P_3 - Forcing in Honeycomb Networks

Jessy Sujana.G[1], T.M. Rajalaxmi[2]
[1]Department of Computer Science,
SSN College of Engineering, Chennai-603 110, India
[2]Department of Mathematics,
SSN College of Engineering, Chennai-603 110, India
jessy_sujana@yahoo.co.in

Abstract

In this paper we compute the P_3-forcing number of honeycomb network. A dynamic coloring of the vertices of a graph G starts with an initial subset S of colored vertices, with all remaining vertices being non-colored. At each discrete time interval, a colored vertex with exactly one non-colored neighbor forces this non-colored neighbor to be colored. The initial set S is called a forcing set of G if, by iteratively applying the forcing process, every vertex in G becomes colored. If the initial set S has the added property that it induces a subgraph of G whose components are all paths of length 3, then S is called a P_3-forcing set of G. A P_3-forcing set of G of minimum cardinality is called the P_3-forcing number of G denoted by $ZP_3(G)$.

Keywords: dynamic coloring, honeycomb networks,forcing set.

1 Introduction

For electric power companies, the continuous monitoring of their systems represents a crucial task. One way to accomplish it consists of placing phase measurement units (PMU) at selected locations in the system. Because of the high cost of a PMU, it is desirable to minimize the number of PMUs used, while maintaining the ability to monitor the entire system. The power system monitoring problem, as introduced in [1], asks for the minimum number of PMUs, and their locations, needed to monitor an electric power system. This problem has been formulated as a graph domination problem by Haynes et al., [6]. However, this type of domination has a different flavor than the standard domination type problem, since the application of the domination rules can be iterated.

Let $G = (V, E)$ be a graph representing an electric power system, where a vertex represents an electrical node and an edge represents a transmission line joining two electrical nodes.

A set $S \subseteq V$ is a dominating set in G if every vertex in $V \setminus S$ has at least one neighbor in S. The minimum cardinality of a dominating set of G is its domination number, denoted by $\gamma(G)$ [2].

Definition 1.1. *For a graph G and a set $T \subseteq V(G)$, the closure of T in G denoted by $C_G(T)$ is recursively defined as follows: Start with $C_G(T) = T$. As long as exactly one of the neighbors of some element of $C_G(T)$ is not in $C_G(T)$, add that neighbor to $C_G(T)$. If $C_G(T) = V(G)$ at some stage, then T is a forcing set of G. A forcing set of minimum cardinality is called the forcing number and is denoted by $Z(G)$.*

This problem may also be viewed as coloring the vertices of G using propagation rules [5]. Let every vertex be initially colored either black or white. If u is a black vertex of G and u has exactly one white neighbor, say v, then we change the color of v to black; this rule is called the color change rule. In this case we say "u forces v" which is denoted by $u \to v$. The procedure of coloring a graph using the color rule is called a zero forcing process or simply a forcing process. Given an initial coloring of G, in which a set of the vertices is black and all other vertices are white, the derived set is the set of all black vertices resulting from repeatedly applying the color change rule until no more changes are possible. If the derived set for a given initial subset of black vertices is the entire vertex set of the graph, then the set of initial black vertices is called a forcing set [13][4].

Forcing process is utilized to study the inverse Eigen value problems, PMU placement problems, and quantum control problems. The forcing process is also called graph infection or graph propagation in the zones identified with quantum dynamics and control theory of quantum mechanical systems. By the monotonous utilization of a similar quantum transformation, this reality has been used to accomplish noise protection, cooling, state preparation, and quantum state transfer. Forcing is also used in computer science in the context of fast-mixed searching.

Chemical structures are conveniently represented by graphs, where atoms correspond to vertices and chemical bonds correspond to edges. This representation inherits much useful information about chemical properties of molecules. [3]. In this paper, we introduce P_3-forcing Problem as follows:

Definition 1.2. *Let G be a graph and let be the set of all P_3 paths in*

G. For a set T of independent P_3 paths in ρ, define $C_G(T)$, the closure of T, as $C_G(S)$ when S is the set of all vertices in the paths in ρ. If $C_G(T) = V(G)$, then T is called a P_3-forcing set of G. The minimum cardinality of a P_3 forcing set of G is the P_3 forcing number of G and is denoted by $ZP_3(G)$. The P_3-forcing problem of a graph G is to determine $ZP_3(G)$.

Analogous to the coloring of vertices in a zero forcing set, we describe coloring of vertices originating from a set of P_3 paths. Let G be a graph in which every vertex is initially colored either black or white. Let P be a path on three vertices, say u, v, w, all of which are colored black. If u, v or w is adjacent to exactly one white neighbor, say x, then we change the color of x to black; this rule is called the color change rule. In this case we say "P forces x" which is denoted by $P \to x$. At a time, P_3 may force three vertices. The procedure of coloring a graph using the color rule is called simply a forcing process. Given an initial coloring of G in which a set of P_3 paths is black and all other vertices are white, the derived set is the set of all black vertices resulting from repeatedly applying the color change rule until no more changes are possible. If the derived set for a given initial subset of black vertices is the entire vertex set of the graph, then the set of initial P_3 paths is called a P_3-forcing set.

The additional condition on a forcing set that it is composed of paths of length 3 ensures more reliability.

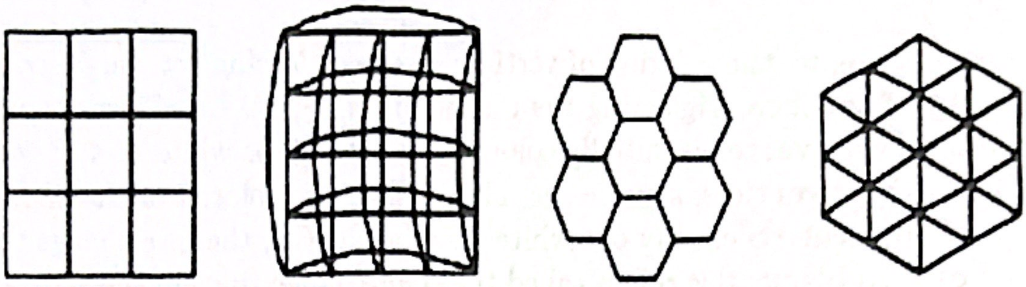
If a vertex v forces u, then v is called a 'live' vertex and u is said to be covered by v. A vertex which is adjacent to two vertices which are not already covered is called a 'dead' vertex.

In this paper we study P_3-forcing problem in honeycomb networks.

2 Honeycomb Networks

Multiprocessor interconnection networks are often required to connect thousands of homogeneously replicated processor-memory pairs, each of which is called a processing node. Instead of using a shared memory, all synchronization and communication between processing nodes for program execution is often done via message passing [7][3]. Design and use of multiprocessor interconnection networks have recently drawn considerable attention due to the availability of inexpensive, powerful microprocessors and memory chips [10].

It is known that there exist three regular plane tessellations, composed of the same kind of regular polygons: triangular, square, and hexagonal. They are the basis for the design of direct interconnection networks with highly competitive overall performance. Grid connected computers and tori see Fig. 1(a) and (b), are based on regular square tessellations, and are popular and well-known models for parallel processing.



(a) 4 x 4 Grid network (b) Torus network (c) Honeycomb network (d) Hexagonal network

Figure 1: Regular plane tessellations

Built recursively using the hexagon tessellation [11], honeycomb networks, see Figure 1(c), are widely used in computer graphics, cellular phone base stations [8], image processing, and in chemistry as the representation of benzenoid hydrocarbons. Honeycomb networks are better in terms of degree, diameter, and total number of links, cost and the bisection width than mesh connected planar graphs. Stojmenovic [11] has studied the topological properties of honeycomb networks, routing in honeycomb networks and honeycomb torus networks. Parhami [9] gave a unified formulation for the honeycomb and the diamond networks [12].

Honeycomb meshes offer a model for multiprocessor interconnection networks with similar properties to those of mesh-connected computer networks, also referred to as grid graphs [8][9]. To define the honeycomb mesh we will use the following notation: for a given $n \in Z$, we denote by $[n]$ the set $\{-n+1, -n+2, \ldots, -1, 0, 1, 2, \ldots, n\}$

Definition 2.1. *The hexagonal honeycomb mesh of dimension* $n \geq 1$, $n \in Z$, $HM(n)$, *has vertex set* $V(HM(n)) = (x, y, z)x, y, z[n]and1 \leq x+y+z \leq 2$ *and two vertices* (x_1, y_1, z_1) *and* (x_2, y_2, z_2) *are adjacent if and only if* $|x_1 - x_2| + |y_1 - y_2| = 1$.

$HM(1)$ is one simple hexagon. The honeycomb mesh of dimension 2, $HM(2)$, is obtained by adding six hexagons to the boundary edges of $HM(1)$. In general, the honeycomb mesh of dimension t, $HM(t)$, is obtained by adding a layer of hexagons around the border of $HM(t\text{-}1)$. The dimension of $HM(n)$ represents the number of layers of hexagons between $HM(1)$ and the border of $HM(n)$. $HM(n)$is a bipartite graph with the bipartite sets $V_1 = \{x, y, z / x, y, z \in [n] \text{ and } x + y + z = 1\}$ and $V_2 = \{x, y, z / x, y, z \in [n] \text{ and } x + y + z = 2\}$. See Fig.2.

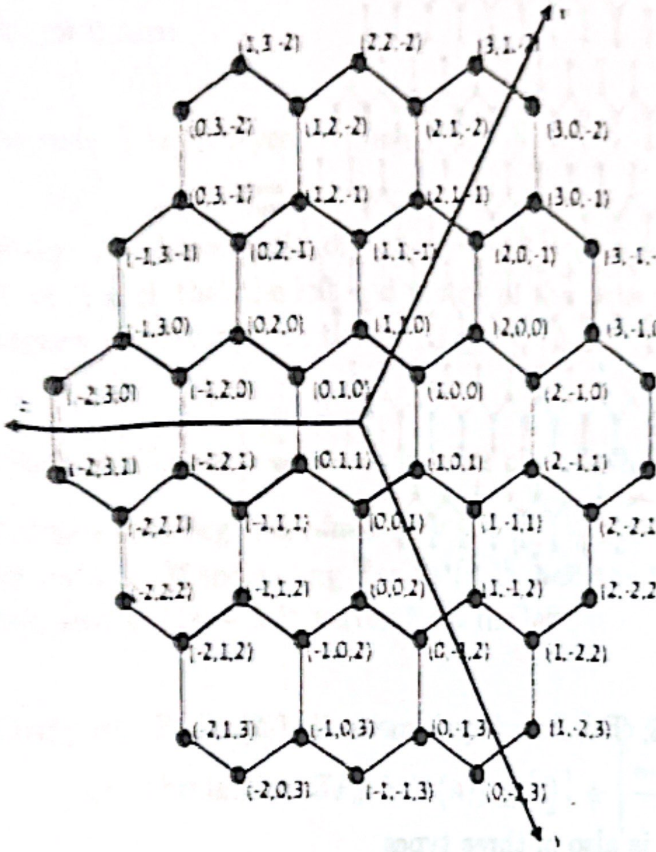


Figure 2: The labeled honeycomb mesh $HM(3)$

3 P_3-Forcing in Honeycomb networks

Definition 3.1. *For $1 \leq i \leq 2n$, Row(i) of $HM(n)$ is a line induced by the vertices of $HM(n)$ with the third co-ordinate equal to n-i+1.*

Lemma 3.2. *Let $HM(n)$ be the honeycomb network of dimension n, then the set of rows Row(i), $1 \leq i \leq \lceil \frac{n}{2} \rceil$ induces a set of alternate dead vertices in Row $\lceil \frac{n}{2} \rceil$.*

The edges of a $HM(n)$ are of three types
(1) an obtuse edge
(2) an acute edge
(3) a vertical edge
as shown in Fig. 3

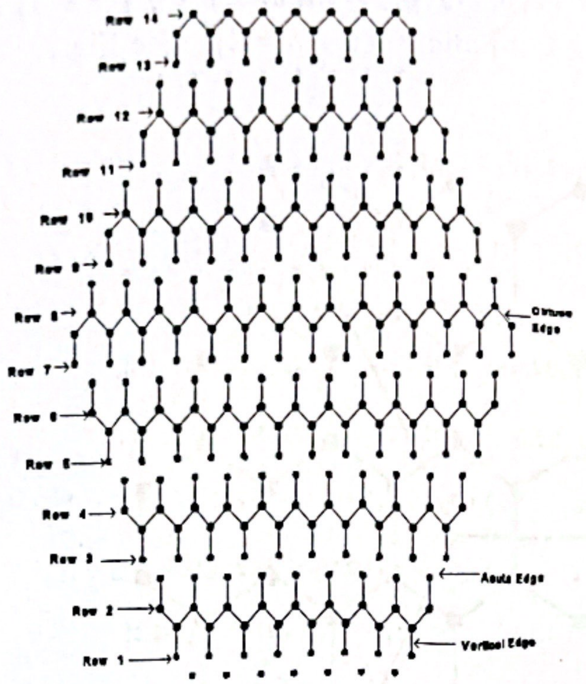


Figure 3: The honeycomb mesh $HM(7)$

A path P_3 in $HM(n)$ is also of three types

Type 1: One edge is obtuse and other end is acute. From the left, if the first edge is obtuse, then the two end vertices are called 'peak vertices' and vertex of degree 2 is called a 'trough vertex'. From the left, if the first edge in P_3 is acute, then the two end vertices are called 'trough vertices' and the vertex of degree 2 is called 'peak vertex'.

Type 2: One edge is vertical and the other is obtuse.

Type 3: One edge is vertical and the other edge is acute.

Lemma 3.3. *If every vertex in Row(n) is a live vertex then they cover all vertices in Row(i),* $n+1 \leq i \leq 2n$.

Proof: The peak vertices in Row(n) and the trough vertices in Row(n+1) induce a perfect matching. The acute edges in Row(n+1) also form a perfect matching such that each trough node is adjacent to exactly one node which is a peak node. This sequence continues till we reach Row($2n$).

We now propose an algorithm which determines an upper bound for $ZP_3(HM(n))$.

Algorithm:

Input: The Honeycomb mesh $HM(n)$

Step 1: Choose paths of Type α in $\lceil \frac{n}{2} \rceil$ consecutive rows beginning from Row 1 such that the left end vertex of the path in Row i is the $(4i-2)^{th}$ vertex of the row from the left, $1 \leq i \leq \lceil \frac{n}{2} \rceil$.

Step 2: Choose $\left\lfloor \frac{n - \lceil \frac{n}{2} \rceil + 1}{2} \right\rfloor$ number of P_3 paths of Type β, one each in alternate rows beginning from Row $m = \lceil \frac{n}{2} \rceil + 2$ and ending with $m = (n-1)$ or $m = n$. If the ending Row is (n-1), add the last P_3 in Row n with its left end as $(4m-5)^{th}$ vertex from the left.

Output: Paths chosen using Step 1 and Step 2 cover all the vertices of $HM(n)$. This implies $ZP_3(HM(n)) \leq \lceil \frac{n}{2} \rceil + \left\lceil \frac{n - \lceil \frac{n}{2} \rceil + 1}{2} \right\rceil$.

Proof of Correctness:

We label all vertices covered by the P_3 path in i^{th} Row as i. The i^{th} path chosen in step 1, labels all its peak vertices to its left as i and the first peak vertex to its right also as i. When the P_3 path in Row $\lceil \frac{n}{2} \rceil$ is chosen, all vertices in that row are labeled $\lceil \frac{n}{2} \rceil$. Further every vertex in Rows 1 to $\lceil \frac{n}{2} \rceil$ are labeled from $\{1, 2, \ldots, \lceil \frac{n}{2} \rceil\}$. Since there is a perfect matching between Row $\lceil \frac{n}{2} \rceil$ and Row $(\lceil \frac{n}{2} \rceil + 1)$ all the trough vertices of Row $(\lceil \frac{n}{2} \rceil + 1)$ are labeled $\lceil \frac{n}{2} \rceil$. But each trough vertex is adjacent to two peak vertices which are not covered already. Hence these trough vertices in Row $(\lceil \frac{n}{2} \rceil + 1)$ constitute a set of independent dead vertices.

The rows selected in Step 2 in Row $(\lceil \frac{n}{2} \rceil + 1)$, labels all vertices in Row $(\lceil \frac{n}{2} \rceil + 1)$ and all dead vertices in Row $\lceil \frac{n}{2} \rceil$ as $(\lceil \frac{n}{2} \rceil + 1)$. This process continues choosing path P_3 in alternate rows till we reach $(n-1)^{th}$ or n^{th} Row. If it is $(n-1)^{th}$ Row, to cover the vertices in Row n, we choose one more path in Row n.

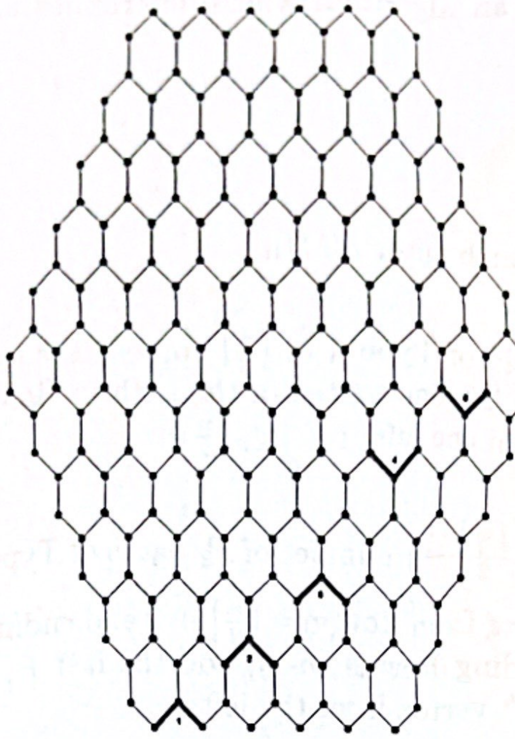


Figure 4: Choice of P_3 paths in $HM(6)$

Thus we have the following result.

Theorem 3.4. $ZP_3(HM(n)) \leq \lceil \frac{n}{2} \rceil + \left\lceil \frac{n - \lceil \frac{n}{2} \rceil + 1}{2} \right\rceil$

See Fig. 4 and Fig. 5

A path P_3 selected in Row(i), $1 \leq i \leq \lceil \frac{n}{2} \rceil$ using the algorithm leaves all the paths of Type 2 to its left covered. On the right it covers one vertex in the first step. The $\lceil \frac{n}{2} \rceil$ paths selected in the first $\lceil \frac{n}{2} \rceil$ rows together cover all vertices of $HM(n)$, till they reach a dead line. The number of paths cannot be reduced. Thus we have the following conjecture.

Conjecture:

$$ZP_3(HM(n)) = \lceil \tfrac{n}{2} \rceil + \left\lceil \frac{n - \lceil \frac{n}{2} \rceil + 1}{2} \right\rceil$$

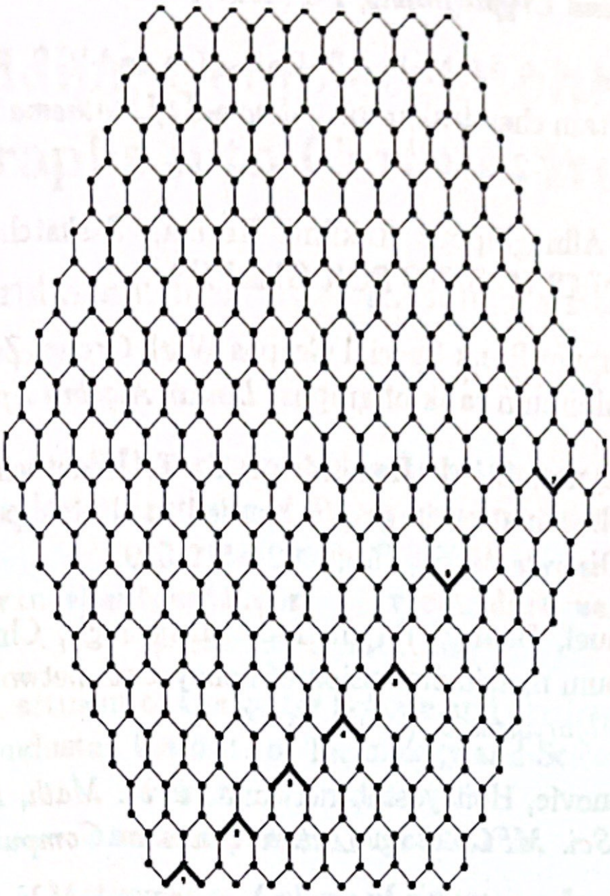


Figure 5: Choice of P_3 paths in $HM(9)$

4 Conclusion

Honeycomb networks are applied widely in the field of cellular networks where the P_3-forcing number determines the minimum number of cellular towers to be placed with maximum coverage. In this paper we have obtained an upper bound for the P_3-forcing number of honeycomb networks. We have also posed a conjecture that the upper bound cannot be reduced any further. Obtaining optimal P_3-forcing number will be helpful in the communication networking fields.

References

[1] T. L. Baldwin, L. Milli, M. B. Boisen, Jr. and A. Adapa, Power system observability with minimal phasor measurement placement, *IEEE Trans. Power Syst.*, 8, (1993), 707-715.

[2] Daniela Ferrero, Seema Varghese, A. Vijayakumar, Power domination in honeycomb networks, *Journal of Discrete Mathematical*

Sciences and Cryptography, 14, (2011), 521-529.

[3] J. Quadras, A. S. M. Mahizl, I. Rajasingh and R. S. Rajan, Domination in certain chemical graphs, *Journal of Mathematical Chemistry*, 53, (2015), 207-219.

[4] Fatemeh Alinaghipour Taklimi Regina, Saskatchewan, Thesis-ZERO FORCING SETS FOR GRAPHS

[5] AIM Minimum Rank-Special Graphs Work Group, Zero forcing sets and the minimum rank of graphs. *Linear Algebra Appl.*, 7, (2008)

[6] T. W. Haynes, S. M. Hedetniemi, S. T. Hedetneimi and M. A. Henning, Domination in graphs applied to electric power networks, *SIAM J.Discrete Math.*, 15, (2002), 519-529.

[7] Paul Manuel, Bharati Rajan, Indra Rajasingh, Chris Monica M, On minimum metric dimension of honeycomb networks, *Journal of Discrete Algorithms* 6, (2008), 20-27.

[8] I. Stoijmenovic, Honeycomb networks, *Proc. Math. Foundations of Comput. Sci. MFCS95 on Lecture Notes in Comput. Sci.*, (1995), 267-276.

[9] I. Stoijmenovic, Honeycomb networks: topological properties and communication algorithms, *IEEE Transactions on Parallel and Ditributed Systems*, 8, (1997), 1036-1042.

[10] M.S. Chen, K.G. Shin, D.D. Kandlur, Addressing, routing, and broadcasting in hexagonal mesh multiprocessors, *IEEE Transactions on Computers* 39, (1990), 10-18.

[11] J. Quadras, K. Balasubramanian and K. Arputha Christy, Analytical expressions for Wiener indices of n-circumscribed pericondensed benzenoid graphs, *Journal of Mathematical Chemistry*, DOI 10.1007/s10910-016-0596-9.

[12] X.Yang, G. M. Megson, Y. Tang and D. J. Evans, Diameter of Parallelogramic Honeycomb Torus, *An International Journal of Computer and Mathematics with Applications*, 50, (2005), 1477-1486.

[13] C. J. Edholm, L. Hogben, M. Huynh, J. Lagrange, D. D. Row, Vertex and edge spread of zero forcing number, maximum nullity, and minimum rank of a graph, *Linear Algebra Appl.*, 436, (2012), 4352-437.