Published in J. Phys. A: Math. Theor. 44 (2011) 205102 (11pp)

Label-based routing for a family of scale-free, modular, planar and unclustered graphs.

Francesc Comellas

Departament de Matemàtica Aplicada IV, EPSC, Universitat Politècnica de Catalunya
c/ Esteve Terradas 5, 08860 Castelldefels, Barcelona, Catalonia, Spain

E-mail: comellas@ma4.upc.edu

Alicia Miralles

Departament de Matemàtica Aplicada IV, EPSC, Universitat Politècnica de Catalunya
c/ Esteve Terradas 5, 08860 Castelldefels, Barcelona, Catalonia, Spain

E-mail: almirall@ma4.upc.edu

Abstract. We give an optimal labeling and routing algorithm for a family of scale-free, modular and planar graphs with clustering zero. Relevant properties of this family match those of some networks associated with technological and biological systems with a low clustering, including some electronic circuits and protein networks. The existence of an efficient routing protocol for this graph model should help when designing communication algorithms in real networks and also in the understanding of their dynamic processes.

PACS numbers: 02.10.Ox, 05.45.Df, 89.20.Ff, 89.75.Da, 89.75.-k

1. Introduction

The study of complex systems is often associated with small-world networks which have an scale-free degree distribution, are highly clustered and, in some cases, are modular. However, there are relevant networks which have a low clustering and a high modularity. This is the case of some electronic circuits and protein networks, as shown in [5]. In [7] we introduced a simple tunable family of scale-free, small-world, self-similar, planar graphs with clustering zero to model these particular systems. Here we provide an optimal labeling and routing for the graphs.

To make the paper self-contained we recall from [7] relevant characteristics and parameters of the graphs, namely their diameter, average distance, clustering, degree distribution and degree correlation coefficient. The vertex labeling introduced in this paper is optimal in length and allows to produce, only from the labels, a shortest path and an optimal routing between any two vertices. Therefore, this family of graphs, with its labeling and routing protocol, is a good mathematical model for relevant real-life systems which are associated with networks that have a low clustering, are almost planar and display an scale-free degree distribution. Thus, the model constitutes a new tool to generate and study new practical algorithms for these technological and biological systems and should also help in the understanding of the underlying mechanisms which shaped their particular topologies.

2. Generation and properties of the graphs $M_d(t)$

We recall here the definition and main properties of the family of graphs introduced by the authors in [7]. These graphs, denoted by $M_d(t)$, are scale-free, small-world. modular, self-similar and planar. They generalize the family introduced and studied in [2], which corresponds to the particular case $d = 1$. However, in this case, the graphs are not scale-free.

The graphs $M_d(t)$ are constructed iteratively by introducing, at each step t, new vertices and edges in a deterministic way, but they have also an equivalent modular recursive construction.

In the iterative construction method, a *generating edge* is any edge whose endvertices have been introduced at different iteration steps. The only edge of $M_d(0)$ is also a generating edge. All other edges are called *passive edges*. A generating edge becomes passive at the next iteration step.

Definition 2.1 *The graph $M_d(t)$ is constructed iteratively as follows:*

For $t = 0$, $M_d(0)$ has two vertices and a generating edge connecting them.

For $t \geq 1$, $M_d(t)$ is obtained from $M_d(t-1)$ by adding to every generating edge in $M_d(t-1)$, d parallel paths P_4 of length three by identifying the two final vertices of each path with the endvertices of the generating edge.

This process is repeated until the desired graph order is reached, see Fig. 1. We note that the graph order can be also adjusted with the parameter d (number of parallel paths that are attached to each generating edge).

The same graph can be produced with a recursive modular construction, as follows: For $t = 0$, $M_d(0)$ has two vertices and a generating edge connecting them.

For $t = 1$, $M_d(1)$ is obtained from $M_d(0)$ by adding to its only edge d parallel paths of length three by identifying the two final vertices of each path with the endvertices of the initial edge.

For $t \geq 2$, $M_d(t)$ is constructed from $2d$ copies of $M_d(t-1)$, by identifying, vertex to vertex, the initial edge of each $M_d(t-1)$ with the generating edges of $M_d(1)$ (Fig. 1.)

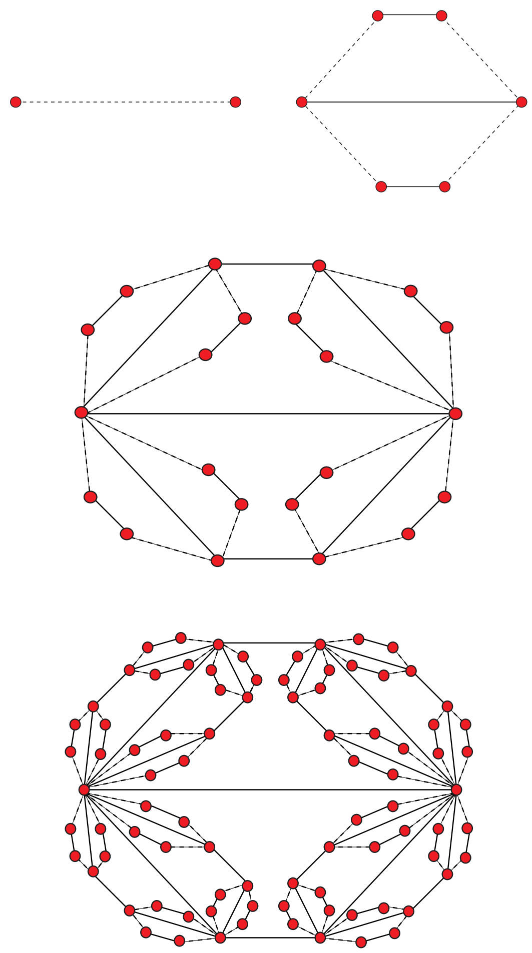

Figure 1. Graphs $M_d(t)$ produced at iterations $t = 0, 1, 2$ and 3 for $d = 2$.

Next we give some relevant properties of the graphs. Their derivation and other topological properties of the graphs $M_d(t)$ can be found in [7] :

Order and size of $M_d(t)$.– The order and size of $M_d(t)$, $t \geq 0$, are $|V(t)| = ((2d)^{t+1} + 2d - 2)/(2d - 1)$ and $|E(t)| = (3d(2d)^t - d - 1)/(2d - 1)$.

Planarity.– The graph $M_d(t)$ is planar as it can be drawn on the plane with no edges crossing. A formal proof of planarity is obtained from the known planarity test which

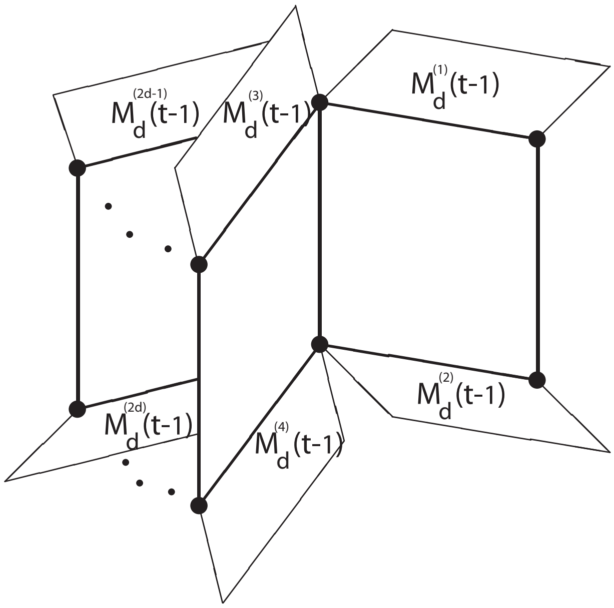

Figure 2. Recursive modular construction of $M_d(t)$ from the juxtaposition of $2d$ copies of $M_d(t-1)$, labeled $M_d^{(1)}(t-1)$, $\cdots$, $M_d^{(2d)}(t-1)$.

states that a graph is planar if it has no cycles of length 3 and $|E| \leq 2|V| - 4, |V| > 3$ or from the Kuratowski's theorem, see for example [4].

Modularity.– From the recursive construction process, we see that the graphs are modular as d copies of $M_d(t-1)$ are joined to form $M_d(t)$. Modularity can be quantified with the function Q introduced by Newman and Girvan in [9] and an algorithm to find communities in a network by maximizing it, see [1, 9, 6]. This family of graphs is highly modular. As an example, $M_3(4)$ and $M_3(5)$ display values of the modularity of $Q = 0.879$ and $Q = 0.941$, respectively.

Degree distribution.– The cumulative degree distribution of the graph $M_d(t)$ follows, for t large and $k >> 1$, a power-law distribution $P_{\rm cum}(k) \sim k^{-\gamma}$ with exponent $\gamma = \frac{\ln(2d)}{\ln(d)}$. Therefore the degree distribution is scale-free. Research on networks associated to electronic circuits show that many of them are almost planar, modular and have a small clustering coefficient and in most cases their degree distributions follow a power-law [5, 8] with exponent values in the same range than those of $M_d(t)$ [7].

Correlation coefficient.– In [7], we give an exact analytical expression for the Pearson correlation coefficient, $r(t)$, for the degrees of the endvertices of the edges of $M_d(t)$.

From the values of the correlation coefficient we see that this family of graphs has the degrees of the endvertices negatively correlated, large degree vertices tend to be connected with low degree vertices, and the graphs are disassortative.

We notice that most technological and biological networks have this property, see [8].

Diameter and average distance.– The diameter of $M_d(t)$ is $D(t) = 3 + 2 \cdot (t-1)$, $t \geq 1$. Therefore, as for t large, $t \sim \ln |V_t|$ we have in this limit that $D(t) \propto \ln |V_t|$.

The modular recursive construction of $M_d(t)$ allows to obtain also the exact analytical value of the average distance $\bar{d}(t)$ (see [7]). For a large iteration step $\bar{d}(t) \simeq t \sim \ln |V_t|$, which shows a logarithmic scaling of the average distance with the order of the graph. The diameter has the same behavior and thus the graphs are small-world.

3. Labeling of $M_d(t)$

We give here a way to label the vertices of $M_d(t)$, $t \geq 0$, which has the property that a routing by shortest paths between any two vertices is generated just from their labels. The method is a generalization of the labeling which was introduced in [3].

The labeling assigns to a vertex introduced at step $t \geq 1$, a label which is a word or string of length t made using symbols from a set $\{s_1, s_2, \cdots, s_{2d}\}$. The two initial vertices of step $t = 0$ have special labels a and b, distinct from the symbols above. This labeling is optimal in the sense that, at a given iteration step, all possible words are used and their number is exactly the same than the number of vertices. In this section we show how we assign each different label to a unique vertex, and thus that the labeling is deterministic.

3.1. Notation and preliminaries

Let $S = \{s_1, s_2, \cdots, s_{2d}\}$ be a set of $2d$ symbols. S^* is the set of all strings (words) generated by S including the null string ε. S_e^* (S_o^*) is the set of all strings generated only by the even (odd) indexed symbols from S. The length of a string w is denoted by $|w|$. The juxtaposition of sets should be interpreted as the concatenation of their elements.

We introduce the sets:

$S_o = \{s_{2i-1} \,|\, i = 1, 2, \cdots, d\}$, odd indexed symbols.

$S_e = \{s_{2i} \,|\, i = 1, 2, \cdots, d\}$, even indexed symbols.

$S^i = \{w \in S^* \,|\, |w| = i\}$, strings with length i.

$S_o^i = \{w \in S_o^* \,|\, |w| = i\}$, strings of odd indexed symbols with length i.

$S_e^i = \{w \in S_e^* \,|\, |w| = i\}$, strings of even indexed symbols with length i.

$\overline{S_o^i} = S^i - S_o^i$, complement set of S_o^i.

We define a function $r : S^* \to S^* \cup \{a, b\}$ which deletes from a word, starting from the left, all odd indexed symbols up to the first occurrence of an even indexed symbol, which is also deleted:

$$r(w) = \begin{cases} b & \text{if} \quad w \in S_o^*, \\ a & \text{if} \quad w \in S_o^* S_e, \\ w'' & \text{if} \quad w = w'sw'' \quad \text{and} \quad w' \in S_o^*,\ s \in S_e, w'' \in S^*, w'' \neq \varepsilon. \end{cases}$$

We note that, in the definition of $r(w)$, w' can be the null string ε. As examples $r(s_{2i}w'') = w''$ $(i = 1, \cdots, d,\ w'' \in S^*, w'' \neq \varepsilon)$ and $r(s_{2i}) = a$ $(i = 1, \cdots, d,)$. We also note that $|r(w)| < |w|$.

Definition 3.1 *Let $w_1, w_2 \in S^*$. The longest common suffix (LCS) of w_1 and w_2 is $c \in S^*$ satisfying $w_1 = w_1' s_{w_1} c$ and $w_2 = w_2' s_{w_2} c$ for some $w_1', w_2' \in S^*$, $s_{w_1}, s_{w_2}, \in S$ and $s_{w_1} \neq s_{w_2}$.*

3.2. Vertex labels for the graph $M_d(t)$

We provide here a definition of $M_d(t)$ which associates its vertices to strings from S^*.

Definition 3.2 *The graph $M_d(t) = (V(t), E(t))$ at step t ($t = 0, 1, \cdots$) is defined as follows:*

$$\begin{aligned}
V(0) &= \{a, b\} \\
V(t) &= S^t \cup V(t-1) = \{a, b\} \cup \bigcup_{t'=1}^{t} S^{t'} \quad (t \geq 1) \\
E(0) &= \{\{a, b\}\} \\
E(1) &= \{\{a, s\} \mid s \in S_e\} \cup \{\{b, s\} \mid s \in S_o\} \cup \{\{s_{2i-1}, s_{2i}\} \mid i = 1, 2, \cdots, d\} \cup E(0) \\
E(2) &= \{\{a, w\} \mid w \in S_o S_e\} \cup \{\{b, w\} \mid w \in S_o S_o\} \\
&\quad \cup \{\{s_{2i-1}w, s_{2i}w\} \mid w \in S, i = 1, 2, \cdots, d\} \\
&\quad \cup \{\{sw, w\} \mid sw \in \overline{S_o} S, s \in S\} \cup E(1) \\
E(t) &= \{\{a, w\} \mid w \in S_o^{t-1} S_e\} \cup \{\{b, w\} \mid w \in S_o^{t-1} S_o\} \\
&\quad \cup \{\{s_{2i-1}w, s_{2i}w\} \mid w \in S^{t-1}, i = 1, 2, \cdots, d\} \\
&\quad \cup \{\{w, r(w)\} \mid w \in \overline{S_o^{t-1}}\} \cup E(t-1) \quad (t \geq 3)
\end{aligned}$$

Fig. 3 shows the graph $M_d(2)$.

This new definition of $M_d(t)$ is equivalent to Def. 2.1 and at each step t introduces $|S^t| = (2d)^t$ new vertices and $(3d)(2d)^{t-1}$ edges, see [7] :

$$\begin{aligned}
|V(t)| &= |V(t-1)| + |S^t| = |V(t-1)| + (2d)^t \\
|E(t)| &= |E(t-1)| + |S_o^{t-1} S_o| + |S_o^{t-1} S_e| + \frac{1}{2}|S^t| + |\overline{S_o^{t-1}}| \\
&= |E(t-1)| + d^{t-1}d + d^{t-1}d + \frac{1}{2}(2d)^t + ((2d)^{t-1} - d^{t-1})(2d) \\
&= |E(t-1)| + (3d)(2d)^{t-1}.
\end{aligned}$$

We note that, since for any step $t \geq 1$ the number of vertices that are added to the graph $M_d(t)$ is equal to $(2d)^t$, this labeling is optimal in the sense that each label is a

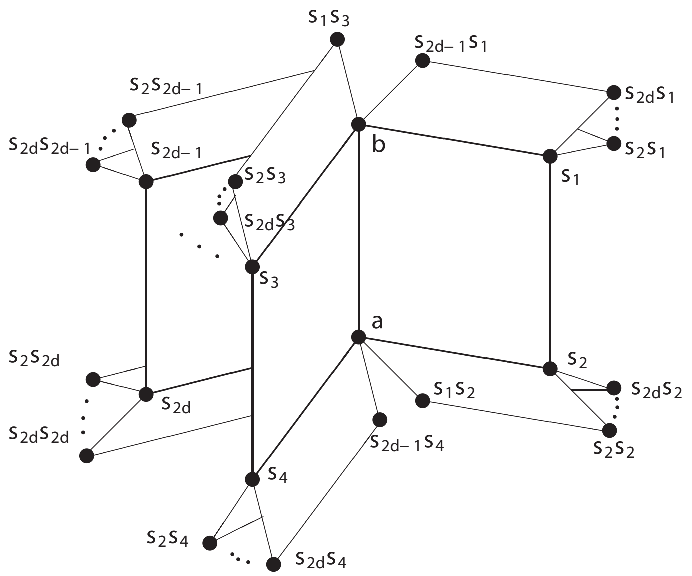

Figure 3. The graph $M_d(2)$.

different 2d-ary word of length t, Because of this, and to facilitate the reading of some proofs, we will refer, when needed, to vertices by their labels.

Definition 3.3 *The $M_d(1)$-structure $M_d(1)(w) = (V_1(w), E_1(w))$ defined by a vertex $w \in S^*$ is:*

$$\begin{aligned} V_1(w) &= \cup\{sw \mid s \in S\} \cup \{w, r(w)\}, \quad \textit{and} \\ E_1(w) &= \{\{w, r(w)\}\} \cup \{\{s_{2i-1}w, s_{2i}w\} \mid i = 1, 2, \cdots, d\} \\ &\quad \cup \{\{r(w), s_{2i-1}w\} \mid i = 1, 2, \cdots, d\} \cup \{\{w, s_{2i}w\} \mid i = 1, 2, \cdots, d\}. \end{aligned}$$

Note that $M_d(1)(w)$ is a subgraph isomorphic to $M_d(1)$, see Fig. 4.

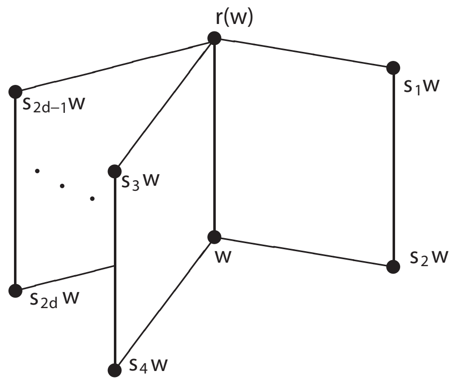

Figure 4. $M_d(1)(w)$ or $M_d(1)$-structure generated by the vertex w.

4. Routing by shortest paths in $M_d(t)$

We give here a shortest path routing protocol between any two vertices. The way to find this shortest path is peculiar, as the routing is generated both from each vertex until a common vertex is attained. However, to obtain the full routing the only information needed are the labels of the source and destination vertices.

To find a shortest path between any two vertices u and v, the routing protocol is as follows. First we compute c, the longest common suffix of u and v, and $M_d(1)(c)$. In $M_d(1)(c)$ we identify two vertices, u_f and v_f associated, respectively, with u and v. Next we produce shortest paths between u and u_f and between v and v_f. The shortest path between u_f and v_f is deduced from Fig. 4.

Proposition 4.1 *Let $u, v \in S^*$ be two vertices of $M_d(t)$, $t \geq 1$, which have c as their longest common suffix. A shortest path between u and v goes through $M_d(1)(c)$.*

Proof. If c is the LCS of u and v, vertices u and v have been generated from edge $\{c, r(c)\}$ and vertex c has been introduced at some step $1 \leq t_0 \leq t$. We prove the proposition by induction and when the vertices u and v have been introduced to the graph at the same iteration step. If they are introduced at different steps the proof follows a similar argument. Let $k = t - t_0$. For $k = 1$, and from the construction process u and v belong to $M_d(1)(c)$ and obviously any shortest path between them is in $M_d(1)(c)$. Suppose now that vertices u and v are introduced at some step $k > 1$. From the construction process there exist two vertices generated from c at step $k-1$, namely u_{k-1} and v_{k-1}, such that $u \in M_d(1)(u_{k-1})$ and $v \in M_d(1)(v_{k-1})$ and with shortest paths between them that intersect $M_d(1)(c)$. Thus, as any path from u to v goes through u_{k-1} or $r(u_{k-1})$ and v_{k-1} or $r(v_{k-1})$, it is also shortest possible and intersects $M_d(1)(c)$. Note that shortest paths between any two existing vertices of $M_d(t)$ are not modified in further iterations and their length is the same in $M_d(t+1)$ as in $M_d(t)$ $\square$

Definition 4.2 *Let $u \in S^*$ be a vertex of $M_d(t)$, $t \geq 1$, such that $u = wc$ for some $w, c \in S^*$ and $c \neq \varepsilon$. We define $u_f(c) = r^k(u)$ where k is the minimum value which verifies $r^k(u) \in V_1(c)$.*

In what follows, $d(x, y)$ denotes the distance between vertices x and y.

Lemma 4.3 *Let u be a vertex of $M_d(t)$ introduced at step $t \geq 1$ then, $d(s_{2k}u, u) = 1$ and $d(s_{2k-1}u, u) = 2$, $1 \leq k \leq d$.*

Proof. It follows from the construction process. See Fig. 5 $\square$

Lemma 4.4 *Let u be a vertex of $M_d(t)$ introduced at step $t \geq 1$. If $w \in S_o^*, |w| = k$ then $d(ws_{2l}u, u) = 1$ and $d(ws_{2l-1}u, u) = 2$, $1 \leq l \leq d$.*

Proof. Let us take $w_k = s_{o_k} \ldots s_{o_1}$ with all indices $o_1, o_2, \ldots, o_k$ odd. We prove the first equality by induction on k. By construction, it is true for $k = 1$. From Lemma 4.3, if a vertex u has been introduced at step τ, then a vertex $s_{2l}u$ is a neighbor of u introduced

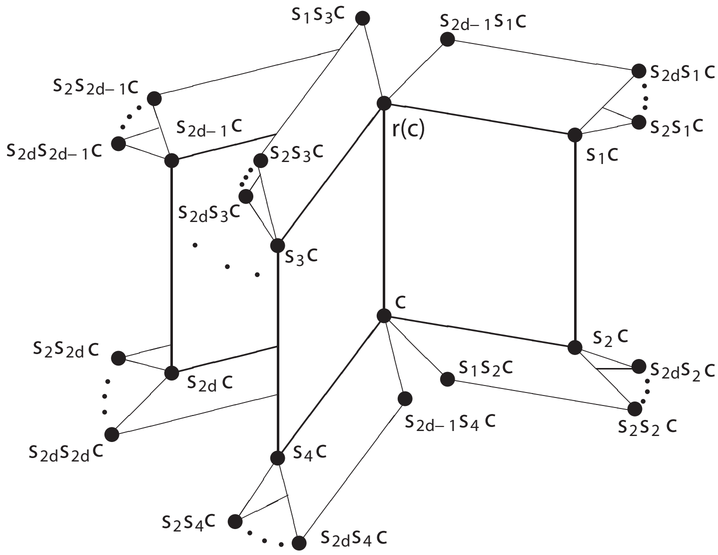

Figure 5. $M_d(1)(c)$ and $M_d(1)(s_ic)$ structures with $i = 1 \ldots 2d$. c is the longest common suffix of any two vertices. Vertex c was introduced at step t and vertex $r(c)$ at some step $t_{r(c)} < t$. We show the labeling process at steps t, $t+1$ and $t+2$.

at step $\tau+1$. The generating edge with endvertices u and $s_{2l}u$ introduces, at step $\tau+2$, new vertices $s_1s_{2l}u, s_3s_{2l}u, \ldots, s_{2d-1}s_{2l}u$ which are also neighbors of u, see Fig. 5 for $c = u$. In particular, $w_1s_{2l}u$ is a neighbor of u. By induction, if $d(w_{k-1}s_{2l}u, u) = 1$, at the next step, from Def. 3.3 with $w = w_{k-1}s_{2l}u$, vertex $r(w_{k-1}s_{2l}u)$ is adjacent to $w_ks_{2l}u$, but $r(w_{k-1}s_{2l}u) = u$ and thus $d(w_ks_{2l}u, u) = 1$. The proof for the second equality follows a similar argument. □

We verify, by construction, that the shortest path to reach $M_d(1)(c)$ (either from u or v) is through the neighbor which has the shortest label, since the length of the label indicates at which step the vertex was created. To find the neighbor ancestor, u', of vertex $u = s_{u_1}s_{u_2}\ldots s_{u_m}s_{u_{m+1}}c$ which has the shortest label, we consider two cases:

- If $s_{u_1} \in S_e$, by construction, u' is the vertex $u' = s_{u_2}\ldots s_{u_m}s_{u_{m+1}}c$. Note that $d(u, u') = 1$.
- If $s_{u_1}, s_{u_2}, \ldots, s_{u_k} \in S_o$ and $s_{u_{k+1}} \in S_e$ with $k+1 \leq m$, u' is the vertex $u' = s_{u_{k+2}}\ldots s_{u_m}s_{u_{m+1}}c$. Note by Lemma 4.4 that $d(u, u') = 1$.

Note that in both cases we can also find the neighbor ancestor of a vertex by applying the function r, thus $u' = r(u)$.

The next proposition provides a routing by shortest path between two vertices u and v with labels different from b and such that vertex u is a suffix of vertex v. It gives also the distance between them.

Lemma 4.5 *Let u and v be two vertices of $M_d(t)$, $t \geq 1$, such that $v = s_{i_1}\ldots s_{i_n}s_{i_{n+1}}u$. Let $i_{j_1}, i_{j_2}, \ldots, i_{j_k}$ be all even indices in the set $\{i_1, \cdots, i_n\}$ and $j_1 < \cdots < j_k$, $k \geq 1$. (If there are no even indices in this set, we take $k = 0$ in the following expressions).*

Then, the distance $d(u,v)$ and the routing between u and v are as follows:

- *If $s_{i_{n+1}} \in S_e$ then $r^{k+1}(v) = u$, $d(u,v) = k+1$ and the routing is given by the following sequence of vertices*

$$v, r(v), r^2(v), \ldots, r^k(v), r^{k+1}(v) = u.$$

- *If $s_{i_{n+1}} \in S_o$ then $r^{k+1}(v) = r(u)$, $d(u,v) = k+2$ and the routing is given by the following sequence of vertices*

$$v, r(v), r^2(v), \ldots, r^k(v), r^{k+1}(v) = r(u), u.$$

Proof. From vertex v, and using the function r, first we find the sequence $xs_{i_{j_1}}$ where $x \in S_o^*$ and $s_{i_{j_1}} \in S_e$. Using Lemma 4.4, $d(v, s_{j_1+1} \ldots s_{i_{n+1}} u) = 1$. The routing begins by this sequence of two vertices, $v, s_{j_1+1} \ldots s_{i_{n+1}} u, \ldots$ that we can also write $v, r(v), \ldots$.We repeat the protocol until we reach $xs_{i_{n+1}}u$ where $x \in S_o^*$. According to Lemma 4.4 we have to consider the following two cases:

- If $s_{i_{n+1}} \in S_e$, we add one unit to the distance and the routing is given by the following sequence of vertices

$$v,\ s_{i_{j_1+1}} \ldots s_{i_{n+1}} u,\ s_{i_{j_2+1}} \ldots s_{i_{n+1}} u,\ \ldots,\ s_{i_{j_k+1}} \ldots s_{i_{n+1}} u,\ u$$

  with $d(v,u) = 1 + k$. Note that we can write this sequence of vertices as,

$$v, r(v), r^2(v), \ldots, r^k(v), r^{k+1}(v) = u$$

- If $s_{i_{n+1}} \in S_o$, the first steps of the routing are

$$v, r(v), r^2(v), \ldots, r^k(v), r^{k+1}(v) = r(s_{i_{j_k+1}} \ldots s_{i_{n+1}} u) \ldots$$

  To produce the final part of the routing we consider these cases:
  - If $u = w_o w_e r(u)$, $w_o \in S_o^*$, $w_e \in S_e$, the routing is

$$v,\ r(v),\ r^2(v),\ \ldots,\ r^k(v),\ r^{k+1}(v) = r(u),\ u.$$

    And the distance $d(u,v) = k+2$.
  - If $u \in S_o^*$ the routing is

$$v,\ r(v),\ r^2(v),\ \ldots,\ r^k(v),\ r^{k+1}(v) = b,\ u.$$

□

Example 4.6 *If u and v are the vertices with labels $v = s_1s_3s_4s_2s_2s_3s_1s_3s_4s_2s_2s_4s_1s_3s_1$ and $u = s_2s_4s_1s_3s_1$, the routing is: $s_1s_3s_4s_2s_2s_3s_1s_3s_4s_2s_2s_4s_1s_3s_1$, $s_2s_2s_3s_1s_3s_4s_2s_2s_4s_1s_3s_1$, $s_2s_3s_1s_3s_4s_2s_2s_4s_1s_3s_1$, $s_3s_1s_3s_4s_2s_2s_4s_1s_3s_1$, $s_2s_2s_4s_1s_3s_1$, $s_2s_4s_1s_3s_1$ and $d(v,u) = 5$.*

With the results obtained so far, we can give now the routing between any two vertices of $M_d(t)$.

In what follows, we denote by $u \rightarrow v$ the (shortest) path from vertex u to vertex v, generated according to the rule of the neighbor ancestor which has the shortest label, and $u \leftrightarrow v$ will denote the shortest path between two vertices of $M_d(1)(c)$ which is obtained from Figure 4.

Theorem 4.7 *Let $u, v \in S^*$ and c is the longest common suffix of u and v. We can write $u = s_{u_1}s_{u_2}\dots s_{u_m}s_{u_{m+1}}s_{u_{m+2}}c$ and $v = s_{v_1}s_{v_2}\dots s_{v_n}s_{v_{n+1}}s_{v_{n+2}}c$. Then the routing between u and v is given by $u \rightarrow u_f \leftrightarrow v_f \leftarrow v$ where u_f and v_f can be attained applying the function r. The distance between the vertices is $d(u, v) = d(u, u_f) + d(u_f, v_f) + d(v_f, v)$.*

Proof. If c is the longest common suffix of the vertices u and v, the routing between u and v has to go through the structure $M_d(1)(c)$. We use Definition 4.2 to find the vertices u_f, v_f reachable from u and v, respectively. We have to consider several cases.

(i) $u_{m+1}, v_{n+1} \in S_e$. From vertex u we attain vertex $u_{m+2}c = r^k(u) \in M_d(1)(c)$ (for some k) and $u_f = u_{m+2}c$. In a similar way $v_f = v_{n+2}c \in M_d(1)(c)$. The distance between u_f and v_f depends on the indices u_{m+2} and v_{n+2}:

- If $u_{m+2}, v_{n+2} \in S_o$, the routing between u_f and v_f is u_f, $r(c)$, v_f and $d(u_f, v_f) = 2$.
- If $u_{m+2} \in S_o$, $v_{n+2} \in S_e$ and $v_{n+2} = u_{m+2} + 1$, u_f and v_f are neighbors and $d(u_f, v_f) = 1$.
- If $u_{m+2} \in S_o$, $v_{n+2} \in S_e$ and $v_{n+2} \neq u_{m+2} + 1$, the routing between u_f and v_f is u_f, $r(c)$, c, v_f and $d(u_f, v_f) = 3$.

(ii) $u_{m+1}, v_{n+1}, v_{n+2} \in S_o$ and $u_{m+2} \in S_e$. From vertex u we reach vertex $c = r^k(u) \in M_d(1)(c)$ (for some k) and $u_f = c$. From vertex v we reach vertex $r(c) \in M_d(1)(c)$, $v_f = c$ and $d(u_f, v_f) = 1$.

(iii) $u_{m+1}u_{m+2} \in S_oS_e$ and $v_{n+1}v_{n+2} \in S_eS_o$. From vertex u we reach vertex $c = r^k(u) \in M_d(1)(c)$ (for some k) and $u_f = c$. From vertex v we reach vertex $v_{n+2}c \in M_d(1)(c)$, $v_f = v_{n+2}c$ and $d(u_f, v_f) = 2$.

Other cases are studied similarly.

Lemma 4.5, produces the routings $u \rightarrow u_f$ and $v \rightarrow v_f$ and gives the distances $d(u, u_f)$ and $d(v, v_f)$. □

5. Conclusion

We have provided a labeling and produced a routing algorithm for a family of graphs which are planar, modular, have a disassortative degree hierarchy and are small-world and scale-free. Another relevant characteristic of the graphs is their zero clustering. This combination of a low clustering coefficient, modularity, and small-world scale-free properties can be seen in real complex systems, like some social and technical networks and those related to several biological systems (metabolic networks, protein interactome, etc) [5, 8], which are also disassortative.

Finally, note that the planar property and the deterministic character of the family, is in contrast with more usual probabilistic approaches, and would make easy the study of other network characteristics and parameters and the development of novel network algorithms (communication, hub location, etc.) which then could be extended to real-life complex systems.

Acknowledgements

We thank the anonymous referee who suggested the notation used in Sections 2 and 3, which helped to simplify most of the proofs.

This research is supported by the Ministerio de Ciencia e Innovación, Spain, and the European Regional Development Fund under project MTM2008-06620-C03-01 and partially by the Catalan Research Council under grant 2009SGR1387.

[1] A. Clauset, M. E. J. Newman, C. Moore, Finding community structure in very large networks, Phys. Rev. E 70 (2004) 066111

[2] F. Comellas, A. Miralles, Modeling complex networks with self-similar outerplanar unclustered graphs. Phys. A 388 (2009) 2227-2233.

[3] F. Comellas, A. Miralles, Vertex labeling and routing in self-similar outerplanar unclustered graphs modeling complex networks. J. Phys. A: Math. Theor. 42 (2009) 425001.

[4] R. Diestel, Graph Theory, 3rd. edition, Springer-Verlag, Heidelberg, Germany, 2005.

[5] R. Ferrer i Cancho, C. Janssen, R.V. Solé, Topology of technology graphs: Small world patterns in electronic circuits, Phys. Rev. E 64 (2001) 046119.

[6] S. Fortunato and M. Barthélemy. Resolution limit in community detection. *Proc. Natl. Acad. Sci. USA* 104:36–41 (2007).

[7] A. Miralles, F. Comellas, L. Chen, Z. Zhang, Planar unclustered graphs to model technological and biological networks. Phys. A 389 (2010) 1955-1964.

[8] M. E. J. Newman, The structure and function of complex networks, SIAM Review 45 (2003) 167–256.

[9] M. E. J. Newman and M. Girvan, Finding and evaluating community structure in networks. *Phys. Rev. E* 69:026113 (2004).