The Complexity of Broadcasting in Planar and Decomposable Graphs *

Andreas Jakoby†
Rüdiger Reischuk
Christian Schindelhauer
Med. Universität zu Lübeck‡

Juli 1994

Abstract

Broadcasting in processor networks means disseminating a single piece of information, which is originally known only at some nodes, to all members of the network. The goal is to inform everybody using as few rounds as possible, that is to minimize the broadcasting time.

Given a graph and a subset of nodes, the sources, the problem to determine its specific broadcast time, or more general to find a broadcast schedule of minimal length has shown to be $\mathcal{NP}$-complete. In contrast to other optimization problems for graphs, like vertex cover or traveling salesman, little was known about restricted graph classes for which polynomial time algorithms exist, for example for graphs of bounded treewidth. The broadcasting problem is harder in this respect because it does not have the finite state property. Here, we will investigate this problem in detail and prove that it remains hard even if one restricts to planar graphs of bounded degree or constant broadcasting time. A simple consequence is that the minimal broadcasting time cannot even be approximated with an error less than $1/8$, unless $\mathcal{P} = \mathcal{NP}$.

On the other hand, we will investigate for which classes of graphs this problem can be solved efficiently and show that broadcasting and even a more general version of this problem becomes easy for graphs with good decomposition properties. The solution strategy can efficiently be parallelized, too. Combining the negative and the positive results reveals the parameters that make broadcasting difficult. Depending on simple graph properties the complexity jumps from $\mathcal{NC}$ or $\mathcal{P}$ to $\mathcal{NP}$.

Classification: graph algorithms, graph decomposition, computational complexity.

1 Introduction

Broadcasting in processor networks means disseminating a single piece of information, which is originally known only at some nodes, called the *sources,* to all members of the network. This is done in a sequence

*presented at the 20. Int. Workshop on Graph-Theoretic Concepts in Computer Science (WG'94), Herrsching, June 1994 to appear in Springer Lectures Notes in Computer Science

†supported by Deutsche Forschungsgemeinschaft, Grant DFG Re672/2, and German-Israeli Foundation for Scientific Research & Development, Grand GIF I-2007-199.06/91

‡Institut für Theoretische Informatik, Wallstraße 40, 23560 Lübeck,Germany
email: jakoby/reischuk/schindel@informatik.mu-luebeck.de

of rounds by pairwise message exchange over the communication lines of the network. In one round each processor can send a message to at most one of its neighbours. The goal is to inform everybody using as few rounds as possible. This number is called the *minimum broadcasting time* of the network.

Broadcasting is a basic task for multiprocessor systems that should be supported by the topology of the network. This problem has been studied extensively, mostly in the case of a single source – for an overview see [HHL88]. In several papers the broadcast capabilites of well known families of graphs like hypercubes, cube-connected-cycles, shuffle exchange graphs or de Bruijn graphs have been investigated and compared. In [HJM90] Hromkovič, Jeschke and Monien have studied the relation between the broadcasting time and the time for solving the related gossiping problem for special families of graphs.

On the other hand, one has tried to find optimal topologies for networks with a given number of nodes such that the broadcasting time is best possible. Here the worst case over all nodes as the single source should be minimized. The problem gets more complicated when restricting to graphs of bounded degree. In [LP88] Listman and Peters have studied several classes of bounded degree graphs in this respect, see also [BHLP92]. Balanced binary trees already achieve a broadcasting time of logarithmic order, therefore the question is the optimal constant factor in front of the logarithm.

In this paper we will investigate the optimization problem for arbitrary networks. That means, given a graph and a subset of nodes as sources, determine its specific broadcast time or more general find a broadcast schedule of minimal length. This problem in general is $\mathcal{NP}$-complete. We will show that this property remains even if one restricts to planar graphs of bounded degree or constant broadcast time. Furthermore, the problem cannot be solved approximately with an arbitrary precision unless $\mathcal{P} = \mathcal{NP}$.

On the other hand, we will investigate for which classes of graphs this problem can be solved efficiently. All what seems to be known was that broadcasting is easy for trees shown by Slater, Cockayne, and Hedetniemi in [SCH81]. Many combinatorial optimization problems for graphs have been shown to be solvable in polynomial sequential time and even in polylogarithmic parallel time for more general classes of graphs: graphs of bounded treewidth (see for example the paper by Arnborg, Lagergren and Seese [ALS91]) and graphs of small connectivity ([R91a]) – an overview can be found in [R91b].

The broadcasting problem seems to be more difficult in this respect since it does not have the finite-state-property or a bounded number of equivalence classes. Thus the methods of [ALS91] and [R91a] are not directly applicable. Still, modifying the framework developed in [R91a] we can show that broadcasting becomes easy for graphs with good decomposition properties. For this purpose we consider edge decompositions of a graph. This notion seems more appropriate for the broadcasting problem than the dual approach based on vertex decompositions. Furthermore, a careful inspection of the possibilities how information can flow within a component and between different components of a graph will be required. For the internal flow components that are connected behave most favourably, but in general connectivity cannot always be achieved by an edge decomposition that generates small components. Disconnected components complicate the analysis of the broadcasting problem significantly, whereas most other optimization problems become easier in such a case. The algorithm even works for a more general version of the broadcasting problem. Furthermore, it can be parallelized efficiently to yield $\mathcal{NC}$-solutions.

As a conclusion we can say that combining these new negative and positive results the parameters that make broadcasting difficult are determined quite precisely. The complexity of this problem jumps from $\mathcal{P}$ to $\mathcal{NP}$ depending on the internal structure of the networks.

2 Definitions and Previous Results

A formal definition of the broadcasting problem can be given as follows.

Definition 1 *Let* $G = (V, E)$ *be a (directed) graph with a distinguished subset of vertices* $V_0 \subseteq V$ *, the* **sources***, and* $T^* \in \mathbb{N}$ *be a deadline. The task is to decide whether there exists a* **broadcast schedule***, that is a sequence of subsets of edges* $E_1, E_2, \ldots, E_{T^*-1}, E_{T^*}$ *with the property* $V_{T^*} = V$ *, where for* $i > 0$ *we define* $V_i := V_{i-1} \cup \{ v \mid (u,v) \in E_i \text{ and } u \in V_{i-1} \}$ *and require* $E_i \subseteq \{ (u,v) \in E \mid u \in V_{i-1} \}$ and $\forall u \in V_{i-1} : | E_i \cap (\{u\} \times V)| \leq 1$.
Let us distinguish between the general multiple source problem **MB** *and the restricted version with only a single source* **SB***.* □

The meaning of the sets E_i and V_i is the following: V_i denotes the set of nodes that have received the broadcast information by round i . For $i = 0$ this is just the set of sources. By the deadline T^* the set V_{T^*} should include all nodes of the network. E_i is the set of edges that are used to send information at round i , where each processor $u \in V_{i-1}$ can use at most one of its outgoing edges.

MB (denoted by ND49 in [GJ79]) has shown to be $\mathcal{NP}$-complete.

Theorem A: *MB for graphs with unbounded degree is* $\mathcal{NP}$ *-complete, even if restricted to a fixed deadline* $T^* \geq 4$ *.*

For a fixed deadline the number of sources obviously has to grow linearly in the size of the whole graph. But even the single source problem is difficult, in this case the deadline has to grow at least logarithmically.

Theorem B: *SB for graphs with unbounded degree is* $\mathcal{NP}$ *-complete.*

The proofs of both results were published by Slater, Cockayne, and Hedetniemi ([SCH81]). For the second result, their reduction of the 3-dimensional matching problem to SB requires a deadline of order $\sqrt[3]{|V|}$ for the broadcast problem. Furthermore, in the same paper it is shown

Theorem C: *SB can be solved in linear time for trees. This also holds for the constructive version of this problem finding an optimal broadcast schedule.*

3 New Results

All theorems above can be improved significantly. For the lower bounds it suffices to consider undirected graphs, the upper bounds given below also hold for the more general case of directed graphs.

3.1 Lower Bounds

Designing more complicated reductions of the 3-dimensional matching problem and the 3-SAT problem we can show

Theorem 1 *MB restricted to planar graphs with bounded degree at least* 4 *and a fixed deadline* T^* *at least* 3 *is* $\mathcal{NP}$ *-complete.*

The reduction to prove this result uses graphs of a specific kind that are guaranteed to have a broadcast schedule of length 4. Now consider an approximation algorithm that for a network G gives an estimate of its minimum broadcast time $T(G)$. The estimate $\tilde{T}(G)$ may be an arbitrary real number, but is required to be within a precision γ of the correct value: $(1+\gamma)^{-1} \cdot T(G) \leq \tilde{T}(G) \leq (1+\gamma) \cdot T(G)$. In this case, any estimate with a precision $\gamma \leq \frac{1}{8}$ could be used to solve the decision problem. Thus we also get

Theorem 2 *There exists no polynomial-time approximating algorithm for MB with a precision* $1/8$, *unless* $\mathcal{P} = \mathcal{NP}$.

If for this minimization problem one restricts to integer values and approximations from above then the statement of the theorem can be improved from precision $1/8$ to any value less than $1/3$.

Broadcasting with a single source does not become substantially easier, even for bounded degree graphs with a logarithmic diameter.

Theorem 3 *SB restricted to graphs* $G = (V, E)$ *with bounded degree at least* 3 *is* $\mathcal{NP}$*-complete, even if the deadline grows at most logarithmically in the size of the graph.*

Also planarity does not make things much simpler as the following result shows.

Theorem 4 *SB restricted to planar graphs* $G = (V, E)$ *of degree* 3 *is* $\mathcal{NP}$*-complete (in this case the deadline grows like* $\sqrt{|V|}$ *).*

3.2 Upper Bounds

On the positive side, we will extend the classes of graphs for which the broadcasting problem can be solved fast. The general approach is to partition a given graph into smaller components and then solve a generalized broadcast problem on each component separately and later combine the partial solutions. A split into smaller components can be done by removing edges (edge separators) or by removing nodes (node separators). In general, edge separation seems to be a weaker notion since graphs that contain nodes of very large degree cannot be split into small pieces by separators of small size, whereas small node separators may exist (consider a star). However, for graphs of bounded degree both notions are closely related. Here we will restrict ourself to edge separators because then the analysis of the broadcast problem is somewhat easier. But similar algorithmic techniques can also be used for graph decompositions based on node separators.

For the purpose of decomposing a graph G it suffices to consider only the case of undirected graphs. Thus, if G is directed in the following definition we simply mean the corresponding undirected graph.

Definition 2 *A graph* $H = (V_H, E_H)$ *is an* **edge decomposition graph** *of a graph* $G = (V, E)$ *if the following conditions hold:*

- *The nodes* G_i *of* V_H *represent induced subgraphs* $G_i = (V_i, E_i)$ *of* G *such that the* V_i *are pairwise disjoint and* $V = \bigcup_{G_i \in V_H} V_i$.
- $\{G_i, G_j\} \in E_H$ *iff there is an edge between a node of* G_i *and a node of* G_j.

H is called an **edge decomposition tree** *of G if H is a tree.*
Define the **cut** *of an edge $\{G_i, G_j\}$, the cut of a node G_i, and the cut of H as those edges of G that connect G_i and G_j, resp. connect G_i to other components, or connect any pair of components:*

$$\begin{aligned} \mathrm{cut}(G_i, G_j) &:= \{\, \{u,v\} \in E \mid u \in V_i \text{ and } v \in V_j \} \quad \textit{for } i \neq j \ , \\ \mathrm{cut}(G_i) &:= \bigcup_{\{G_i,G_j\} \in E_H} \mathrm{cut}(G_i, G_j) \qquad \text{and} \qquad \mathrm{cut}(H) := \bigcup_{G_i \in V_H} \mathrm{cut}(G_i) \ . \end{aligned}$$

A graph $G = (V, E)$ is $(\boldsymbol{\kappa}, \boldsymbol{\mu}, \boldsymbol{c})$**–edge decomposable** *if there exists an edge decomposition graph $H = (V_H, E_H)$ such that for all $G_i \in V_H$:*

$$|\,\mathrm{cut}(G_i)\,| \ \leq \ \kappa \ , \qquad |\,V_i\,| \ \leq \ \mu \qquad \text{and} \qquad \mathtt{cc}(G_i) \leq c \ ,$$

where $\mathtt{cc}(G_i)$ denotes the number of connected component of G_i. □

The decomposition process partitions a graph into different components. Each component G_i itself may be connected or fall into several connected components. When constructing an optimal broadcast schedule this issue turns out to be important. For example, a $\sqrt{n} \times \sqrt{n}-$ 2-dimensional grid is $(O(\sqrt{n}), (O(\sqrt{n}), 1)$–edge decomposable into a tree. For a circle of length n the parameters are $(4, 2, 2)$. Taking the number of connected components within each component into consideration will allow us to determine good upper bounds for the algorithmic effort to solve the broadcasting problem.

Other approaches have been proposed how to decompose a graph into smaller components, based on the notions of treewidth ([RS86]), see for example [ALS91, BK91, L90], and k-connected components using minimal node separators [H90, HR89]. In general, it is $\mathcal{NP}$-complete to construct optimal decompositions with respect to the relevant parameters. However, suboptimal solutions can be found in polynomial time.

In the following we assume that an arbitrary edge decomposition of the network is given and analyse only the complexity of constructing an optimal broadcast schedule with the help of that decomposition. Note that the broadcast time will always be best possible, only the complexity of finding the schedule will depend on the parameters of the decomposition.

Theorem 5 *For a graph $G = (V, E)$ of maximal degree d with a given (κ, μ, c)–edge decomposition tree the MB-problem can be solved in time*

$$\begin{aligned} & \mathrm{O}\Big(|V|^c \cdot (2(\kappa + \mu))^{4\kappa} \cdot (|V| \cdot (\mu + \kappa)^3 \cdot (d+1)^{\mu} + |E|)\Big) \\ \leq \quad & \exp \mathrm{O}\Big(c \cdot \log |V| \ + \ \kappa \cdot \log(\kappa + \mu) \ + \ \mu \cdot \log d\Big) \ . \end{aligned}$$

The algorithm we have designed actually works for a more general version of the broadcasting problem, in which the sources may receive the broadcast information at different rounds and each node of the network may have its individual deadline. Let us call this the *general broadcasting problem* **GB** (see definition 3).

The time bound becomes polynomial for classes of graphs that can be decomposed into smaller components using not too large separators. Let **llog** and **lllog** denote the logarithm function iterated twice, resp. three times.

Corollary 1 *Restricted to graphs $G = (V, E)$ with*

- $(\mathrm{O}(\frac{\log n}{\mathrm{llog}\ n}), \mathrm{O}(\frac{\log n}{\mathrm{llog}\ n}), \mathrm{O}(1))$ *–edge decomposition trees or*

- *to graphs with bounded degree and* $(\mathrm{O}(\frac{\log n}{\mathrm{llog}\ n}), \mathrm{O}(\log n), \mathrm{O}(1))$ *–edge decomposition trees*

the MB–problem (and even the GB–problem) can be solved in polynomial time.

So far, we have only considered the decision version of the MB-problem, resp. the task to determine the minimal length of a broadcast schedule. But applying ideas similar to the one in [R91a] one can also design an algorithm for constructing an optimal broadcast schedule by using the same techniques as for the decision problem.

Theorem 6 *Constructing an optimal broadcast schedule can be done in the same time bounds as stated for the decision problem in Theorem 5.*

Using the machinery developed in [R91a] we can also derive a fast and processor efficient parallel algorithm. Even if the decomposition tree is not nicely balanced using path compression techniques the problem can be solved with a logarithmic number of iterations (with respect to the number of components) on a standard parallel machine model like a PRAM with concurrent write capabilities. The basic task one has to solve is how a chain of 2 components can be replaced by a single component that externally behaves identically with respect to broadcasting.

Theorem 7 *For a graph* $G = (V, E)$ *of maximal degree* d *with a given* (κ, μ, c) *–edge decomposition tree MB can be solved in parallel time*

$$O\Big(\log|V| \cdot (c \cdot \log|V| \ + \ \kappa \cdot \log(\kappa + \mu) \ + \ \mu \cdot \log d)\Big)$$

with a processor bound of $\quad \exp O\Big(c \cdot \log|V| \ + \ \kappa \cdot \log(\kappa + \mu) \ + \ \mu \cdot \log d\Big)$.

For nicely decomposable classes of graphs these bounds put MB into $\mathcal{NC}$.

Corollary 2 *Restricted to graphs* $G = (V, E)$ *with*

- $(\mathrm{O}(\frac{\log n}{\mathrm{llog}\ n}), \mathrm{O}(\frac{\log n}{\mathrm{llog}\ n}), \mathrm{O}(1))$ *–edge decomposition trees or*
- *to graphs with bounded degree and* $(\mathrm{O}(\frac{\log n}{\mathrm{llog}\ n}), \mathrm{O}(\log n), \mathrm{O}(1))$ *–edge decomposition trees*

MB is in $\mathcal{NC}^2$ *.*

All these bounds apply to the GB–problem as well. Also the constructive variant can be solved with the same effort.

In the remaining part of this extended abstract we can only present some of the basic ideas to prove these results. For a complete version see [JRS93].

4 Proofs of the Lower Bounds

The $\mathcal{NP}$-hardness of the minimum broadcasting time problem will be obtained by a reduction of the 3DM problem (3-Dimentional Matching, see [GJ79]). The graph $G' = G(A, B, C, M)$ of an instance (A, B, C, M) with $M \subseteq A \times B \times C$ of the 3DM problem is defined as follows: Each element of the sets A, B, and C and each triple of M is represented by a vertex. The membership relation between set elements and triples defines the edges between these vertices.

$$
\begin{aligned}
G' &:= (V', E') \text{ with } \quad V_A := \{ \alpha_x \mid x \in A\}, \quad V_B := \{ \beta_x \mid x \in B\}, \\
V_C &:= \{ \gamma_x \mid x \in C\}, \quad V_M := \{ \mu_y \mid y \in M\}, \quad V' := V_A \cup V_B \cup V_C \cup V_M, \\
E' &:= \{ (\mu_y, v_x) \mid y \in M, \ v_x \in V_A \cup V_B \cup V_C \text{ and } x \in y\}
\end{aligned}
$$

The reduction will use a restricted planar version of the 3DM problem, which is still $\mathcal{NP}$-complete [DF86]. For an instance (A, B, C, M) the following properties are required:

- $G(A, B, C, M)$ is planar.
- For each element x of $A \cup B \cup C$ there are at most 3 triples in M containing x (thus, $|M|$ is bounded by $3q$ where $q := |A| = |B| = |C|$).

Proof of Theorem 1 and 2: Planar 3DM will be reduced to the broadcasting problem as follows: Let (A, B, C, M) be an instance of 3DM with $|A| = q$ and let $G' = G(A, B, C, M)$ be the matching graph. The corresponding broadcasting graph G is obtained by replacing each node $\alpha_i \in V_a$ of G' by a chain $\alpha_{i,1}$, $\alpha_{i,2}$, and $\alpha_{i,3}$ of length 3 (see Figure 1). The other nodes and edges remain unchanged. $V_{A,1}$ is chosen as the set of sources, and the deadline is set to 3.

Observe that G has degree 4 and is planar if G' is planar.

Lemma 1 *G has a broadcast schedule of length 3 iff the q sources in round 1 send the information to a subset of the nodes in V_M that defines a matching.*

Theorem 1 and 2 now follow easily. ∎

Proof of Theorem 3: Consider the tree $T_{q,t}$ in figure 2, where its root is the only source and $q \leq \exp t/3$ outgoing edges α_i. It has the following properties:

- Within $\delta := 3\lceil\log q\rceil - 1$ rounds $T_{q,t}$ can reach a state such that in the next round the information of the source can be propagated simultaneously over all outgoing edges α_i.
- If a broadcast schedule for $T_{q,t}$ finishes by round t then none of these edges can propagate the information before round $\delta + 1$.

Connect each leaf of the tree with a node of $V_{A,1}$ of the graph G defined above and call this new graph G'. Let $t := 3\lceil\log q\rceil + 3$ and let w be the the root of $T_{q,t}$. Then, G' with source w has a broadcast schedule of length at most $3\log q + 3$ iff M contains a matching. The resulting graph has degree 5, but is not necessarily planar. By additional effort G' can be modified to decrease the node degree to 3. ∎

Proof of Theorem 4: To achieve planarity in the single source case we construct a direct reduction for the satisfiability problem 3SAT. Although a planar version of 3SAT remains $\mathcal{NP}$-complete it does not help much in this case because the connections to the source will destroy the planarity. A simple exchange of an edge crossing by a planar subgraph with 4 input/output-edges also does not seem to work for the broadcasting problem.

We have found a way to make such a replacement legal under special circumstances, namely if the direction of the information flow over the edges is known in advance and if at most one of the two input edges is used. The first property can easily be achieved for 3SAT, while the second requires special coding tricks. The details are quite long and can be found in the full version [JRS93]. ∎

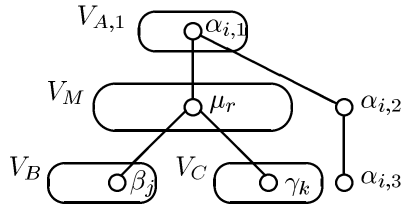

Figure 1: *The broadcasting graph corresponding to an instance of the 3DM problem.*

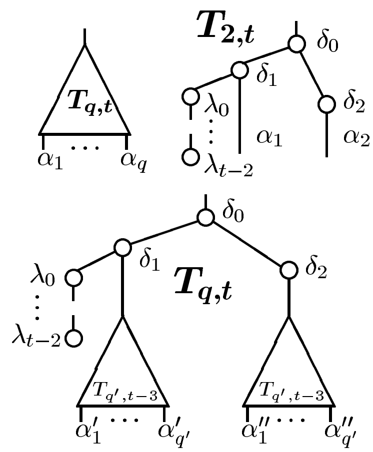

Figure 2: *The recursive construction of the trees* $T_{q,t}$ *. The* λ_i *form a chain that forces the left son of* δ_0 *to be informed first in order to obey the deadline* t *. For* $q > 2$ *the tree* $T_{q,t}$ *is obtained from 2 copies of* $T_{\lceil q/2 \rceil, t-3}$.

5 Efficient Algorithms for Decomposable Graphs

We start with a generalization of the broadcast problem. So far, each source node has got the broadcast information at round 0. In the more general case, a source may get the information with an arbitrary delay of $\sigma(v)$ rounds. Furthermore, we require that the information has to reach each node v by round $\rho(v)$, instead of a global deadline T^* identical for all nodes. This generalization may be of less interest with respect to practical applications. Nevertheless, it is necessary in order to apply an approach based on graph decompositions, as it has been for several other graph theoretic decision and optimization problems.

Definition 3 *General Broadcasting Problem* **GB***:*
Given a graph $G = (V, E)$ *and two partial functions* $\sigma, \rho : V \to \mathbb{N}$ *, decide whether there exists a broadcast schedule* $E_1, E_2, \ldots,$ *with*

$$V_i = V_{i-1} \cup \{ v \mid (u,v) \in E_i \text{ and } u \in V_{i-1} \} \cup \{ v \in V \mid \sigma(v) = i \} ,$$
$$E_i \subseteq \{ (u,v) \in E \mid u \in V_{i-1} \} \quad \text{and} \quad \forall u \in V_{i-1} : | E_i \cap (\{u\} \times V)| \leq 1$$

such that $\forall v \in V : v \in V_{\rho(v)}$. □

The following exponential upper bounds can be achieved by standard enumeration methods.

Lemma 2 *GB can be solved in time* $O(|V|^3 (d+1)^{|V|})$ *, where* d *denotes the degree of the graph* $G = (V, E)$ *.* ∎

Lemma 3 *GB restricted to graphs* $G = (V, E)$ *with maximum degree* d *can be solved by a CRCW–PRAM with* $O(|V| \cdot (d+1)^{|V|})$ *processors in time* $O(|V|^2)$. ∎

Definition 4 *Let a graph $G = (V, E)$ and a broadcast schedule $\mathcal{E} = E_1, E_2, \ldots$ for G be given. The round in which a node receives the information to be broadcasted first is called its* **starting round**. *A node u is called* **active** *at a round t if sends the information to another node v ($\{u, v\} \in E_t$) at this round. A node u is called* **busy** *if it is active after its starting round until all its neighbors have received the information. $\mathcal{E}$ is* **busy** *if all nodes are busy and no node receives the broadcast information several times.* □

Observe that each broadcast schedule can easily be tranformed into a busy broadcast schedule of the same or smaller length.

Proof of Theorem 5: Let $H = (V_H, E_H)$ be a (κ, μ, c)–edge decomposition tree of the graph $G = (V, E)$ with $V_H = \{G_1, \ldots, G_k\}$. Figure 3 shows a component G_2 which is connected to three other components G_1, G_3 and G_4.

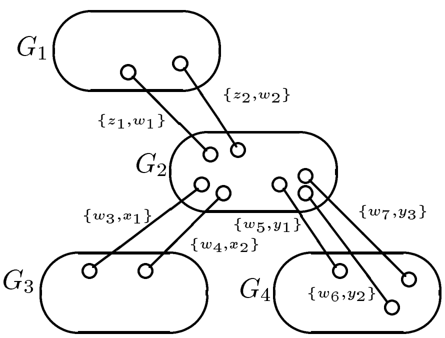

Figure 3: A node G_2 of an edge decomposition tree and its neighbors.

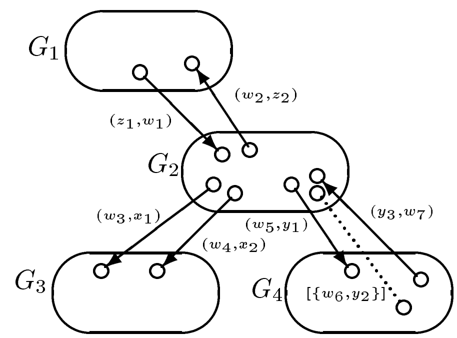

Figure 4: A possible information flow within a broadcast schedule: from G_2 to other components in one or in both directions, some edges may not be used.

Let $\mathcal{G}_i := \{ G_i^1, \ldots, G_i^{c_i} \}$ with $\boldsymbol{G_i^a} = (V_i^a, E_i^a)$ be the set of connected components of the G_i and define $\mathrm{cut}(G_i^a)$ as the set of edges of $\mathrm{cut}(G_i)$ with one endpoint in G_i^a.

$$\mathrm{cut}(G_i^a, G_j) \;:=\; \mathrm{cut}(G_i, G_j) \cap \mathrm{cut}(G_i^a), \quad \mathrm{cut}(G_i^a, G_j^b) := \mathrm{cut}(G_i^a, G_j) \cap \mathrm{cut}(G_i^b) .$$

To describe a broadcasting schedule of G, each edge $\{u, v\}$ of G is labelled by a tupel $(\tau(u, v), r(u, v))$. The first value $\tau(u, v)$ denotes the round when this edge is used and the second $r(u, v)$ the direction (from u to v or form v to u if u and v are connected in both directions). If this edge is not used we set $\tau(u, v) := -1$.

If we restrict $\mathcal{E}$ to $\mathrm{cut}(G_i^a, G_j^b)$ it suffices to denote the round $\tau(G_i^a, G_j^b)$ when an edge of $\mathrm{cut}(G_i^a, G_j^b)$ is used for the first time and for each other edge $\{u, v\}$ with $\tau(u, v) \geq 0$ the **relative round** $\boldsymbol{\widehat{\tau}(u, v)} := \tau(u, v) - \tau(G_i^a, G_j^b)$. If no edge in $\mathrm{cut}(G_i^a, G_j^b)$ is used let $\tau(G_i^a, G_j^b) := -1$. For G_i^a let $\boldsymbol{\tau(G_i^a)}$ be the first round when a node of G_i^a receives the broadcast information. Similarly define $\boldsymbol{\widehat{\tau}(G_i^a, G_j^b)} := \tau(G_i^a, G_j^b) - \tau(G_i^a)$ if $\tau(G_i^a, G_j^b) \geq 0$, else let $\widehat{\tau}(G_i^a, G_j^b) := -1$.

The following two lemmata show that with the help of the relative rounds $\widehat{\tau}$ the number of possible protocolls of information exchange between two components can be bounded quite substantially. This property will be basic for the time efficiency of the algorithm described below. It serves as a replacement for the "finite state property" shared by graph theoretic problems like independent set or vertex cover, for which efficient solutions based on the decompositional approach have been given before.

Lemma 4 *If $\mathcal{E}$ is a busy broadcast schedule then for all $\{u, v\} \in \mathrm{cut}(G_i^a, G_j^b)$ holds: the numbers $\widehat{\tau}(G_i^a, G_j^b) + \widehat{\tau}(u, v)$ and $\widehat{\tau}(G_j^b, G_i^a) + \widehat{\tau}(u, v)$ are smaller than $|\mathrm{cut}(G_i)| + |V_i| \;\leq\; \kappa + \mu$.* ∎

Let G_i, G_j be neighbouring components and G_i^a and G_j^b connected components of G_i, resp. G_j with nonempty $\mathrm{cut}(G_i^a, G_j^b)$. Define a **state** between the two neighbours as a tupel

$$\boldsymbol{\gamma_{i,j}} \;:=\; \left(\left[\widehat{\tau}(G_i^a, G_j^b), \widehat{\tau}(G_j^b, G_i^a) \mid \mathrm{cut}(G_i^a, G_j^b) \neq \emptyset\right] \,,\; \left[(\widehat{\tau}(e), r(e)) \mid e \in \mathrm{cut}(G_i, G_j)\right]\right).$$

Figure 4 illustrates a complex information flow between a component and its neighbours. The **surface** $\boldsymbol{\Gamma_{i,j}}$ is the set of all possible states $\gamma_{i,j}$ of busy broadcast schedules. A **state** S_i of a component G_i is a vector consisting of the starting round $\tau(G_i^a)$ for all connected components of G_i and tupels $\gamma_{i,j}$ for all neighbouring components of G_i. As above, let $\boldsymbol{\Gamma_i}$ be the set of all possible states S_i that may appear in busy schedules.

Lemma 5 *For a component G_i with cutsize $|\mathrm{cut}(G_i)| \leq \kappa_i$, size $|V_i| \leq \mu_i$ and $c_i = \mathsf{cc}(G_i)$ connected components, the size of Γ_i is bounded by $\gamma(\kappa_i, \mu_i, c_i) \;:=\; |V|^{c_i} \cdot (2(\kappa_i + \mu_i))^{3\kappa_i}$.*

Proof:

$$\begin{aligned} |\Gamma_i| &\leq \prod_{G_i^a \in \mathcal{G}_i} |V| \cdot \prod_{\mathrm{cut}(G_i,G_j)\neq\emptyset} |\,\Gamma_{i,j}\,| \\ &\leq |V|^{c_i} \cdot \prod_{\mathrm{cut}(G_i,G_j)\neq\emptyset} (\kappa_i + \mu_i)^{2|\mathrm{cut}(G_i,G_j)|} \cdot (2(\kappa_i + \mu_i))^{|\mathrm{cut}(G_i,G_j)|} \\ &\leq |V|^{c_i} \cdot (2(\kappa_i + \mu_i))^{3\kappa_i}\,. \end{aligned}$$

∎

The following strategy solves the minimum broadcasting time problem for graphs $G = (V, E)$ with a given (κ, μ, c)–edge decomposition tree $H = (V_H, E_H)$. Let $\boldsymbol{\Delta(G_i, S_i)}$ denote the minimal schedule length of the local broadcast problem for the graph G_i and external information exchange as specified by state S_i ($= \infty$ if there is no schedule for state S_i). Observe that this value is independent of the structure of G outside of G_i.

Step 1: For each component G_i and each state $S_i \in \Gamma_i$ determine $\Delta(G_i, S_i)$.

Step 2: Choose an arbitrary component G_r and declare G_r as the root of H. Let $G_{i,0}$ be the father of G_i in H according to the orientation with respect to G_r. Let G_i^* denote the subgraph of G containing G_i and all its descendents. Evaluate the function $\Delta(G_i^*, S_{i,0})$ for all G_i and $S_{i,0}$ starting with the leaf components of H.

Lemma 6 *Let $G_{i,1}, \ldots, G_{i,\ell_i}$ denote the sons of G_i and let $S_{i,j}$ be a state connecting G_i and $G_{i,j}$. For all G_i the minimal deadline for the general broadcast problem for G_i^* with respect to external information exchange $S_{i,0}$ can be computed as*

$$\Delta(G_i^*, S_{i,0}) \;=\; \min_{\substack{S_i=(\tau_1,\ldots,\tau_{cc(G_i)},\\ S_{i,0},\ldots,S_{i,\ell_i})\in\Gamma_i}} \; \max_{j\in[1\ldots l_i]} \{\Delta(G_{i,j}^*, S_{j,i})\} \,\cup\, \{\Delta(G_i, S_i)\}\,.$$

Proof: This property can be shown by induction on the depth of the subgraphs G_i. ∎

Therefore, $\Delta(G_r^*) \;=\; \Delta(G_r^*, \lambda)$ denotes the minimal schedule length for the graph G itself.

The correctness of step 1 follows directly from the definition of a surface and the definition of the general broadcast problem. The correctness of step 2 follows directly from Lemma 6.

According to lemma 2 for each G_i and S_i the computation of $\Delta(G_i, S_i)$ requires at most $\mathrm{O}\left((\mu_i+\kappa_i)^3 \cdot (d+1)^{\mu_i}\right)$ many steps. Since lemma 5 gives a bound on the number of states of each component step 1 can be executed in time

$$\sum_{G_i} \gamma(\kappa_i, \mu_i, c_i) \cdot \mathrm{O}\Big((\mu_i+\kappa_i)^3 (d+1)^{\mu_i}\Big) \leq \mathrm{O}\Big(|V|^{c+1} (2(\kappa+\mu))^{3\kappa} \cdot (\mu+\kappa)^3 \cdot (d+1)^{\mu}\}\Big).$$

The computation of $\Delta(G_i^*, S_{i,0})$ is independent of the remaining structure of G. Thus for each G_i, given all values $\Delta(G_i, S_i)$ and $\Delta(G_{i,j}^*, S_{j,i})$ the computation of all $\Delta(G_i^*, S_{i,0})$ can be executed in time $\mathrm{O}(\gamma(\kappa_i, \mu_i, c_i) \cdot \ell_i)$. Summing up over all G_i gives the bound

$$\sum_{G_i} \mathrm{O}(\gamma(\kappa_i, \mu_i, c_i) \cdot \ell_i) \ \leq \ \mathrm{O}\Big(\gamma(\kappa, \mu, c) \cdot \sum_{G_i} \ell_i\Big) \ \leq \ \mathrm{O}(|E| \cdot |V|^c \cdot (2(\kappa+\mu))^{3\kappa}) \ .$$

∎

By using tree contraction methods the evaluation of the Δ -function can also be done in parallel requiring only a logarithmic number of iterations. The details are described in [R91a].

Conclusions

We have shown that the single source broadcasting problem remains hard for planar graphs with high internal connectivity. After we have presented the lower bound in Theorem 1 Middendorf was able to improve our construction to degree 3 and deadline 2 [M93].

On the other hand, even a much more general version with many sinks and individual deadlines can be solved efficiently on graphs with moderate internal connections.

The algorithms described are based on edge decomposition of graphs. The same technique with a slightly worse time bound due to a larger number of states also works for other graph decomposition methods, for example based on treewidth. These issues are included in the full paper [JRS93].

References

[ALS91] S. Arnborg, J. Lagergren, and D. Seese, *Easy Problems for Tree-Decomposable Graphs* J. Algorithms 12, 1991, 308-340.

[BHLP92] J.-C. Bermond, P. Hell, A. Liestman, and J. Peters, *Broadcasting in Bounded Degree Graphs*, SIAM J. Disc. Math. 5, 1992, 10-24.

[BK91] H. Bodlaender, T. Kloks, *Better Algorithms for the Pathwidth and Treewidth of Graphs,* Proc. 18 ICALP, 1991, 544-555.

[DF86] M. Dyer and A. Frieze, *Planar 3DM Is $\mathcal{NP}$ -Complete*, J. Algorithms 7, 1986, 174-184.

[GJ79] M. Garey and D. Johnson, *Computers and Intractability, A Guide To the Theory of $\mathcal{NP}$ -Completeness*, Freeman 1979.

[H90] W. Hohberg, *The Decomposition of Graphs into k –Connected Components for Arbitrary k ,* Technical Report, TH Darmstadt, 1990.

[HHL88] S. Hedetniemi, S. Hedetniemi, and A. Liestman, *A Survey of Gossiping and Broadcasting in Communication Networks*, Networks 18, 1988, 319-349.

[HJM90] J. Hromkovič, C.–D. Jeschke, and B. Moinien, *Optimal Algorithms for Dissemination of Information in Some Interconnection Networks*, Proc. 15. MFCS, 1990, 337-346.

[HR89] W. Hohberg, R. Reischuk, *Decomposition of Graphs – A Uniform Approach for the Design of Fast Sequential and Parallel Algorithms on Graphs,* Technical Report, TH Darmstadt, 1989.

[JRS93] A. Jakoby, R. Reischuk, C. Schindelhauer, *The Complexity of Broadcasting in Planar and Decomposable Graphs,* Technical Report, TH Darmstadt, 1993.

[L90] J. Lagergren, *Efficient Parallel Algorithms for Tree-Decomposition and Related Problems,* Proc. 31. FoCS, 1990, 173-182.

[LP88] A. Liestman and J. Peters, *Broadcast Networks of Bounded Degree*, SIAM J. Disc. Math. 4, 1988, 531-540.

[M93] M. Middendorf, *Minimum Broadcast Time is* $\mathcal{NP}$ *–complete for 3–regular planar graphs and deadline 2*, IPL 46, 1993, 281-287.

[R91a] R. Reischuk, *An Algebraic Divide–and–Conquer Approach to Design Highly Parallel Solution Strategies for Optimization Problems on Graphs*, Technical Report, TH Darmstadt, 1991.

[R91b] R. Reischuk, *Graph Theoretical Methods for the Design of Parallel Algorithms*, Proc. 8. FCT, 1991, 61-67.

[RS86] N. Robertson, P. Seymour, *Graph Minors II. Algorithmic Aspects of Tree-Width*, J. Alg. 7, 1986, 309-322.

[SCH81] P. Slater, E. Cockayne, and S. Hedetniemi, *Information Dissemination in Trees*, SIAM J. Comput. 10, 1981, 692-701.