Computing the average path length and a label-based routing in a small-world graph
Philippe Giabbanelli, Dorian Mazauric, Stéphane Pérennes

To cite this version:
Philippe Giabbanelli, Dorian Mazauric, Stéphane Pérennes. Computing the average path length and a label-based routing in a small-world graph. 12èmes Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel), 2010, Belle Dune, France. pp.TBA. inria-00472215

HAL Id: inria-00472215
https://inria.hal.science/inria-00472215
Submitted on 10 Apr 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Computing the average path length and a label-based routing in a small-world graph†

Philippe J. Giabbanelli, Dorian Mazauric, and Stéphane Pérennes

Mascotte, INRIA, I3S(CNRS,UNS), Sophia Antipolis, France

1 Introduction

Numerous networks such as social networks were shown to share properties, including a power-law degree distribution, which is known as the scale-free property. It was shown that when an individual joins a population by befriending the most social individuals, then the scale-free property naturally emerges. Furthermore, if one friend introduces the newcomer to one of his friends, then the small-world property is also obtained [3]: the average distance between individuals is small, and there is a high clustering coefficient (i.e., likeliness that if \( a \) is friend with \( b \) and \( b \) is friend with \( c \) then \( a \) is also friend with \( c \)). In Section 2, we recall the definition from [6] of a model based on these principles, and we propose a labelling scheme. We use the labels to enumerate all edges of the model and, in Section 3, to design a decentralized routing algorithm. To show that their model was small-world, the authors of [6] chose to study the diameter rather than the average path length. In Section 4, we provide an analytical expression of the average path length, using the recursive structure of the graph. Our expression proves that the graph is precisely small-world (i.e., its average path length grows logarithmically in the network’s size), since alternatives include ultra-small. Analytical expressions for the average path length have shown to be of particular interest [4], thus a promising future work consists of adapting our proof to other recursive graphs such as [5].

2 Definition and labelling

The undirected graph defined by Zhang, Rong and Guo [6] relies on an iterative process. We denote by \( ZRG_0 \) the empty graph, \( ZRG_1 \) is the first step corresponding to a cycle of three nodes, and “\( ZRG_t \) is obtained by \( ZRG_{t-1} \) by adding for each edge created at step \( t-1 \) a new node and attaching it to both end nodes of the edge” [6]. For example, Figure 1(a) represents \( ZRG_4 \). The design of a labelling scheme that allows decentralized routing was investigated for a similar undirected graph \( M_t \) [2]. The model starts with \( M_0 \) being the empty graph, and \( M_1 \) being a pair of nodes \( s_0,s_1 \) connected by an active edge. At time \( t = 2 \), two nodes \( a \) and \( b \) are added as a path \( s_0,a,b,s_1 \); the edge \( \{s_0,s_1\} \) that was previously active becomes inactive, while the edges \( \{s_0,a\} \) and \( \{b,s_1\} \) become active. Thus, this model consists of adding a path for each active edge and making the two end edges of the new paths active. Formally, at time \( t \), a path \( u,i,j,v \) is created for all active edges \( \{u,v\} \). Then, \( \{u,v\} \) becomes inactive while \( \{u,i\} \) and \( \{j,v\} \) are active. The

†Research support by région PACA, ANR AGAPE, and ANR DIMAGREEN.
result for \( M_4 \) is illustrated in Figure 1(b). Both \( ZRG_t \) and \( M_t \) are defined by a recursive addition of patterns, but they exhibit very different properties. Firstly, the clustering coefficient of \( M_t \) is 0, whereas the value is asymptotically 0.69 for \( ZRG_t \), which is deemed high. Indeed, the clustering coefficient relies solely on triangles (i.e., cycles of length 3) and \( M_t \) is triangle-free. Secondly, the labelling scheme of \( M_t \) cannot be used for \( ZRG_t \). In \( M_t \), for a given edge with two end nodes labelled \( L(u) \) and \( L(v) \), the two nodes created at time \( t + 1 \) are labelled 0\( L(u) \) and 1\( L(u) \) respectively. By adding one node instead of two, labels are no longer unique: if \( L(u) = 0 \) and \( L(v) = 1 \) at time \( t - 1 \) then we add a node labelled 00 connecting them at time \( t \), and at time \( t + 1 \) we will add a node between 00 and 00 which will also be labelled 00.

Our labelling scheme for \( ZRG_t \) works as follows. At time \( t = 1 \), we label the vertices in the initial cycle 0, 1, and 2, and we define an orientation of the graph 0 \( \rightarrow \) 1 \( \rightarrow \) 2. At time \( t = 2 \), the edge from 0 to 1 is labelled \( L(u)L(v) \). At time \( t \), for a given node \( u \) created at time \( t - 1 \) with label \( L(u) \), we label by 0\( L(u) \) the node created for the incoming edge and 1\( L(u) \) the node created for the outgoing edge. The result is illustrated in Figure 1(a). Similarly to the proof in [2], Theorem 1 establishes that this scheme guarantees distinct labels.

Overall, a node added at time \( t \) will have two incoming and outgoing edges of the node labelled \( u \) for \( t \). For each node \( u \) of \( t - 1 \), the added nodes are labelled 0\( L(u) \) and 1\( L(u) \), which are distinct labels. Since all labels in \( t - 1 \) are distinct, this holds for \( t \).

We observe that the node \( u \) labelled 0 is connected to all nodes whose labels start with 01 and is followed only by 0s. Indeed, at time \( t = 1 \), we create a node labelled 01 for the outgoing edge of 0. At time \( t = 2 \), the outgoing edge from 0 is the incoming edge of 01 thus the new node is labelled 011, and this follows by induction. Using the same reasoning, Table 1 lists all possible edges of the graph. The table uses the following short notation: \( \bar{X} \) and \( X \) denote a (possibly empty) sequence of 0s and 1s respectively, and \( |X| \) denotes the number of symbols in a label \( X \).

### 3 Decentralized routing

Let us consider that we want to find the shortest path from a node \( u \) to a node \( v \). First, we present a few cases to illustrate the main principles of our routing algorithm. (a) If \( L(u) = X \) and \( L(v) = Y \), with \( y \) a single letter, then we say that \( u \) and \( v \) are cousins. Informally, they have been created at the same step for the incoming and outgoing edges of the node labelled \( X \). Thus, the shortest path consists of going to \( X \) and then

* The graph \( ZRG_t \) is undirected, and the orientation is defined only for the labelling and the proof of the average distance.
Computing the average path length and a label-based routing in a small-world graph

<table>
<thead>
<tr>
<th>Label of $u$</th>
<th>Label of $v$</th>
<th>Table 1: Edges of $ZRG$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0, 1, 2</td>
<td>0, 1, 2</td>
<td>$xy$</td>
</tr>
<tr>
<td>$x, y$</td>
<td>$x, y$</td>
<td>$01g, 20h$</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>$12g, 01h$</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>$20g, 12h$</td>
</tr>
<tr>
<td>$X,</td>
<td>X</td>
<td>&gt; 1$</td>
</tr>
</tbody>
</table>

**Algorithm 1 Move($u, v$)**

1. IF $u = v$, stop.
2. IF $u$ is neighbor of $v$, Move($u, u$)
3. IF $L(u) = xy$ and $L(v) = ho$, Move($u, v$)
4. IF $L(u) = Xy$ and $L(v) = Xh$, Move($L(Xu), L(Xh)$)
5. Move(oldAncestor($u$), oldAncestor($v$))


directly to the target. Note that cousins do not exist in the graph $M_t$; a node chooses between sending the message to its ‘brother’ or its parent, and always chooses the latter unless the target is the former. (b) If the nodes are not cousins, then we move simultaneously from $u$ and $v$ to their oldest ancestors, which defined as the node with the shortest label and is obtained via oldAncestor($u$). We check if the nodes on which we moved are cousins, in which case we apply case (a), or we keep on moving until we reach either a node of $ZRG_0$ or a node of $ZRG_1$. (c) If both $u$ and $v$ belong to nodes of $ZRG_1$, then they are direct neighbors and we stop. If both belong to nodes of $ZRG_2$, then they move to the node having as label their common letter. For example, if $L(u) = 12$ and $L(v) = 01$ then they move on 1. Otherwise, w.l.o.g. we consider that $u$ is on a node of $ZRG_2$ and $v$ is on a node of $ZRG_1$. The node $u$ has a label $xy$ and we move to $y$, which is guaranteed to be a neighbor of $v$. The process is given by Algorithm 1 in which $L(X)$ refers to the node with label $X$.

4 Average path length

The average path length of a connected graph $G$ is defined by $\ell(G) = \frac{\sum_{u \in V(G)} \sum_{v \in V(G)} d(u, v)}{|V(G)|(|V(G)| - 1)}$, where $d(u, v)$ is the length of a shortest path between $u$ and $v$. To compute $\ell(ZRG_i)$, we define an auxiliary function $g(t)$ that denotes the number of nodes added at step $t \geq 1$.

**Lemma 1.** $g(t) = \frac{3}{2}2^t, t \geq 2,$ and $g(1) = 3$

**Proof.** We prove by induction that $g(t) = 2g(t-1), t \geq 2$, and the recurrence is straightforwardly solved into $g(t) = \frac{3}{2}2^t$. We observe that $g(1) = g(2) = 3$ and $g(3) = 6$. We assume that $g(t) = 2g(t-1)$ holds up to $t$. By construction of $ZRG_{t+1}$, one node at time $t + 1$ is added for all edges created at time $t$, and there are $2g(t)$ such edges since two were created for each of the $g(t)$ nodes.

**Theorem 2.** $\ell(ZRG_i) = \frac{-3^{i+1}+2^t+1}{3.2^{t-1}-1}$

**Proof.** For the sake of clarity, let $\ell(ZRG_i) = f(t)/(|V(ZRG_i)|.(|V(ZRG_i)| - 1))$. First, we prove that $f(t + 1) = 4f(t) + 3g(t + 1)^2 - g(t + 1) - g(t + 1)h(t + 1); t \geq 2,$ where $g(t)$ is the number of nodes added at time $t \geq 2$ and $h(t) = \frac{1}{2} + 1$ is a correcting factor that will be explained in the proof. Let $V(ZRG_i) = \{v_1, ..., v_n\}$ denote the nodes in $ZRG_i$, and let $A_{i+1} = \{a_1, ..., a_n\}$ denote the nodes added to $V(ZRG_i)$ at time $t + 1$. We associate each new node $a_i$ to the node $v_i$, accessed by the outgoing edge $(a_i, v_i) \in E(ZRG_{i+1})$. Our proof is in three steps: (a) we approximate the shortest path distance between nodes of $A_{i+1}$ and nodes of $V(ZRG_i)$, (b) we approximate the shortest path distance between nodes of $A_{i+1}$, and (c) we correct the approximations by introducing a correcting factor.

(a) Let's consider that we virtually ‘collapse’ each $a_i$ with its associated $v_i$ : all routings involving an $a_i$ are assumed to go through its $v_i$ and the edge $(a_i, v_i)$ is free. Instead of having one path for each pair $(v_i, v_j)$, we now have four paths between the pairs $(v_i, v_j)$, $(v_i, a_j)$, $(a_i, v_j)$ and $(a_i, a_j)$. Thus, $f(t + 1) \approx 4f(t)$. We now have to consider the edge connecting each $a_i$ to its associated $v_i$. There are $g(t + 1)$ nodes in $A_{i+1}$ and also $g(t + 1)$ nodes in $V(G_i)$ (see Lemma 1) : since all of the formers need to send a message to all of the latters, and the other way around, the extra factor is $2g(t + 1)^2$. Thus, $f(t + 1) \approx 4f(t) + 2g(t + 1)^2$.

(b) Each of the new nodes in $A_{i+1}$ wants to reach all nodes in $A_{i+1}$ but itself. Thus, a new node reaches $g(t + 1) - 1$ new nodes, hence $(g(t + 1)/(g(t + 1) - 1))$ for all nodes. By combining with (a), we have $f(t + 1) \approx 4f(t) + 3g(t + 1)^2 - g(t + 1)$.

(c) The approximations considered so far that the new node $a_i$ was always routing through its associated node $v_i$. However, some routes are shorter through the alternative neighbor $v_{i-1}$ rather than through $v_i$ (see Figure 2(a)). There are on average $h(t)/2$ corrections for outgoing routes and as many for incoming routes. For the sake of clarity, the following argument only discusses outgoing requests from $a_i$ but, due
to symmetry of requests, we use the notation $h(t)$ for the average number of requests from an $a_i$ that are best routed through the alternative neighbor rather than the associated one. Thus, we want to prove that $f(t + 1)$ must be corrected by a factor $g(t + 1)h(t + 1)$ where $h(t + 1) = 2h(t - 1), t \geq 3, h(3) = 3$. We assume that we need to correct $g(t)h(t)$ nodes at time $t$ and prove it by induction. First, we have $f(3) = 4f(2) + 3g(3)^2 - g(3) - g(3)h(3) = 4.42 + 3.6^2 - 6 - 6.3 = 252$, which is verified. At time $t + 1$, we consider a node $a_i$ added at time $t + 1$. By induction, its associated vertex $v_i$ was already routing messages toward $h(t)$ targets on average through $v_{i-1}$ rather than its associated $a_i$. Thus, $a_i$ will also use $v_{i-1}$ to route through the same $h(t)$ targets. Since these targets were created at time $t$, they also have new nodes and $a_i$ will use $v_{i-1}$ to route to $h(t) - 1$ of these new nodes; the last one is not included since one of its endpoints was routed through the associated node $a_i$.

The average distance $\ell$ in a graph with $N$ nodes is said to be small when $\ell$ is proportional to $\ln(N)$ and ultrasmall when $\ell$ is proportional to $\ln(\ln(N))$ [1]. We observe that $\lim_{t \to \infty} \frac{\ln(\ln(|V(ZRG_t)|))}{\ln(|V(ZRG_t)|)} \approx 0$. Thus, the average size is almost exactly $\ln(|V(G)|)$ for large $t$. This is deemed small, and can be expected since the graph is biconnected outerplanar and the weak dual of such a graph is a tree. Since the size of the graph is exponential in $t$, it is important that the graphs obtained for small values of $t$ have a similar ratio, which is confirmed by the behaviour illustrated in Figure 2(c).

The proof of Theorem 2 could be adapted to extensions of ZRG, such as [5] in which a node is created with probability $p$. An approximation and an upper bound were given for the average path length, thus it would be of particular interest to extend our method to this probabilistic case.

Références