# A simple $O(mn)$ algorithm for recognizing Hamming graphs

W. Imrich
Institut für Mathematik und Angewandte Geometrie
Montanuniversität Leoben
A–8700 Leoben
Austria

S. Klavžar *
University of Maribor
PF, Koroška cesta 160
62000 Maribor
Slovenia

**Abstract**

We show that any isometric irredundant embedding of a graph into a product of complete graphs is the canonical isometric embedding. This result is used to design a simple $O(mn)$ algorithm for recognizing Hamming graphs.

## 1 Introduction

All graphs considered in this paper are finite undirected graphs without loops and multiple edges. Throughout the paper, for a given graph $G$, let $n$ and $m$ stand for the number of its vertices and edges, respectively.

Graphs that can be embedded isometrically in a Cartesian product of complete graphs are called Hamming graphs. Interest in Hamming graphs has been decisively stimulated by the work of Graham and Pollak [10, 11]

*This work was supported in part by the Ministry of Science and Technology of Slovenia under the grant P1-0206-101-92.

in communication theory and Firsov [8] in linguistics. In biology, Hamming graphs appear as "quasi-species" [6].

Several algorithms for recognizing Hamming graphs have been proposed. The first algorithm is due to Winkler [16]. Its running time is bounded by $O(n^5)$. Aurenhammer, Formann, Idury, Schäffer and Wagner [2] improved Winkler's algorithm to run in $O(D(m,n) + n^2)$ time, where $D(m,n)$ is the time needed to compute the distance matrix of the graph. Wilkeit [14] presented another algorithm running in $O(n^3)$ time.

In 1985 Graham and Winkler [12] published a fundamental paper on isometric embeddings into Cartesian product graphs. Using their results together with a result of Aurenhammer and Hagauer [4] one can give an $O(mn)$ algorithm to recognize Hamming graphs. Aurenhammer and Hagauer [3] also proposed another algorithm of the same complexity to recognize binary Hamming graphs.

In this paper we present an $O(mn)$ algorithm to recognize Hamming graphs. Our algorithm is also based on the theory of isometric embeddings into Cartesian product graphs. However, our approach needs a minimum of theory and the algorithm itself is simple and straightforward. We have taken care that everything up to and including Section 4 is self-contained. In particular, this includes all results about binary Hamming graphs.

In the next section we state the necessary definitions and recall a connection between Hamming graphs and the Cartesian product of graphs. In Section 3 we introduce the relation $\Theta$ and reprove a characterization of binary Hamming graphs. Section 4 follows with an algorithm for the recognition of binary Hamming graphs. In the last section we prove that any isometric irredundant embedding of a graph into a product of complete graphs is the canonical isometric embedding. With the aid of this result we extend the algorithm for binary Hamming graphs to all Hamming graphs.

## 2 Hamming graphs and the Cartesian product

Let $\Sigma$ be a finite alphabet and let $w_1$ and $w_2$ be words of equal length over $\Sigma$. Then the *Hamming distance* between $w_1$ and $w_2$, $H(w_1, w_2)$, is the number of positions $k$ in $w_1$ and $w_2$ such that the $k$-th symbol in $w_1$ differs from the $k$-th symbol in $w_2$. A graph $G$ is called a *Hamming graph*, if each vertex $v \in V(G)$ can be labelled by a word of fixed length, $a(v)$, such that $H(a(u), a(v)) = d_G(u, v)$ for all $u, v \in V(G)$. Here $d_G(u, v)$ denotes the usual shortest path distance in $G$ between $u$ and $v$. In particular, if $\Sigma = \{0, 1\}$, we call $G$ a *binary Hamming graph*.

The *Cartesian product* $G \,\Box\, H$ of graphs $G$ and $H$ is the graph with vertex set $V(G) \times V(H)$ and $(a, x)(b, y) \in E(G \,\Box\, H)$ whenever $ab \in E(G)$ and $x = y$, or $a = b$ and $xy \in E(H)$. The Cartesian product is commutative, associative and $K_1$ is a unit. Also, $G \Box H$ is connected if and only if both $G$ and $H$ are connected. For $G_1 \Box G_2 \Box \cdots \Box G_k$ we shall also write $\prod_{i=1}^{k} G_i$.

For $G = \prod_{i=1}^{k} G_i$ let $p_i : V(G) \to V(G_i)$, $i \in \{1, 2, \ldots, k\}$, be the natural projection of $G$ onto the $i$-th factor $G_i$, i.e. for $v = (v_1, v_2, \ldots, v_k) \in V(G)$ we set $p_i(v) = v_i \in V(G_i)$. For $X \subseteq V(G)$ let $p_i(X) = \{p_i(x) \mid x \in X\}$. In particular, for $e = uv \in E(G)$ let $p_i(e) = \{p_i(u), p_i(v)\}$. We also introduce a *product coloring* $c : E(G) \to \{1, 2, \ldots, k\}$ *(with respect to the product representation)* as follows. For $uv \in E(G)$ we set $c(uv) = i$ if and only if $u$ and $v$ differ in coordinate $i$. Clearly $c$ is a mapping from $E(G)$ into $\{1, 2, \ldots, k\}$. It is not an edge coloring in the usual sense, because incident edges may have the same color.

A subgraph $H$ of a graph $G$ is an *isometric* subgraph, if $d_H(u, v) = d_G(u, v)$ for all $u, v \in V(H)$. In addition, if $\alpha : V(H) \to V(G)$ maps edges into edges and $\alpha(H)$ is an isometric subgraph of $G$, we call $\alpha$ an *isometric embedding* of $H$ into $G$.

The following observation is a starting point in studying isometric subgraphs of Cartesian products of graphs.

**Lemma 2.1** *For $G = \prod_{i=1}^{k} G_i$, $k \geq 1$, let $u = (u_1, u_2, \ldots, u_k)$ and $v = (v_1, v_2, \ldots, v_k)$ be arbitrary vertices of $G$. Then $d_G(u, v) = \sum_{i=1}^{k} d_{G_i}(u_i, v_i)$.*

With Lemma 2.1 it is an easy exercise to characterize Hamming graphs:

**Theorem 2.2** *A graph $G$ is a Hamming graph if and only if $G$ is an isometric subgraph of a Cartesian product of complete graphs.* □

Theorem 2.2 in particular implies that binary Hamming graphs are isometric subgraphs of hypercubes (the $k$-dimensional hypercube $Q_k$, $k \geq 1$, is the Cartesian product of $k$ copies of $K_2$).

## 3 The relation $\Theta$

In this section we present a characterization of binary Hamming graphs which is essential for a fast recognition algorithm.

Let $G$ be a connected graph. Define a relation $\Theta$ on $E(G)$ as follows. If $e = uu' \in E(G)$ and $e' = vv' \in E(G)$, then $e \Theta e'$ if and only if

$$d(u, v) + d(u', v') \neq d(u, v') + d(u', v).$$

This relation, which was first introduced in an alternative form in [5], plays a central role in our investigation. The relation $\Theta$ is well-defined, reflexive and symmetric, yet it need not be transitive. We denote its transitive closure by $\Theta^*$.

**Lemma 3.1** *Let $P$ be a shortest path in a graph $G$. Then no two different edges of $P$ are in relation $\Theta$.*

**Proof.** Let $u_0u_1\cdots u_m$ be a shortest path, and let $e = u_iu_{i+1}$, $e' = u_ju_{j+1}$, $i < j$. Then

$$d(u_i,u_j) + d(u_{i+1},u_{j+1}) = (d(u_{i+1},u_j)+1) + (d(u_i,v_{j+1}-1),$$

hence $e$ is not in the relation $\Theta$ with $e'$. □

It follows from Lemma 3.1 that for a tree on $n$ vertices $\Theta^*$ consists of $n-1$ equivalence classes, each containing a single edge. Lemma 3.1 also implies that two adjacent edges are in relation $\Theta$ if and only if they lie in a common triangle.

**Lemma 3.2** *Suppose $P$ is a path connecting the endpoints of an edge $e$. Then $P$ contains an edge $f$ with $e\Theta f$.*

**Proof.** Let $u_0u_1\cdots u_mu_0$ be a closed path with $e = u_mu_0$ and $e_i = u_{i-1}u_i$ for $i = 1, 2, \ldots, m$. Set

$$\mu(e,e_i) = d(u_m,u_{i-1}) + d(u_0,u_i) - d(u_m,u_i) - d(u_0,u_{i-1})$$

and consider

$$s = \sum_{i=1}^{m} \mu(e,e_i)\ .$$

Clearly $s = 2$, which means that at least one of the summands $\mu(e,e_i) \neq 0$, i.e. $e\Theta e_i$. □

For an edge $uv$ of a graph $G$ let

$$V_{uv} \ = \ \{\, w \mid w \in V(G),\, d_G(w,u) < d_G(w,v)\,\}.$$

**Lemma 3.3** *Let $e = uv$ be an edge of a connected bipartite graph $G$ and let*

$$E_e \ = \ \{\, f \mid f \in E(G),\, e\Theta f\,\}.$$

*Then $G\backslash E_e$ has exactly two components. Furthermore, they are induced by the vertex sets $V_{uv}$ and $V_{vu}$.*

**Proof.** $G \backslash E_e$ is disconnected by Lemma 3.2.

Let $w \in V_{uv}$ and let $P$ be a shortest $u - w$ path. Then $vP$ is a shortest $v - w$ path. Hence, using Lemma 3.2 again, no edge of $P$ is in relation $\Theta$ with $e$. It follows that $P$ connects $u$ and $w$ in $G \backslash E_e$.

Since $G$ is bipartite, $d(w,u) \neq d(w,v)$. Therefore $V_{uv}$ and $V_{vu}$ form a partition of $V(G)$. In addition, no edge $ww' \in E(G)$, where $w, w' \in V_{uv}$, is in relation $\Theta$ with $e$, and the proof is complete. □

For Cartesian product graphs Lemma 2.1 immediately implies:

**Lemma 3.4** *Let $G = \prod_{i=1}^{k} G_i$ and let $e, e' \in E(G)$.*
*(i) If $c(e) = c(e') = i$ and $p_i(e) = p_i(e')$ then $e\Theta e'$.*
*(ii) If $c(e) \neq c(e')$ then $e\Theta e'$ does not hold.*

Consider the product $G \,\Box\, H$ of two graphs. Then Lemma 3.4 (i) claims that if $ab \in E(G)$, then $(a,x)(b,x)$ is in relation $\Theta$ with $(a,y)(b,y)$, for any $x$, $y \in V(H)$. Lemma 3.4 (ii) on the other hand says that if $ab \in E(G)$ and $xy \in E(H)$, then $(a,z)(b,z)$ is not in relation $\Theta$ with $(c,x)(c,y)$, for any $c \in V(G)$ and $z \in V(H)$, i.e. edges with different colors with respect to the product coloring are not in relation $\Theta$. In other words, every color class (with respect to the product coloring) is the union of one or more equivalence classes with respect to $\Theta^*$.

**Theorem 3.5** [16, Winkler] *A graph $G$ is a binary Hamming graph if and only if $G$ is bipartite and $\Theta^* = \Theta$.*

**Proof.** Assume $G$ is an isometric subgraph of a hypercube. Then $G$ is clearly bipartite.

Define a relation $R$ on $E(G)$ as follows. For $e, e' \in E(G)$ let $eRe'$ if and only if $c(e) = c(e')$. Let $c(e) = c(e') = i$. Then $p_i(e) = p_i(e') = \{0,1\}$ and hence, by Lemma 3.4 (i), $e\Theta e'$. Furthermore, by Lemma 3.4, $e$ and $e'$ are not related by $\Theta$ if $c(e) \neq c(e')$. It follows $R = \Theta$. As $R$ is transitive, we conclude $\Theta^* = \Theta$.

Conversely, let $G$ be bipartite and let $\Theta^* = \Theta$. Let $e_1 = x_1y_1$, $e_2 = x_2y_2$, $\ldots$, $e_k = x_ky_k$, $k \geq 1$ be representatives of each equivalence class of $\Theta^*$. Define an embedding $\alpha : V(G) \to Q_k = \{0,1\}^k$ in the following way. Let $v \in V(G)$. For $i = 1, 2, \ldots, k$ let the $i$-th coordinate of $\alpha(v)$ be 0 if $v \in V_{x_iy_i}$ and 1 if $v \in V_{y_ix_i}$. We claim that $\alpha$ is an isometric embedding.

Let $uv \in E(G)$ and assume $uv$ belongs to the equivalence class of $e_i$. By Lemma 3.3, $\alpha(u)$ and $\alpha(v)$ differ in the $i$-th coordinate. Furthermore, if $j \neq i$ then $d(u, x_j) + d(v, y_j) = d(u, y_j) + d(v, x_j)$. Thus $\alpha(u)$ and $\alpha(v)$ have same $j$-th coordinate. It follows that $\alpha$ maps edges to edges.

As $G$ is bipartite and $\Theta = \Theta^*$, no two adjacent edges are in the same equivalence class. Furthermore, if $P$ is a shortest path between $u$ and $v$ then, by Lemma 3.1, $\alpha(u)$ and $\alpha(v)$ differ in just as many coordinates as the number of edges of $P$, i.e., the distance $d_G(u, v)$. □

# 4 Recognizing binary Hamming graphs

In a direct implementation of Theorem 3.5 we must check for all pairs of edges whether they are in relation $\Theta$. For a given graph $G$ this leads to an algorithm with time complexity $O(m^2)$. In order to improve this complexity, we introduce relation $\Theta_1$, which is due to Feder [7].

Let $T$ be a spanning tree of a graph $G$. Then edges $e, e' \in E(G)$ are in the relation $\Theta_1$ if and only if they are in the relation $\Theta$ and if at least one of the edges $e$ and $e'$ belongs to $T$. Let $E_1$, $E_2$, $\ldots$, $E_k$ be the equivalence classes of the relation $\Theta_1^*$. For $i = 1, 2, \ldots, k$ let $G_i$ denote the graph $(V(G), E(G) \backslash E_i)$ and let $C_{i,1}$, $C_{i,2}$, $\ldots$, $C_{i,m_i}$ denote the connected components of $G_i$. Form the graphs $G_i^*$, $i = 1, 2, \ldots, k$, by letting $V(G_i^*) = \{C_{i,1}, C_{i,2}, \ldots, C_{i,m_i}\}$ and taking $C_{i,j}C_{i,j'}$ to be an edge of $G_i^*$ if some edge in $E_i$ joins a vertex in $C_{i,j}$ to a vertex in $C_{i,j'}$.

We now define the natural contraction $\alpha_i : V(G) \to V(G_i^*)$ by setting $\alpha_i(v) = C_{i,j}$ if $v \in C_{i,j}$. Finally, we obtain a mapping

$$\alpha : V(G) \to \prod_{i=1}^{k} G_i^* \qquad (*)$$

by setting

$$\alpha(v) = (\alpha_1(v), \alpha_2(v), \ldots, \alpha_k(v)). \qquad (*)$$

It is important for our algorithm that $k \leq n-1$. Indeed, let $e = uv \in E_i$. By Lemma 3.2 any path in $G$ from $u$ to $v$ must traverse at least one edge from $E_i$. This is in particular true for the path between $u$ and $v$ in $T$, hence there is at least one edge of $T$ belonging to $E_i$.

This also means that $\Theta = \Theta_1^*$ in a binary Hamming graph. For, suppose $e\Theta f$. Then there is an $e' \in T$ with $e'\Theta e$. Since $\Theta = \Theta^*$ we also have $e'\Theta f$ and $e\Theta_1^* f$. Hence $\Theta \subseteq \Theta_1^*$. Since $\Theta_1 \subseteq \Theta$ we also have $\Theta_1^* \subseteq \Theta^* = \Theta$, which proves the assertion.

**Algorithm BHG**

*Input*: a connected graph $G$.

*Output*: TRUE, and a labeling, if $G$ is a binary Hamming graph. FALSE, otherwise.

1. If $G$ is not bipartite then return FALSE and stop.
2. Compute $\Theta_1^*$.
3. Compute $G_i$, $i = 1, 2, \ldots, k$ and $\alpha(v)$, $v \in V(G)$.
4. For $i = 1, 2, \ldots, k$, compute $m_i$, i.e., the number of components in $G_i$. If for some $i$, $m_i > 2$, then return FALSE and stop.
5. Return TRUE and the labeling of $G$ obtained in step 3.

**Theorem 4.1** *Algorithm BHG correctly recognizes binary Hamming graphs and can be implemented to run in $O(mn)$ time using $O(m)$ space.*

**Proof.** Suppose that $G$ is a binary Hamming graph. Then $\Theta = \Theta_1^*$ and every $G_i$ has exactly two components by Lemma 3.3. It thus remains to show that $\alpha$ is an isomorphism, if every $G_i$ has exactly two components.

Let $P$ be a shortest path between $u$ and $v$ in $G$. By Lemma 3.1 no two edges of $P$ are in the relation $\Theta$ and thus all edges of $P$ belong to different $\Theta_1^*$ classes. This means that the labels of $u$ and $v$ differ in just as many coordinates as the number of edges of $P$, i.e. the distance $d_G(u, v)$. This proves the correctness of the algorithm.

Concerning the running time we first note that it is trivial to check bipartiteness in $O(m)$ time and space.

Let $T$ be a spanning tree of $G$ and let $uv$ be an edge of $T$. Then we can calculate the distances from $u$ and $v$ to all other vertices in $O(m)$ time and $O(m)$ space. Hence we get all the edges in $G$ related to $e$ under $\Theta_1$ in the same time, and in time $O(mn)$ over all edges in $T$. To get the equivalence classes of $\Theta_1^*$ we merge equivalence classes of two edges whenever we determine that they are in the relation $\Theta_1$. During the execution of the procedure there will be at most $m-1$ such union operations and $m(n-1)$ find operations. It is well-known that these can be done in $O(nm)$ time using $O(m)$ space (see, for example, [1]).

When $E_i$ is known, it is easy to construct the graph $G_i$ in $O(m)$ time. As $k \leq n-1$, all the $G_i$ can be obtained in $O(mn)$ time. Since every edge in $G_i$ corresponds to an edge of $G$, we get all the $G_i$ using $O(m)$ space. In addition, we don't need to store the complete information on $\alpha(v)$, it is enough to store the value $\alpha_i(v)$ if it is different from $\alpha_i(u)$ for some $uv \in E(G)$. $\Box$

Graham [9] proved that there exists a fixed small $c$ such that $m \leq c\, n \log n$ for any subgraph of a hypercube. Thus:

**Corollary 4.2** *Algorithm BHG can be implemented to run in $O(n^2 \log n)$ time using $O(n \log n)$ space.*

## 5 Recognizing Hamming graphs

We are going to modify Algorithm BHG to recognize general Hamming graphs.

Let $\alpha$ be the mapping $(*)$ defined in Section 4. Call an isometric embedding $\beta : G \to \prod_{i=1}^{m} H_i^*$ *irredundant* if $|H_i| \geq 2$, $i = 1, 2, \ldots, m$, and for all $h \in V(H_i)$, $h$ occurs as a coordinate value of the image of some $g \in V(G)$. This is the principal result in the theory of isometric embeddings into Cartesian product graphs:

**Theorem 5.1** [12, Graham and Winkler] *The mapping $\alpha$ is an isometric embedding of $G$ into $\prod_{i=1}^{k} G_i^*$, the so-called* canonical embedding. *Furthermore, the embedding $\alpha$ is irredundant and has the largest possible number of factors among all irredundant isometric embeddings of $G$.*

Graham and Winkler [12] proved Theorem 5.1 for the relation $\Theta$, while Feder [7] showed $\Theta_1^* = \Theta^*$.

For the isometric embeddings into products of complete graphs we have:

**Theorem 5.2** [16, Winkler] *Any two isometric embeddings of a graph into products of complete graphs are equivalent.*

Equivalence in Theorem 5.2 essentially means that equivalent isometric embeddings can be obtained from one another by discarding unused factors, permuting factors, and permuting vertices within a factor.

The following theorem is crucial for our algorithm.

**Theorem 5.3** *Let $\beta : G \to \prod_{i=1}^{m} H_i$ be an isometric irredundant embedding of a graph $G$ into a product of complete graphs $H_i$. Then this embedding is the canonical isometric embedding.*

**Proof.** By Theorem 5.2 we have to show that any two edges $e$, $e'$ in $\beta(G)$ are in the relation $\Theta^*$ if they have the same color with respect to the product coloring of $\prod_{i=1}^m H_i$.

Consider now $i$-layers $U$ and $V$ with respect to $H_i$. We claim that $p_i(\beta(G) \cap U) \subseteq p_i(\beta(G) \cap V)$ or vice versa. If this is not the case, there are vertices $u, u' \in U$ and $v, v' \in V$ such that

$$u \in \beta(G) \cap U, \;\; u' \notin \beta(G) \cap U$$

and

$$v \in \beta(G) \cap V, \;\; v' \notin \beta(G) \cap V,$$

where $p_i(u) = p_i(v')$, $p_i(u') = p_i(v)$ (see Fig. 1).

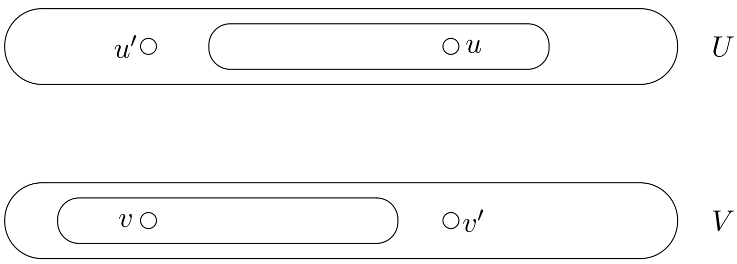


Figure 1: Layers $U$ and $V$

Suppose the distance between $U$ and $V$ in $\prod_{i=1}^m H_i$ is $k$. Then $k + 1 = d_H(u, v) = d_{\beta(G)}(u, v)$, which is only possible if $u' \in \beta(G)$ or $v' \in \beta(G)$. This proves the claim.

Thus, let the $i$-colored edges $e, e'$ in $\beta(G)$ be in the $i$-layers $U$ and $V$, respectively. If $p_i(\beta(G) \cap U) \subseteq p_i(\beta(G) \cap V)$, then by Lemma 3.4 (i) there is an edge $e''$ in $\beta(G) \cap V$ with $e \Theta e''$. Since $\beta(G) \cap V$ is complete we have $e'' \Theta e'$, and hence $e \Theta^* e'$. □

**Algorithm HG**

*Input*: a connected graph $G$.

*Output*: TRUE, and a labeling, if $G$ is a Hamming graph. FALSE, otherwise.

1. Compute $\Theta_1^*$.
2. Compute $G_i$, $i = 1, 2, \ldots, k$ and $\alpha(v)$, $v \in V(G)$.
3. If for some $i = 1, 2, \ldots, k$, $G_i^*$ is not a complete graph, then return FALSE and stop.
4. Return TRUE and the labeling of $G$ obtained in step 2.

**Theorem 5.4** *Algorithm HG correctly recognizes Hamming graphs and can be implemented to run in $O(mn)$ time using $O(m)$ space.*

**Proof.** Correctness of the algorithm follows from Theorem 5.3.

The complexity can be argued as in the proof of Theorem 4.1 with the exception of Step 3. Every edge of a factor graph $G_i^*$ correspond to an edge of $G$ and furthermore, this correspondence is injective. It follows $\sum_{i=1}^{k} |E(G_i^*)| \leq m$. Hence to implement Step 3 in the desired time and space it is sufficient to count the number of edges in the $G_i^*$'s. □

## References


[1] A. Aho, J. Hopcroft and J. Ullman, The Design and Analysis of Computer Algorithms (Addison-Wesley, Reading, 1974).

[2] F. Aurenhammer, M. Formann, R. Idury, A. Schäffer and F. Wagner, Faster isometric embeddings in products of complete graphs, Discrete Appl. Math., to appear.

[3] F. Aurenhammer and J. Hagauer, Recognizing binary Hamming graphs in $O(n^2 \log n)$ time, in: Proc. 16th Int. Workshop on Graph Theoretical Concepts in Computer Science, Lecture Notes in Comput. Sci., Vol. 484 (Springer, New York, 1991) 90–98.

[4] F. Aurenhammer and J. Hagauer, Computing equivalence classes among the edges of a graph with applications, Discrete Math. 109 (1992) 3–12.

[5] D. Djoković, Distance preserving subgraphs of hypercubes, J. Combin. Theory Ser. B 14 (1973) 263–267.

[6] M. Eigen and R. Winkler-Oswattisch, Transfer-DNA: the early adaptor, Naturwissenschaften 68 (1981) 217–228.

[7] T. Feder, Product graph representations, J. Graph Theory 16 (1992) 467–488.

[8] V. Firsov, Isometric embedding of a graph in a Boolean cube (Russian), Kibernetica 1 (1965) 95–96.

[9] R. Graham, On primitive graphs and optimal vertex assignments, Ann. New York Acad. Sci. 175 (1970) 170–186.

[10] R. Graham and H. Pollak, On the addressing problem for loop switching, Bell System Tech. J. 50 (1971) 2495–2519.

[11] R. Graham and H. Pollak, On embedding graphs in squashed cubes, in: Graph Theory and Applications, Lecture Notes in Math., Vol. 303 (Springer, New York, 1972) 99–110.

[12] R. Graham and P. Winkler, On isometric embeddings of graphs, Trans. Amer. Math. Soc. 288 (1985) 527–536.

[13] W. Imrich, Embedding graphs into Cartesian products, Ann. New York Acad. Sci. 576 (1989) 266–274.

[14] E. Wilkeit, Isometric embeddings in Hamming graphs, J. Combin. Theory Ser. B 50 (1990) 179–197.

[15] P. Winkler, Proof of the squashed cube conjecture, Combinatorica 3 (1983) 135–139.

[16] P. Winkler, Isometric embeddings in products of complete graphs, Discrete Appl. Math. 7 (1984) 221–225.

[17] P. Winkler, The metric structure of a graph, in: C. Whitehead, ed., Theory and Applications, London Math. Soc. Lecture Note Ser., Vol. 123 (Cambridge University Press, Cambridge, 1987) 197–221.