University of Ljubljana
Institute of Mathematics, Physics and Mechanics
Department of Mathematics
Jadranska 19, 1000 Ljubljana, Slovenia

Preprint series, Vol. 40 (2002), 801

DISTANCE IN THE DIRECT PRODUCT OF GRAPHS

Iztok Peterin Janez Žerovnik

ISSN 1318-4865

February 22, 2002

Ljubljana, February 22, 2002

Distance in the direct product of graphs

Iztok Peterin
FEECS, University of Maribor
Smetanova 17, si-2000 Maribor, Slovenia
iztok.peterin@uni-mb.si

Janez Žerovnik *
FME, University of Maribor
Smetanova 17, si-2000 Maribor, Slovenia
janez.zerovnik@uni-lj.si

February 7, 2002

Abstract

A formula for the distance between any two vertices in the direct product of graphs is given in terms of even and odd distance in the factors. An $O(n^3)$ algorithm for computing even and odd distance in a graph is outlined.

1 Introduction

The cartesian, the direct and the strong product are among the most studied standard products of graphs [3]. While for the cartesian and for the strong product there are well-known formulas for the distance in the product in terms of distances in the factors, the corresponding result for the direct product has not been published yet, to the best of our knowledge. More precisely, it is well-known that the distance between arbitrary vertices (u_1, u_2) and (v_1, v_2) in the Cartesian product of graphs G_1 and G_2 is the sum of distances between u_1 and v_1 in G_1 and u_2 and v_2 in G_2. The distance between any two vertices (u_1, u_2) and (v_1, v_2) in the strong product of graphs G_1 and G_2 is the maximum of distances between u_1 and v_1 in G_1 and between u_2 and v_2 in G_2. This is also true for direct product in

*Also at IMFM, Jadranska 19, si-1111 Ljubljana, Slovenia. Partially supported by the Ministry of Science and Technology of Slovenia.

the case when G_1 and G_2 are bipartite. In general case this is not true. Indeed, the smallest graph on which we can find counter examples is $K_2 \times K_3$.

In this note we will define even and odd distance in the graph and using these it will be possible to write a simple formula for the distance in the direct product in terms of even and odd distances of the factors. A polynomial algorithm with time complexity $\mathcal{O}(n^3)$,$n = |V(G)|$, for computing the even and odd distance for any graph G will be given. It follows that the distances in the direct product of G_1 and G_2 can be computed in time $\mathcal{O}(n^3)$, where $n = \max\{n_1 = |V(G_1)|, n_2 = |V(G_2)|\}$. More precisely, after $\mathcal{O}(n^3)$ time preprocessing, any distance in the product can be computed in constant time. (Of course, one needs $\mathcal{O}(n_1^2 n_2^2)$ for writing down all distances.)

2 Preliminaries

We will consider only undirected finite graphs without loops and multiple edges which are called *simple* graphs. A graph G is called *bipartite* if its vertex set can be represented as the union of two disjoint sets V_1 and V_2, such that every edge of G has one endpoint in V_1 and other endpoint in V_2. H is *subgraph* of G, when H is a graph satisfying $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. A *neighborhood* of a vertex u is the set $N(u) = \{v \mid uv \in E(G)\}$. A *path* P_k is a sequence of $k \geq 1$ distinct vertices $v_1, v_2, \ldots, v_k$ such that $v_i v_{i+1}$, $1 \leq i < k$ are edges of G. The *length* of a path P_k is number of its edges, $L(P) = k-1$. A *walk* between u and v in G is a path in which vertices are not necessary distinct. Graph G is connected if there is a path between any two vertices u, v of G. Let u and v be arbitrary vertices of G. The *distance* between u and v in G is the length of a shortest path from u to v and infinite if no such path exist. The *even distance* between u and v in G is the length of a shortest even walk from u to v and infinite if no such walk exist. The *odd distance* between u and v in G is the length of a shortest odd walk from u to v and infinite if no such walk exist. We denote the distance between u and v with $d_G(u,v)$, the even distance with $d_G^e(u,v)$ and the odd distance with $d_G^o(u,v)$. It is clear that even (odd) distance of a bipartite graph is either equal to the distance of a graph G or it is infinite, since all walks between two vertices are even or all walks are odd. For a later reference we formally write an observation. Easy proof is omitted.

Lemma 1 *A connected graph G is bipartite if and only if there are two vertices u and v such that exactly one of $d_G^e(u,v)$ and $d_G^o(u,v)$ is infinite.*

The *direct product* of graphs G_1 and G_2 is a graph on the vertex set $V(G_1) \times V(G_2)$. Two vertices (u_1, u_2) and (v_1, v_2) are adjacent when $u_1 v_1 \in E(G_1)$ and $u_2 v_2 \in E(G_2)$. The mappings $p_1 : (u,v) \mapsto u$ and $p_2 : (u,v) \mapsto v$ are called *projections* from $G_1 \times G_2$ onto the factors G_1, and G_2, respectively. It is easy to see that the projections p_1 and p_2 map edges to edges in the direct product. This projection property may be a reason that the direct product is in some respect more difficult than other standard products. For example, for connectivity of the product graph it is not enough that both factors are connected, as holds for the cartesian and the strong product. The result for direct product is due to Weichsel [5], and here it is given as Corollary 5 to show a simple application of the theorem Theorem 4. Contrary to the unique factorization properties for the cartesian and the strong product, the direct product can have nonunique factorization in class of simple graphs. Unique factorization holds in the class of nonbipartite, connected graphs where every vertex has a loop. The last result on this topic is due to Imrich [2]. There are several open problems related to the direct product. Perhaps the most famous is the Hedetniemi conjecture [1]: $\chi(G_1 \times G_2) = \min\{\chi(G_1), \chi(G_2)\}$, where $\chi(G)$ denotes the chromatic number of a G.

3 Main result

We are now able to prove main result, but first we give a Lemma and for brevity introduce some new notation. Let (u_1, u_2) and (v_1, v_2) be arbitrary vertices of $G_1 \times G_2$. We will use following symbols:

$$\begin{aligned} d_{G_1\times G_2} &= d_{G_1\times G_2}((u_1,u_2),(v_1,v_2)), \\ d^e_{G_1\times G_2} &= d^e_{G_1\times G_2}((u_1,u_2),(v_1,v_2)), \\ d^o_{G_1\times G_2} &= d^o_{G_1\times G_2}((u_1,u_2),(v_1,v_2)). \end{aligned}$$

The following Lemma is an obvious corrolary of the definition of even and odd distance.

Lemma 2 *Let G be a graph. Then*

$$d_G(u,v) = \min\{d^e_G(u,v), d^o_G(u,v)\}.$$

Lemma 3 *Let G_1 and G_2 be any connected graphs on at least two vertices. Then*

$$\begin{aligned} d^e_{G_1\times G_2} &= \max\{d^e_{G_1}(u_1,u_2), d^e_{G_2}(v_1,v_2)\}, \\ d^o_{G_1\times G_2} &= \max\{d^o_{G_1}(u_1,u_2), d^o_{G_2}(v_1,v_2)\}. \end{aligned}$$

Proof. We will prove only the first equality. The proof of the second is analogous.

Let (u_1, u_2) and (v_1, v_2) be arbitrary vertices of $G_1 \times G_2$. Let us denote with $x_0x_1 \dots x_k$ some shortest even walk between $u_1 = x_0$ and $v_1 = x_k$ in G_1 and with $y_0y_1 \dots y_k$ some shortest even walk between $u_2 = y_0$ and $v_2 = y_k$ in G_2. If some of the walks do not exist, the corresponding maximum is infinite and is more or equal to $d^e_{G_1\times G_2}$. We can assume without loss of generality that

$$d^e_{G_1}(u_1,v_1) = k \geq l = d^e_{G_2}(u_2,v_2).$$

Then

$$(x_0, y_0)\,(x_1, y_1)\ldots(x_l, y_l)\,(x_{l+1}, y_{l-1})\,(x_{l+2}, y_l)\ldots\big(x_{l+i}, y_{l-(i \bmod 2)}\big)\ldots(x_k, y_l)$$

is an even walk between (u_1, v_1) and (u_2, v_2) in $G_1 \times G_2$ since $k - l$ is an even number. Thus we have

$$d^e_{G_1 \times G_2} \leq \max\{d^e_{G_1}(u_1, v_1), d^e_{G_2}(u_2, v_2)\}.$$

Now let P be a shortest even walk between vertices (u_1, v_1) and (u_2, v_2) in $G_1 \times G_2$. If such a walk does not exist, the $d^e_{G_1 \times G_2}$ is infinite and we are done. Each edge of P is mapped in to an edge in G_1, resp. G_2 by the projection p_1, resp. p_2. So $p_1(P)$, resp. $p_2(P)$ is an even walk from u_1, resp. u_2 to v_1, resp. v_2 in G_1, resp. G_2 of length $L(P)$. Thus

$$d^e_{G_1}(u_1, v_1) \leq L(p_1(P)) = L(P) = d^e_{G_1 \times G_2},$$
$$d^e_{G_2}(u_2, v_2) \leq L(p_2(P)) = L(P) = d^e_{G_1 \times G_2}.$$

So we have

$$d^e_{G_1 \times G_2} \geq \max\{d^e_{G_1}(u_1, v_1), d^e_{G_2}(u_2, v_2)\}.$$

■

Both Lemmas give us the main result.

Theorem 4 *Let G_1 and G_2 be any connected graphs on at least two vertices. Then*

$$d_{G_1 \times G_2} = \min\{\max\{d^e_{G_1}(u_1, v_1), d^e_{G_2}(u_2, v_2)\}, \max\{d^o_{G_1}(u_1, v_1), d^o_{G_2}(u_2, v_2)\}\}.$$

As an example of application of the theorem we will give a short proof of the following corollary. The result was first proved by Weichsel [5].

Corollary 5 *Let G_1 and G_2 be connected graphs with at least one edge. Then $G_1 \times G_2$ is connected if and only if at least one of G_1 and G_2 is nonbipartite.*

Proof. Let G_1 and G_2 be both bipartite, u vertex of G_1 and v, w adjacent vertices of G_2. There is no odd walk in G_1 which starts and ends in u and no even walk in G_2 from v to w. Thus by Theorem 4 $d_{G_1 \times G_2}((u, v), (u, w))$ is infinite and $G_1 \times G_2$ is not connected.

Assume now that $G_1 \times G_2$ is not connected and let $(u_1, u_2), (v_1, v_2)$ be vertices from two different connected components. By Theorem 4 both $d^e_{G_1 \times G_2}$ and $d^o_{G_1 \times G_2}$ are infinite. By Lemma 3 at least one of $d^e_{G_1}(u_1, v_1)$ and $d^e_{G_2}(u_2, v_2)$ is infinite. Without loss of generality assume that $d^e_{G_1}(u_1, v_1)$ is infinite. Then $d^o_{G_1}(u_1, v_1)$ must be finite, because G_1 is connected. Again by Lemma 3, at least one of $d^o_{G_1}(u_1, v_1)$ and $d^o_{G_2}(u_2, v_2)$ must be infinite, and therefore $d^o_{G_2}(u_2, v_2)$ is infinite. Recalling that G_1 and G_2 are connected, it follows that both G_1 and G_2 are bipartite by Lemma 1. ■

As suggested by Pisanski, the Theorem 4 may have an application in studies of growth of direct product of graphs, see [4].

4 Algorithm

We will present a simple polynomial algorithm for computing the even and odd distance of an arbitrary graph G with time complexity $\mathcal{O}(n^3)$. For this we need two simple lemmas and the concept of G^s, the *Boolean square* of G. G^s is a graph on a vertex set $V(G)$. Two vertices u and v are adjacent in G^s if (and only if) there exist a vertex w that uw and wv are edges in G.

The correctness of the algorithm is based on the following two easily proved lemmas. We give short proofs for completeness of presentation.

Lemma 6 *Let G be a (connected) graph. Then $d_G^e(u,v) = 2d_{G^s}(u,v)$.*

Proof. Let u and v be some arbitrary vertices of G. Let $P = x_0x_1 \ldots x_k$ be a shortest even walk from $u = x_0$ to $v = x_k$. Then $x_0x_2x_4 \ldots x_k$ is a walk in G^s and we have $d_G^e(u,v) \leq 2d_{G^s}(u,v)$. Let now $P = y_0y_1 \ldots y_k$ be some shortest path between $u = y_0$ and $v = y_k$ in G^s. By the definition of G^s there exists such vertices $w_0, w_1, \ldots, w_{k-1}$ in G that $y_0w_0y_1w_1y_2 \ldots y_{k-1}w_{k-1}y_k$ is a walk from u to v in G. Thus $d_G^e(u,v) \geq 2d_{G^s}(u,v)$. ■

Lemma 7 *Let G be a (connected) graph. Then*

$$d_G^o(u,v) = \min_{w \in N(u)} \{1 + d_G^e(w,v)\}.$$

Proof. It is obvious that the odd distance from u to v is 1 plus even distance from v to some neighbor of u. ■

Algorithm 8 ***Even and odd distance***
Input: *The adjancecy list of (connected) graph G.*
Output: *Even distance matrix $(D_{u,v}^e)$, odd distance matrix $(D_{u,v}^o)$.*
1. Generate graph G^s (in $\mathcal{O}(mn)$ time).
2. Generate matrix $(D_{u,v})$ of distances in G^s (in $\mathcal{O}(n^3)$ time).
3. $(D_{u,v}^e) = 2(D_{u,v})$. (in $\mathcal{O}(n^2)$ time).
4. Compute $(D_{u,v}^o)$ using Lemma 7. (in $\mathcal{O}(n^3)$ time).

More details and remarks for each step are given below

1. Graph G^s can be generated in time $O(mn)$ as follows:

   For every $e = uv \in E(G)$ and for every $w \in N(v)$ do

   connect u and w in G^s.

2. The distance matrix of G^s can be generated by running a BFS (breadth first search) from each vertex. Note that the number of edges of G^s may be $\mathcal{O}(n^2)$, therefore the time complexity of this step is $\mathcal{O}(n^3)$.

3. $(D^e_{u,v}) = 2(D_{u,v})$ clearly needs $\mathcal{O}(n^2)$ time.

4. The following clearly computes the odd distances in time $\mathcal{O}(mn)$.

   For every pair of vertices $u, v \in V(G)$ do

   $D^o_{u,v} = \infty$,

   For every $w \in N(u)$

   $$D^o_{u,v} = \min\{D^o_{u,v}, 1 + D^e_{w,v}\}.$$

References

[1] S. Hedetniemi, Homomorphisms of graphs and automata, *Tehnical Report 03105-44-T,* University of Michigan, 1966.

[2] W. Imrich, Factoring cardinal product graphs in polynomial time, *Discrete Math.,* 192 (1998) 119-144.

[3] W.Imrich and S.Klavžar, Product Graphs: Structure and Recognition, *John Wiley & Sons*, New York, 2000.

[4] T. Pisanski, T. W. Tucker, Growth in products of graphs, Preprint series no. 762, IMFM Ljubljana, 2001.

[5] P.M. Weichsel, The Kronecker product of graphs, *Proc. Amer. Math. Soc.,* 13 (1962) 47-52.