The Complexity of Optimal Addressing in Radio Networks

Article *in* IEEE Transactions on Communications · December 1982

DOI: 10.1109/TCOM.1982.1095431 · Source: IEEE Xplore

Kenneth Steiglitz
Princeton University

REFERENCES

[1] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications.* Englewood Cliffs, NJ: Prentice-Hall, to be published, 1983.

[2] R. J. Benice and A. H. Frey, "An analysis of retransmission systems," *IEEE Trans. Commun. Technol.*, vol. COM-12, pp. 135–145, Dec. 1964.

[3] ——, "Comparison of error control techniques," *IEEE Trans. Commun. Technol.*, vol. COM-12, pp. 146–154, Dec. 1964.

[4] E. Y. Rocher and R. L. Pickholtz, "An analysis of the effectiveness of hybrid transmission schemes," *IBM J. Res. Develop.*, pp. 426–433, July 1970.

[5] A. R. K. Sastry, "Improving ARQ performance on satellite channels under high error rate conditions," *IEEE Trans. Commun.*, vol. COM-23, pp. 436–439, Apr. 1975.

[6] C. Fujiwara, M. Kasahara, K. Yamashita, and T. Namekawa, "Evaluations of error control techniques in both independent-error and dependent-error channels," *IEEE Trans. Commun.*, vol. COM-26, pp. 785–793, June 1978.

[7] I. G. Sitglitz, "Sequential decoding with feedback," Ph.D. dissertation, Massachusetts Inst. Technol., Lexington, Aug. 1963.

[8] H. Yamamoto and K. Itoh, "Viterbi decoding algorithm for convolutional codes with repeat request," *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 540–547, Sept. 1980.

[9] A. Drukarev and D. J. Costello, Jr., "ARQ error control using sequential decoding," in *Proc. IEEE Int. Conf. Commun.*, Seattle, WA, June 1980.

[10] ——, "Hybrid ARQ error control using sequential decoding," *IEEE Trans. Inform. Theory*, to be published.

[11] R. W. Lucky, J. Salz, and E. J. Weldon, *Principles of Data Communications.* New York: McGraw-Hill, 1968.

[12] R. E. Kahn, S. A. Gronemeyer, J. Burchfiel, and R. C. Kunzelman, "Advances in packet radio technology," *Proc. IEEE*, vol. 66, pp. 1468–1496, Nov. 1978.

[13] S. Lin, *An Introduction to Error-Correcting Codes.* Englewood Cliffs, NJ: Prentice-Hall, 1970.

[14] J. M. Morris, "Optimal blocklengths for ARQ error control schemes," *IEEE Trans. Commun.*, vol. COM-27, pp. 488–493, Feb. 1979.

[15] W. W. Peterson and E. J. Weldon, *Error-Correcting Codes*, 2nd ed. Cambridge, MA: M.I.T. Press, 1972.

[16] I. Richer, "Sequential decoding with a small digital computer," Massachusetts Inst. Technol., Lexington, Tech. Rep. 491, Jan. 1972.

The Complexity of Optimal Addressing in Radio Networks

SAM TOUEG AND KENNETH STEIGLITZ, FELLOW, IEEE

Abstract—**We consider the complexity of finding optimal fixed- or variable-length unambiguous address codes for the nodes of a packet radio network. For fixed-length codes this problem is proved to be NP-complete, and its complexity for variable-length codes is still unknown. Some suboptimal heuristic algorithms are proposed.**

Paper approved by the Editor for Computer Communication of the IEEE Communications Society for publication after presentation at the IEEE Conference on Information Sciences and Systems, Baltimore, MD, March 1979. Manuscript received July 14, 1981; revised March 30, 1982. This work was supported in part by the National Science Foundation under Grant ECS-7916292 and by the U.S. Army Research Office, Durham, NC under Grant DAAG29-79-C-0024.

S. Toueg was with the Department of Electrical Engineering and Computer Science, Princeton University, Princeton, NJ 08544. He is now with the Department of Computer Science, Cornell University, Ithaca, NY 14853.

K. Steiglitz is with the Department of Electrical Engineering and Computer Science, Princeton University, Princeton, NJ 08544.

I. BASIC DEFINITIONS

The nodes of a packet radio network are *transmitter-receiver radios*, also called *repeaters*. A node v is *radio-connected* to a node u if radio signals transmitted by u can be received by radio v, we also say that v is *within receiving range from u*. Messages called *packets* are passed between network nodes. Associated with each packet is an *acyclic route* which is a sequence $v_1, v_2, \cdots, v_q$ of nodes such that v_{i-1} is radio-connected to v_i for $i = 1, 2, \cdots, q-1$. Node v_1 is the *source radio*, and v_q is the *destination radio* for the packet. A packet wends its way from node to node along its route until it reaches its destination and is delivered there. This point-to-point packet transmission takes place as follows (a more detailed description can be found in [2]). Each radio has a node identifier, or *address* (called *selector* in [2]), which is, for the moment, assumed unique. A source node generates a packet, with associated addressing information in the packet header, and transmits it to the nearby radio which is identified in the header as the first node along the packet's route. All the radios that are within receiving range from the source will "hear" the packet and process its header to determine if they should relay the packet (to the next node along the packet's route), deliver it, or discard it: only one of these radios, the one identified in the packet header, will actually relay it (or deliver it, if it is the packet destination); all the other radios discard the packet. A packet header contains the identifier of the next node along its route with some additional routing information which depends on the implementation. In one implementation, every packet carries in its header the entire packet's route given as the sequence of identifiers of all the radios along the route. In another implementation, some routing information is stored in *renewal tables* in intermediate nodes, denoted *renewal points* in [2], along a packet's route. The renewal table of a node maintains an entry with the identifiers of the next few successive nodes for each route passing through the node. The initialization and updating of the renewal tables by *route setup packets* is described in [2] and is of no concern to us here. With this scheme, a packet sent along a previously set up route contains the identifiers of only the next few nodes along this route (up to the next renewal point). When a packet arrives at a renewal point, the routing identifiers in the packet header are rewritten according to the node renewal table entry for the packet's route.

It is clear that, with both implementations, a reduction in the length of identifier codes results in a reduction of packet header lengths. This reduction is more significant with the first implementation described, but in the second implementation the space taken by the renewal tables is reduced as well. The minimization of identifier code lengths is now investigated.

II. FIXED-LENGTH OPTIMAL ADDRESSING CODES

Formally, a packet radio network is a directed graph $G = (V, E)$; the vertices V represent transmitter–receiver radios, and the edges E represent radio links, i.e., $E = \{(u, v) \mid v$ is radio-connected to $u\}$. If we give a unique identifier (*absolute address*) to each node in a network $G = (V, E)$, and to this absolute address we correspond a fixed-length binary code, then the length of each such code is $\lceil \log_2 |V| \rceil$ bits. But this simple address coding scheme is not optimal. In fact, we can reduce the number of bits required for addressing by assigning to each node a *local address* (in addition to its absolute

0090-6778/82/1100-2455$00.75 © 1982 IEEE

address) in the following way. Let u be a node of a network $G = (V, E)$, and $S_u = \{v \mid (u, v) \in E\}$ be the set of *successors* of u in G, i.e., S_u is the set of radios which are within receiving range from u. When u transmits a packet, it is received by all the radios in S_u. Then, for a packet to be unambiguously sent from u to a particular radio v in S_u it is necessary and sufficient to assign a different local address to each node in S_u (and to include the local address of v in the packet header, of course). Therefore, unambiguous address coding requires any two nodes v and w to have a different local address if there is a node u such that both (u, v) and (u, w) are edges in G. In the following discussion, we consider only unambiguous addressing codes. To simplify the notation, we shall assume that communication links between radios are symmetric, i.e., if (u, v) is an edge of G then (v, u) is also an edge of G (or, equivalently, G is an undirected graph). Then two nodes v and w must have different local addresses if they are both adjacent to a common node u. So if the node-adjacency matrix of a network G is A, then the nodes v and w of G must have different local addresses if (v, w) is an edge in the graph G^2 whose adjacency matrix is $A^2 = A \times A$, where Boolean addition and multiplication are used in computing the matrix product (note that G^2 will generally include (v, v) loops, and in our remark v and w are assumed to be distinct nodes of G^2). Thus, if we use the term "color" instead of "local address," the minimum number of colors required to color the nodes of G^2 such that no adjacent (distinct) nodes have the same color corresponds to the optimal fixed-length unambiguous addressing code (optimal in the sense that the number of bits required for this code is minimum with respect to all the other fixed-length unambiguous addressing codes). If the minimum number of colors required to color the network is k, then local address binary codes are $\lceil \log_2 k \rceil$ bits long. It is a well-known result that the minimal coloring of a graph G is an NP-complete problem [1], and it turns out that the minimal coloring of G^2 is also NP-complete.

Theorem: The following problem, "given a graph G and a constant k, is G^2 colorable with k colors?", is NP-complete.

Proof: A k-coloring of G^2 can be guessed and checked in polynomial time, so the problem is in NP. We show that the 3-satisfiability problem [1] is polynomially transformable to the G^2 colorability problem. Given an expression F in 3-CNF with n variables and t factors, we construct, in polynomial time, an undirected graph $G = (V, E)$ with $4n + t(n + 1)$ nodes, such that G^2 can be colored with $n + 1$ colors if and only if F is satisfiable. Let $x_1, x_2, \cdots, x_m$ and $F_1, F_2, \cdots, F_t$ be variables and factors of F, respectively. Without loss of generality we assume that $n \geqslant \max(4, t)$; we can always add dummy variables (i.e., variables that do not appear in F) to satisfy the inequality. The nodes of G are

1) $x_i, \bar{x}_i, v_i$, and r_i, for $1 \leqslant i \leqslant n$,
2) F_j, for $1 \leqslant j \leqslant t$, and
3) s_{ij}, for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant t$.

The edges of G are

1) all (v_i, r_j) such that $i \neq j$,
2) (r_i, x_i) and $(r_i, \bar{x}_i)$, for $1 \leqslant i \leqslant n$,
3) (x_i, s_{ij}) if x_i is not a term of factor F_j, and $(\bar{x}_i, s_{ij})$, if $\bar{x}_i$ is not a term of F_j,
4) (s_{ij}, F_j), for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant t$.

It is easy to check that G^2 consists of two disconnected subgraphs G_1 and G_2. G_1 is the graph described in [1, pp. 392–393]. It is shown there that G_1 can be colored with $n + 1$ colors if and only if F is satisfiable. The nodes of G_2 are

1) r_i, for $1 \leqslant i \leqslant n$, and
2) s_{ij}, for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant t$.

The edges of G_2 are

1) all (r_i, r_j) such that $i \neq j$,
2) (r_i, s_{ij}) if x_i or $\bar{x}_i$ is not a term of factor F_j,
3) all (s_{ij}, s_{kj}) such that $i \neq k$, and
4) (s_{ij}, s_{ik}) if x_i or $\bar{x}_i$ is not a term of both F_j and $F_k (j \neq k)$.

We now show that G_2 can always be colored with $n + 1$ colors. We denote the colors by $0, 1, 2, \cdots, n$. We color the nodes r_i with the color $i \bmod (n + 1)$, for $1 \leqslant i \leqslant n$, and the nodes s_{ij} with the color $i + j \bmod (n + 1)$, for $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant t$. It is easy to verify that any two adjacent nodes of G_2 have a different color. It is now clear that G_2 can be colored with $n + 1$ colors if and only if F is satisfiable. □

It is therefore unlikely that a polynomial-time algorithm resulting in optimal fixed-length unambiguous addressing codes can be found. Nevertheless, finding good or near-optimal addressing codes is not a hopeless task for the following reasons. Let G be a network and d be the maximal node degree of G. From our previous discussion about unambiguous address codings it is clear that we need at least d distinct local addresses (i.e., colors) for the network G. But in G^2 the maximal node degree is $d^2 - d$; therefore $d^2 - d + 1$ colors are always sufficient to color G^2 (the algorithm to color a degree k graph with $k + 1$ colors is trivial). Then d is a lower bound and $d^2 - d + 1$ is an upper bound on the number of distinct local addresses needed for unambiguous addressing in G. The corresponding bounds for the number of bits of the address binary code are $\lceil \log_2 d \rceil$ and $\lceil \log_2 (d^2 - d + 1) \rceil$. Therefore, the trivial algorithm for coloring G^2 cannot result in an address code which is longer than twice the length of the optimal code. We may note that the bounds given are tight bounds; there are degree d graphs which can be colored with only d colors, and degree d graphs which cannot be colored with fewer than $d^2 - d + 1$ colors.[1] But there is reason to believe that heuristic algorithms for coloring achieve better results on the average. In fact, a very simple linear sequential algorithm for graph coloring has a high performance ratio (i.e., number of colors required by the algorithm over the optimal number of colors) of $2 + \epsilon$ a.e. when applied to constant-density random graphs [3]. So, for random graphs, this heuristic algorithm results a.e. in codes which are only $1 + \epsilon$ bits longer than the optimal ones. Then a practical heuristic algorithm for unambiguous address code minimization could consist of the following two steps.

1) Given a network G, with a node-adjacency matrix A, compute the adjacency matrix $A^2 = A \times A$ (Boolean product) of the graph G^2. Unambiguous address coding requires distinct codes to be assigned to adjacent nodes of G^2.

2) Apply the graph coloring sequential algorithm on G^2. Each color corresponds to a local address. If the algorithm results in k different colors then $\lceil \log_2 k \rceil$ bits are required for the fixed-length binary code of each local address.

II. VARIABLE-LENGTH OPTIMAL ADDRESSING CODES

If we do not insist on fixed length codes for local addresses we may consider variable length codes having the *prefix*

[1] A family of such graphs, for $d = p + 1$ and p any prime number, was found by M. Yannakakis.

property (the prefix property guarantees that any contiguous sequence of address codes, denoting the routing information of a packet, can be unambiguously decoded by the radios along the packet's route). If we assume that all the nodes in the graph have the same probability of appearing as addressing information for packets in a network, then the search for the unambiguous addressing codes that achieve the minimum *average* code length (over all the nodes) leads to the following problem.

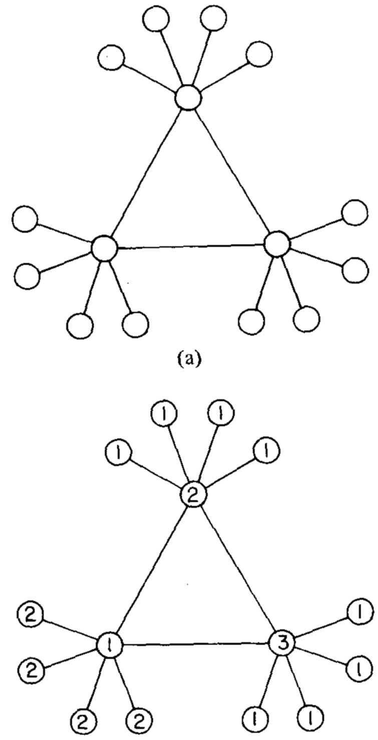
Optimal Graph Coding Problem: Given a graph $G = (V, E)$, find the optimal prefix-property code for the nodes of G (i.e., the prefix-property code with the minimum total number of bits for coding all of G's nodes) such that adjacent nodes have distinct codes.

We first note that to find the optimal unambiguous prefix-property addressing code of a network G, we must solve the optimal graph coding problem for the graph G^2. We may also note that the well-known Huffman coding recursive algorithm solves the graph coding problem for a restricted class of graphs. In fact, finding the optimal prefix-property coding for a source text with k distinct characters appearing with the respective frequencies of $n_1, n_2, \cdots, n_k$ characters corresponds to solving the optimal graph coding problem for a graph with k independent sets with $n_1, n_2, \cdots, n_k$ nodes, respectively, each node of each independent set being connected to all the nodes of the other independent sets.

The complexity of the optimal graph coding problem is not yet known. We might only hope that a polynomial-time algorithm (some generalization of the recursive Huffman coding algorithm?) or else a proof of intractability will soon be found. It is interesting to note that the minimum number of unambiguous local addresses in a network (which results in the optimal fixed-length coding) does not necessarily yield an optimal variable-length prefix-property coding for the network. For example, consider the graph G illustrated in Fig. 1(a). It is clear that three colors are necessary and sufficient to color this graph. One such minimal coloring of G is shown in Fig. 1(b) (the colors are denoted by the numbers 1, 2, and 3). We have nine "one's," five "two's," and one "three." The optimal prefix-property codes for these colors are obtained by applying the Huffman coding recursive algorithm, and the results are illustrated in Fig. 1(c). Then to code the entire graph we need $9 \times 1 + 5 \times 2 + 1 \times 2 = 21$ bits. It turns out that the optimal prefix-coding of this graph is obtained when four colors are used instead of the minimal three. This coloring and the corresponding Huffman codes are given in Fig. 2(a) and (b). In this case we need only $12 \times 1 + 1 \times 2 + 1 \times 3 + 1 \times 3 = 20$ bits to code the entire graph, and we can prove that this is optimal for this graph.

Finally, a practical heuristic algorithm for the minimization of variable-length addressing codes of a network G can be obtained by appending the following Step 3 to the end of the heuristic algorithm described in Section I.

3) Let $n_1, n_2, \cdots, n_k$ be the respective frequencies in G of the k colors in the coloring of G^2. Apply the Huffman coding recursive algorithm to these k colors; the resulting codes are local address codes.

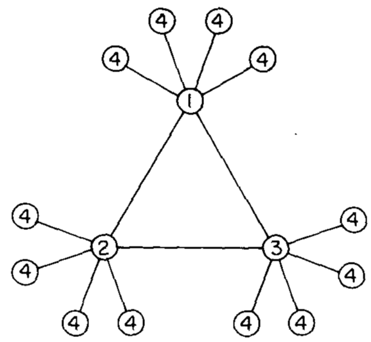


(b)

Color	"one"	"two"	"three"
Frequency	9	5	1
Code	0	10	11
Code length	1	2	2

(c)

Fig. 1. (a) A graph G. (b) A minimal coloring of G. (c) Corresponding address codes for G.



(a)

Color	"one"	"two"	"three"	"four"
Frequency	1	1	1	12
Code	10	110	111	0
Code length	2	3	3	1

(b)

Fig. 2. (a) A nonminimal coloring of G. (b) Corresponding optimal address codes.

ACKNOWLEDGMENT

We would like to thank R. E. Kahn who proposed the packet radio addressing problem to us in a private communication.

REFERENCES

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*. Reading, MA: Addison-Wesley, 1974.

[2] R. E. Kahn *et al.*, "Advances in packet radio technology," *Proc. IEEE*, vol. 66, pp. 1468–1496, Nov. 1978.

[3] J. F. Traub, Ed., *Algorithms and Complexity*. New York: Academic, 1976, pp. 11–12.