time approximation algorithm A for Π with $R_A^\infty = r\}$.

R_A is called the "absolute performance ratio"; R_A^∞ is called the "asymptotic performance ratio". These ratios will always satisfy $1 \leq R_A^\infty \leq R_A \leq \infty$. Ratios that are closer to 1 indicate a better performance. $R_{min}(\Pi)$ is called the "best achievable asymptotic performance ratio".

For a further discussion of these measures see [4]. For approximation algorithms A for maximization problems define $R_A(I) = \dfrac{OPT_\pi(I)}{m_\pi(I,A(I))}$. ($R_A(I)$ denotes the ratio between the optimal value and the approximation for input I.)

3. A bound for the best achievable asymptotic performance ratio on MCE.

Theorem 3.1. If P ≠ NP, then no polynomial time approximation algorithm A for MINIMUM COST EMULATION can satisfy $R_A^\infty < 2$.

Proof. Let P ≠ NP. Suppose A is a polynomial time approximation algorithm for MINIMUM COST EMULATION with $R_A^\infty < 2$. Then there exist $\gamma < 2$ and $N \in \mathbb{N}^+$, such that for all connected, undirected graphs G,H with $cc(G,H) \geq N$, $cc(A(G,H)) \leq \gamma . cc(G,H) < 2\, cc(G,H)$. Let such an $N \in \mathbb{N}^+$ be given . (γ is not used.)

We will now give a polynomial time algorithm B_N that solves HAMILTONIAN CIRCUIT for graphs with nodes of degree 3. (This subproblem of HAMILTONIAN CIRCUIT is NP-complete [4].)

Let $H = (V_H, E_H)$ be a connected, undirected graph with nodes of degree 3. The graph $H^o = (V_H^o, E_H^o)$ is obtained in the following manner: each edge $(v_o, v_1) \in E_H$ is replaced by a path with length 3: i.e. we introduce additional nodes v_{01}, v_{1o}, and edges (v_o, v_{01}), (v_{01}, v_{1o}) and (v_{1o}, v_1). Furthermore to each node $v \in V_H \subseteq V_H^o$ we add two extra branches, consisting of one extra node each. An example of this transformation is given in fig. 3.1.

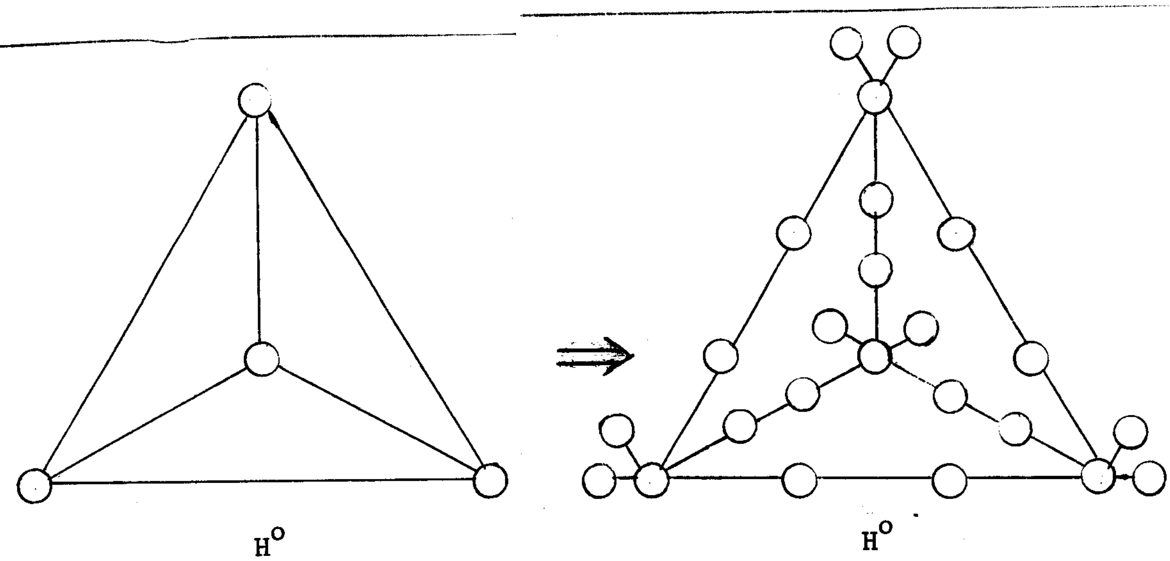
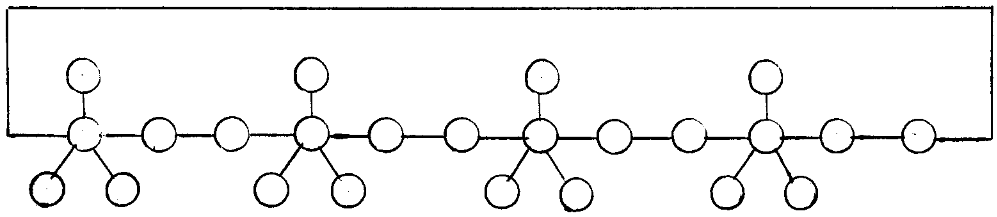


fig. 3.1.

So $|V_H^o| = 6.|V_H|$. Let $G=(V_G,E_G)$ be the graph consisting of a cycle with $3.|V_H|$ nodes, in which to every third node on the cycle 6N-3 branches are added, each consisting of one node (see fig. 3.2.) So $|V_G| = 6N|V_H| = N\,|V_H^o|$.



G, with N = 1, $|V_H| = 4$. fig. 3.2.

Claim 3.1.1. a. If H contains a Hamiltonian circuit, then $cc(G,H^o) = N$.

b. If H does not contain a Hamiltonian circuit, then $cc(G,H^o) \geq 2\,N$.

Proof. a). It is clear that $cc(G,H^o) \geq \frac{|V_G|}{|V_{H^o}|} = N$. Now suppose H contains a Hamiltonian circuit. We can map the successive nodes of the cycle in H with degree 6N-1 on the successive nodes $v \in V_H \subseteq V_H^o$, visited by the Hamiltonian circuit. The other nodes can be mapped in such a way that the resulting function f is a uniform emulation, i.e. $cc(f) = N$.

b) If an emulation f maps two nodes v_o, $v_1 \in V_G$ with degree(v_o)= degree (v_1)= 6N-1 on the same node $w \in V_H^o$, then $cc(f) \geq 2N$ (at least 12 N nodes are mapped upon 6 nodes). If f maps a node $v \in V_G$ with degree (v)= 6N-1 on a node $w \in V_H^o$ with degree (w) = 2, then at least 6N nodes must be mapped upon 3 nodes, so $cc(f) \geq 2N$. So if $cc(G,H^o) < 2N$, then there exists an emulation f of G on H^o, that maps nodes $v \in V_G$ with degree (v)= 6N-1 on nodes $w \in V_H \subseteq V_H^o$ on a one-to-one basis. Now let v^o, v^1, v^2, ..., $v^{|V_H|}$ be the successive nodes in G with degree 6N-1. Then $f(v^o)$, $f(v^1)$, $f(v^2)$,...,$f(v^{|V_H|})$ (seen as nodes in G) form a Hamiltonian circuit. □

From claim 3.1.1. it immediately follows that $cc(A(G,H^o)) < 2N$ if and only if H contains a Hamiltonian circuit. This gives a method to test in polynomial time whether H contains a Hamiltonian circuit. (G and H^o can be constructed in polynomial time.) This contradicts our assumption that P ≠ NP. □

<u>Corollary 3.2</u>. If P ≠ NP, then R_{min} (MCE) ≥ 2.

4. <u>Relation of MCE with BANDWIDTH</u>.

<u>Definition</u>. Let $G=(V_G,E_G)$ be a connected, undirected graph and let $f: V_G \to \{o,\dots,|V_G|-1\}$ be a bijection. The bandwidth of f is max $\{|f(v)-f(w)| \mid (v,w) \in E_G\}$. The bandwidth of G is min ({Bandwidth (f)| f is a bijection of V_G to $\{o,\dots,|V_G|-1\}$).

Define BANDWIDTH to be the following problem: given a graph $G = (V_G,E_G)$, find a bijection $f : V_G \to \{o,\dots,|V_G|-1\}$ with a minimum bandwidth.

There exist a number of heuristics for BANDWIDTH (see e.g. [5,7]). Little is known about the performance ratios of these algorithms. For instance, it is not known whether there exists a polynomial time approximation algorithm A for BANDWIDTH with an absolute (or asymptotic) performance ratio that is bounded by some finite constant. We will see that the existence of a "good" approximation algorithm for MCE implies the existence of an approximation algorithm for BANDWIDTH that is at most "twice as bad".

<u>Definition</u>. The path with n nodes is the graph $P_n=(V_n,E_n)$ with $V_n=\{o,\dots,n-1\}$ and $E_n=\{(i,j) \mid i,j \in V_n \wedge |i-j|=1\}$.

<u>Lemma 4.1</u>. Let $G=(V,E)$ be an undirected graph. Let $K \in \mathbb{N}^+$, and let $n \geq \lceil |V|/K \rceil$. Then

Bandwidth (G) ≤ K

⇒ G can be emulated on P_n with computation cost K

⇒ Bandwidth (G) ≤ 2K-1.

Proof. Similar to the proof of lemma 2.1. in [2]. □

Corollary 4.2. Suppose there exists a polynomial time approximation algorithm A for MCE with $R_A(G,H) \leq g(|V_G|,|V_H|)$ for a certain function g. Then there exists a polynomial time approximation algorithm A' for bandwidth minimization with $R_{A'}(G) \leq 2g(|V_G|,|V_G|)$.

Proof. Use algorithm A with $H=P_{|V_G|}$. □

Corollary 4.3. R_{min} (bandwidth minimization) $\leq 2.R_{min}$ (MCE).

5. Relation of MCE with CLIQUE. Define CLIQUE to be the problem, given a graph, to find the maximum subgraph all of whose points are mutually adjacent. In 1973 Johnson [6] showed that for every polynomial time approximation algorithm for CLIQUE that had been suggested, there exists an $\varepsilon > o$, such that $R_A(G) \geq O(|V_G|^{\varepsilon})$, (for infinitely many graphs $G=(V_G,E_G)$). Presently no algorithm that gives better ratios is known.

Similar to the argument in section 4 one can show that a "good" polynomial time approximation algorithm A for MCE gives a "good" polynomial time approximation algorithm A' for CLIQUE.

Definition. The complete graph on n nodes is the graph $K_n = (V_n,E_n)$ with $V_n = \{o,\ldots,n-1\}$ and $E_n = \{(i,j) | i,j \in V_n \wedge i \neq j\}$.

Lemma 5.1. Let $G = (V_G,E_G)$ be an undirected graph, and $n \in N^+$.

K_n can be emulated on G with computation cost $\leq c$, if and only if G contains a clique with $\geq \lceil \frac{n}{c} \rceil$ nodes.

Proof. Suppose f emulates K_n on G with $cc(f) \leq c$. Then it is easily seen that $f(K_n)$ is a clique with at least $\lceil \frac{n}{c} \rceil$ nodes.

Suppose G contains a clique with $\lceil \frac{n}{c} \rceil$ nodes. We can map the nodes of K_n onto the nodes of this clique, such that every node of the clique has c or c-1 nodes mapped upon it. In this way an emulation f of K_n on G with $cc(f) = c$ is obtained. □

Corollary 5.2. Suppose there exists a polynomial time approximation algorithm A for MCE with $R_A(G,H) \le g(|V_G|,|V_H|)$, for a certain function g. Then there exists a polynomial time approximation algorithm A' for CLIQUE with $R_{A'}(G) \le g(K,|V_G|)$, for all connected graphs G of which the largest clique contains K nodes.

Proof. Let $G = (V_G,E_G)$ be given. By assumption one can calculate $A'(G) = \max \{\lceil \frac{i}{A(K_i,G)} \rceil \mid 1 \le i \le |V_G|\}$ in polynomial time. ($A(K_i,G)$ denotes the approximated computation cost of an emulation of K_i on G, produced by algorithm A).

Suppose the largest clique in G contains K nodes. A'(G) is the desired approximation of K: for all i, $cc(K_i,G) = \lceil \frac{i}{K} \rceil$. So $A(K_i,G) \ge \lceil \frac{i}{K} \rceil$ and $\lceil \frac{i}{A(K_i,G)} \rceil \le \lceil \frac{i}{\lceil \frac{i}{K} \rceil} \rceil \le K$. Furthermore $A(K_K,G) \le 1.\ g(K,|v_G|)$, so $A'(G) \ge \lceil \frac{K}{g(K,|V_G|)} \rceil$. This shows $R_{A'}(G) \le g(K,|V_G|)$. □

Corollary 5.3. R_{min} (CLIQUE) $\le R_{min}$ (MCE).

6. Relation of MCE with BALANCED COMPLETE BIPARTITE SUBGRAPH. Similar results can be obtained, relating approximation algorithms for MCE to approximation algorithms for the BALANCED COMPLETE BIPARTITE SUBGRAPH problem (BCBS). A (bipartite) graph $G=(V_G,E_G)$ is said to contain a BCBS with 2.K nodes, if there are disjunct $V_1,V_2 \subseteq V_G$, $|V_1|=|V_2|=K$ with for all $v_1 \in V_1$, $v_2 \in V_2$ $(v_1,v_2) \in E_G$. The BCBS problem asks to find the largest BCBS in a given bipartite graph G. (BCBS is NP-complete [4]).

Definition. The Balanced Complete Bipartite graph with 2.K nodes is the (undirected) graph $BCB_{2K} = (V_{2K},E_{2K})$, with $V_{2K} = \{v^i{}_j \mid i \in \{1,2\}, 1 \le j \le K\}$ and $E_{2K} = \{(v^1{}_{j_1}, v^2{}_{j_2}) \mid v^1{}_{j_1}, v^2{}_{j_2} \in V_{2K}\}$.

Lemma 6.1. Let $G = (V_G,E_G)$ be an undirected, bipartite graph and $K,c \in \mathbb{N}^+$, $\lceil \frac{K}{c} \rceil \ge 2$. BCB_{2K} can be emulated on G with computation cost $\le c$, if and only if G contains a BCBS with $2.\lceil \frac{K}{c} \rceil$ nodes.

Proof. Suppose f emulates BCB_{2K} on G with cc(f) $\le$ c. Take V_1 = $f(\{v^1_j \mid 1 \le j \le K\})$ and $V_2 = f(\{v^2_j \mid 1 < j \le K\})$. Now $|V_1| \ge \lceil \frac{K}{c} \rceil \ge 2$, $|V_2| \ge \lceil \frac{K}{c} \rceil \ge 2$, and every node in V_1 is connected to every node in V_2. So V_1 and V_2 must be disjunct (G is bipartite) and G has a BCBS with 2.K nodes. (The remainder of the proof is more or less similar to lemma 5.1.) □

Corollary 6.2. Suppose there exists a polynomial time approximation algorithm A for MCE with $R_A(G,H) \le g(|V_G|,|V_H|)$, for a certain function g. Then there exists a polynomial time approximation algorithm A' for BCBS with $R_{A'}(G) \le g(2K.|V_G|)$ for all connected, bipartite graphs G of which the largest BCBS consists of 2.K nodes.

Corollary 6.3. R_{min} (BCBS) $\le$ R_{min} (MCE).

7. Discussion. The results in this paper leave a number of questions open. For instance, we do not know whether $2 \le R_{min}$ (MCE) $< \infty$ or R_{min} (MCE) $= \infty$, i.e. whether there is a polynomial time approximation algorithm for MINIMUM COST EMULATION that finds emulations whose computation costs differ at most a constant factor from the optimal computation cost. We gave some evidence that it will be hard to find such an algorithm: the existence of such an algorithm would imply existence of polynomial time approximation algorithms for BANDWIDTH, CLIQUE and BALANCED COMPLETE BIPARTITE SUBGRAPH that give approximations to within a constant factor from the optimal solution. Up to now, it is not know whether such algorithms exist.

References.

[1] Bodlaender, H.L. and J. van Leeuwen, On the complexity of finding uniform emulations, Tech.Rep. RUU-CS-85-4, Dept. of Computer Science, University of Utrecht, Utrecht, 1985.

[2] Bodlaender, H.L., The complexity of finding uniform emulations on paths and ring networks, Tech.Rep. RUU-CS-85-5, Dept. of

Computer Science, University of Utrecht, Utrecht, 1985.

[3] Fishburn, J.P. and R.A. Finkel, Quotient networks, IEEE Trans. Comput. C-31 (1982) 288-295.

[4] Garey, M.R. and D.S. Johnson, Computers and Intractability, a guide to the theory of NP-completeness, W.H. Freeman, San Francisco, Calif., 1979.

[5] Gibbs, N.E., W.G. Poole and P.K. Stockmeyer, An algorithm for reducing the bandwidth and profile of a sparse matrix, SIAM J. Numer. Anal. 13 (1976) 236-250.

[6] Johnson, D.S., Approximation algorithms for combinatorial problems, J. Comp. Syst. Sci. 9 (1974) 256-278.

[7] Wai-Hung Liu and A.H.Sherman, Comparative analysis of the Cuthill-McKee and the reverse Cuthill-McKee ordering algorithms for sparce matrices, SIAM J. Numer. Anal. 13 (1976) 198-213.