Research Master's Degree Internship Report

Degree-Based Routing in Small World Networks

Author :
Olivier Ruas

Supervisors :
George Giakkoupis
Anne-Marie Kermarrec
Team : ASAP

Abstract

In my internship we have studied decentralized routing in small-world networks. Our model was an variant of Kleinberg's model : we used a 1-dimensional grid in which the number of long-range contacts is not fixed and follow a power law probability distribution, the edges are considered bidirectional and drawn independently following a harmonic distribution as in Fraigniaud and Giakkoupis work ([12]). We increase the number of neighbors to a expected $O(\ln n)$ number of neighbors instead of a constant expectation as in the previous work. Also we add knowledge to every node : they know the neighbors of their neighbors. This model was motivated by Kleinberg in [5] and by the work of Manku, Naor and Wieder ([7]) which have shown that this improves the efficiency of routing to an optimal expected value in their model. We have shown that the expected number of steps for routing a message in this model using a greedy algorithm, with respect to the knowledge the nodes have, is $O(1)$ which is indeed optimal.

Keywords. Small-world networks, Random graphs, Probabilities, Routing, Distributed algorithm.

Contents

1 Introduction

1.1 State of the art

The study of complex networks is interesting since they are more and more used in a wide range of fields. In particular social networks are really difficult to handle since the definition is really informal and can be applied to a lot of different models of networks. We will focus on small-world networks which are networks in which there are short paths between almost all pairs of nodes. We can model such networks by the mathematical random graphs which are well studied in mathematics, using discrete probabilities and combinatronics. We take a node space and we randomly create links between the nodes. The difficult part is to obtain, with this method, graphs with the same properties as small-world networks. In order to do so we will classify the random graphs by models depending on how they are generated and study the whole model at once.

For each model we want to know how efficient the routing of a message from one node to an other is. This is an interesting problem studied a lot in every model of networks because most of the protocols upon networks are done thanks to message passing, then the efficiency of routing will have a huge impact on the efficiency of these protocols. To measure the efficiency we will use the number of hops, or nodes visited by the decentralized algorithm during its execution to find the path from the source to the target. The results depend on the model and the algorithm but we will focus on greedy decentralized algorithms. The efficiency of the greedy algorithm is known for a large set of models but we end up with results which make us think that we could have better only with few changes.

1.1.1 The small-world phenomenon

Everyone has already experienced the fact that, while meeting someone far from home, discovering that we share an acquaintance in common. Further more, we usually speak about the "six-degree" separation between two people.

The latter common expression comes from an experiment made by Stanley Milgram in the 60's ([8]). To point out the existence of short paths, random people in Omaha (Nebraska) were selected to send a message to the same target in Boston (Massachusetts). They were given few pieces of information about the target as his name, location and occupation. The participants only have the right to forward the message to someone they know. If the holder of a message does not know the target, he will forward it to the person he knows who is more likely to know the target. The receiver will have to do the same until the message reaches someone knowing the target who, finally, will forward to the target. By only forwarding the message to acquaintances of them, chains of acquaintances between the sources and the target are highlighted (everyone receiving the message had to write his name on the message). The results of the experiments shown that the length of the chains were from two to ten with a median of five intermediate persons. It was the first time such phenomenon was highlighted and at this time the results were quite surprising.

Nevertheless, we have to be careful with the results, since only 44 out of the 160 letters reached the target, and there was an unique target which seemed to know a lot of people (he was not randomly selected). Still we can wonder what we can learn from this experiment; by seeing every person as a node and the acquaintanceship as an edge, we obtain an interesting property about the resulting network : not only there exists short paths between every pair of nodes but nodes can construct them only by knowing local information. If we model people by nodes and the relation of friendship by edges between these nodes, this means that a greedy algorithm should be efficient to find short paths in the network we got. Such a network (i.e.

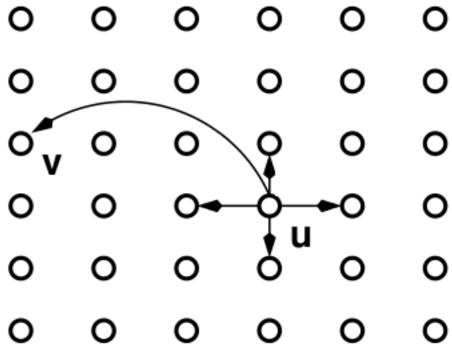


Figure 1: Kleinberg's network in dimention 2, n = 6, showing the 2 close contacts and the long-range contact of u.

in which there are short paths between almost every pair of nodes) is called a "small-world" network.

The study of these networks are useful since they can model several real networks: the collaboration graph of film actors, the power-grid of the western United States or the neural network of the worm Caenorhabditis elegans exhibit the small-world property, for example ([13],[14],[15],[21], [23],[25]).

1.1.2 Kleinberg's model

Modeling a small-world network is not that easy; just taking a graph with random neighbors for every node was not efficient because such a network does not grasp the property of social network which is that most of the friends of a person are friends together. One of the first model was done by Watts and Strogatz ([21]) : they took n nodes spaced uniformly on a ring and, using the natural L_1 distance associated, every node was linked to its k closest neighbors (with k a small constant) and to a fixed number of random nodes which were selected uniformly at random across the network. This network was grasping the property that the common friends of two nodes are likely to be themselves friends together but without being too clustered so it has a low diameter. Nevertheless, although the short paths exist, Kleinberg ([5], [22]) have shown that in such a network there is no decentralized algorithm capable of constructing paths of small expected length.

So Jon Kleinberg in ([5]) defined a grid-based model holding the small-world property : n^2 nodes are displayed on the two dimensional $n \times n$ grid graph, and every node has an edge with the 4 closer nodes (called local contacts) and one other randomly picked node (the long-range contact). See picture 1 for and example (made from a picture in [5]). The main difference with the previous model is that there is a directed edge from u to v with a probability proportional to $d(u,v)^{-r}$ (the previous model is thus a particular case of this one, with $r = 0$ and $k = 1$) where $d(u,v)$ is the L_1 distance between u and v over the network and r (called the clustering exponent) a real.

$$d(u,v) = |u_x - v_x| + |u_y - v_y| \text{ with } u = (u_x, u_y) \text{ and } v = (v_x, v_y)$$

As the nodes just know the location of their acquaintances and the target, without knowing anything else about the network, the greedy algorithm works as follows : you simply forward your message to your neighbor which is the closest to the target and this node will have to do the same until the message reaches the target. It has been shown (in [5]) that this greedy routing algorithm, which is decentralized, reaches the target in:

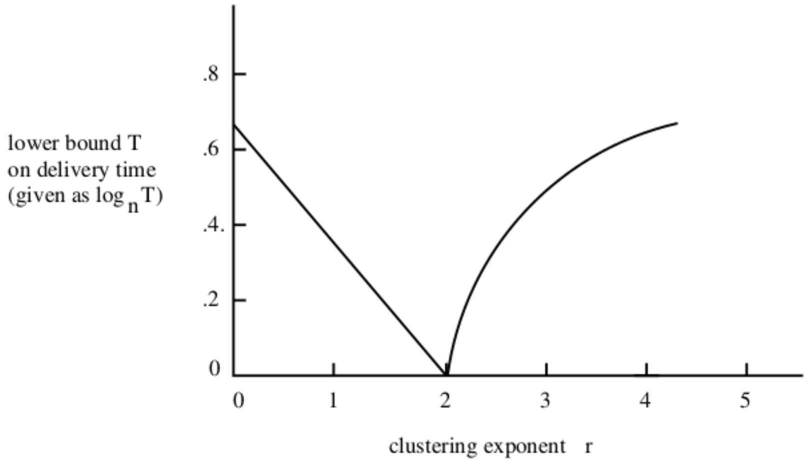


Figure 2: Influence of the clustering exponent on the efficiency.

- $\Omega(n^{\frac{2-r}{3}})$ hops in average for $0 \leq r < 2$
- $\Omega(n^{\frac{r-2}{r-1}})$ for $2 < r$
- $O(\log^2 n)$ for $r = 2$

We can see r as an expression of the correlation between the local structure and long-range connections. The figure 2 (extracted from [5]) show that 2 is a critical value, for which the long-range contacts are uniformly distributed over distance scales (the probability for the long-range contact to be at a distance between 2^i and 2^{i-1} is about the same for $i \in [0.. \log n]$).

Then when r is close to 2, you have a good chance to halve the distance between the message and the target at each forwarding, no matter how far away the message is. If the distribution is too homogenous (i.e. r is below 2, as in the first model), nodes can not find shorts path although they exist, and if r is above 2, the network is too clustered.

1.1.3 Extensions of the model

The above results extend to d-dimensional grid networks ([5]), and the results are quite similar: the decentralized greedy algorithm can find paths of length polynomial in $\log n$ if and only if $r = d$. This seems normal considering the previous talk, below d the network is too homogenous, above it is too clustered. For $r = d$ we obtain $O(\log^2(n))$ hops in average. For all these models, the variant with a constant number of long-range contacts for each node has been studied and in this model the greedy algorithm have the same asymptotic efficiency.

An other idea was to have a variable number of long-range contacts for every node, it is relevant since in real social networks nodes are very different and do not have the same degree. Fraignaud and Giakkoupis ([2]) have studied such a network : in the d-dimensional grid network, which can be represented by $[0..n-1]^d$, every node has its $2d$ closest nodes as close contacts and the number of long-range contacts is randomly chosen, the probability of having k long-range contacts is proportional to $\frac{1}{k^\alpha}$ and the expected value is 2, with $0 \leq \alpha$. Then the directed edges are chosen with the same method as before. This model and the power-law distribution are motivated by the fact that the degree in some real social networks seem to follow a power-law with an exponent between 2 and 3 ([13],[14],[15]).
With this model the greedy algorithm has an expected delivery time of $\Omega(\log^2(n))$. By making

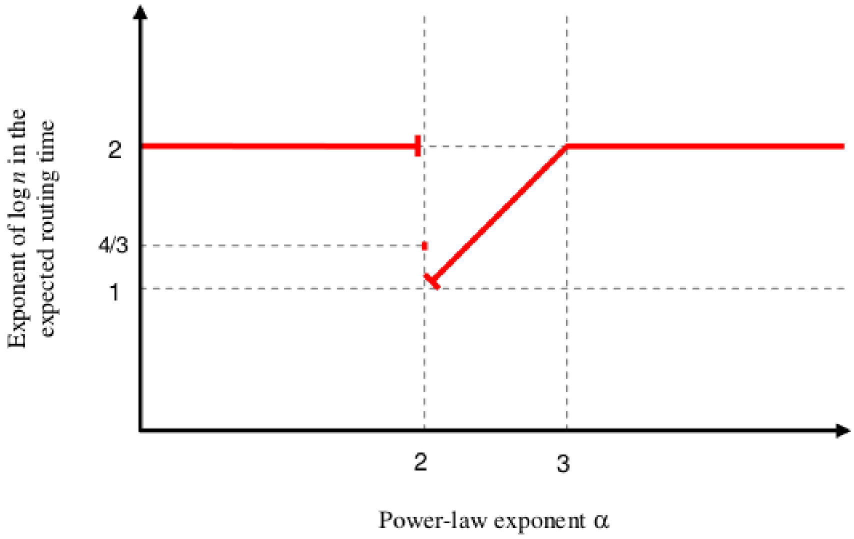


Figure 3: Results of greedy algorithm in undirected graph with variable numbers of neighbors

the edges undirected (for each directed edge (u, v), we add the edge (v, u)), the expected delivery time is improved :

- $\tilde{\Theta}(\ln^{\frac{4}{3}} n)$, if $\alpha = 2$
- $\tilde{\Theta}(\ln^{\alpha-1} n)$, if $2 < \alpha < 3$
- $\tilde{\Theta}(\ln^2 n)$, else

As we can see in Figure 3 (extracted from [2]), by adjusting the value of α, we can get as close as we want of $\ln n$ but we can't reach it.

Until now, the efficiency of the greedy routing algorithm was coming from the homophily of the network : the tendency of individuals to associate with individual similar to them. It is a property we find in real life (you are more likely to know someone in your city that someone living in an other country) and represented in Kleinberg's network by the close contacts and the harmonic distribution which favors bonds between nodes no to far from each other. By making the number of edges of a node a variable, we add degree disparity which allows some nodes to act like hubs between distant parts of the network.

1.1.4 Using knowledge about neighborhood

The greedy algorithm is based on the fact that you only know the distance of your neighbors from the target and you forward the message to the one who minimizes this distance. Then we can wonder what happen if we get more information about our neighbors, as their own neighborhood. Dodds, Roby and Watts ([10]) made a study similar to Milgram's in which they asked each participant to say the reason why they have chosen the person they will forward the message to. It appears that despite the location is the first reason for forwarding, the two first steps about 25% of the forwarding were due to partial knowledge about the neighborhood (it was known to have traveled near the target's location or has family from this place). Knowing this makes relevant to allow nodes to have more knowledge about the network.

First we consider that nodes know the neighbors of their neighbors. Manku, Naor and Wieder studied ([7]) this assumption over Kleinberg's original model where the greedy algorithm forward not to the node which will minimize the distance with the target but to the one who has a neighbor which minimizes it. We call this method of looking one step further

1-lookahead.
Since in a network with k out-going links per nodes the average length of short path is $\Omega(\log n/\log k)$ ([11]) so it should be possible to achieve routing times that match the network diameter of $O(\log n/\log\log n)$ with $k = O(\log n)$. Manku, Naor and Wieder ([7]) have shown that the 1-lookahead policy have really good results and match this $O(\log n/\log\log n)$ bound.

Since one step lookahead offers paths with an optimal length (because they asymptotically match the network diameter), additional lookahead can not improve the asymptotic length of the paths.

Both of the previous work are interesting and improve the efficiency of the initial model of Kleinberg but we can wonder how efficient the routing should be if we combine the both ideas.

1.2 Our contribution

During my internship we have considered a model based on Fraigniaud and Giakkoupis work ([12]) in which we increase the knowledge of each node. This model is a 1-dimension grid in which each node has a variable number of long-range contacts chosen independently at random from the 1-harmonic distribution. The probability of having k long-range contact is proportional to $1/k^2$ with a $O(\ln n)$ expectation. We treat the long-range contacts as bidirectional and assume that every node has at least one long-range contact. We study the complexity of the greedy algorithm which is different from the one studied by Kleinberg in ([5]) and Fraigniaud and Giakkoupis in [12] since we add knowledge to every node: every node knows its neighbors and the neighbors of its neighbors. So we will studied the same algorithm used in [7] by Manku, Naor and Wieder : a node will forward not to the node which will minimize the distance with the target but to the one who has a neighbor which minimizes it.

In this model and with such protocol we found that the expected number of steps to reach the target from the source is $O(1)$ which is optimal.

1.3 Related work

C. Martel and V. Nguyen have studied networks similar to the grid graph and computed their diameter ([16], [17], [26]) but not only the grid-based networks are small-world networks, Kleinberg ([18]) has studied routing in several networks, including a b-tree base network in which the distance between two nodes is based on the height of their lower common ancestor. As the grid was a simple abstraction of geography, this hierarchy is an acceptable abstraction of occupations, hobbies etc... He also proposed a model which generalized both of the previous ones (the grid-based model and the hierarchical model) by grouping all the nodes by sets, each node can be in multiple sets, and the distance between two nodes is based on the size of the smallest set containing both of them.
In all these models, with good distribution of long-range contacts, greedy rooting, with respect to the distance and the topology of the network, reaches the target in polylogarithmic expected number of steps.

D. Liben-Nowell, J.Novak, R. Kumar, P. Raghavan, and A. Tomkins have studied ([25]) a lot of real networks using geographic routing which appear to be small-world networks. They perform in such a network (the LiveJournal social network) the Milgram's experiment. The interest was to get rid of human participation which can leads to add some noise to the results (e.g. the people who have drop the letters in Milgram's experiment, not because they did not know anyone to forward it to but because they did not wanted to) in order to know if individuals can find short paths only by using geographic information.

In [23], S. Lattanzi, A. Panconesi, and D. Sivakumar have modeled the social networks of co-authorships in computer science. Two persons are friends if there are co-authors and instead of using the geographic distance they used a "interest" distance to decide to which friend you should forward to. They obtained a social network of roughly 15 million individuals and simulate on it the Milgram's experiment to show that they network was indeed a small-world network though the unusual distance used.

There are also a lot of work done to find lower-bounds for greedy routing algorithms in such networks, as [27] by M. Dietzfelbinger, J. Rowe, I. Wegener, and P. Woefel in which they have shown that, with unidirectional links, the expected number of steps for routing is $\Omega(\ln^2 n)$ in the ring-based network, no matter the distribution used to drawn the number of long-range contacts of the nodes, as long as the expected number of long-range contacts is a constant (our work is not affected by this since we have a $O(\ln n)$ expected number of long-range contacts and the edges and bidirectional). In [1], Dietzfelbinger and Woelfel extended this result to ring with more long-range contacts.

Manku, Naor and Wieder studied ([7]) the 1-lookahead method over several networks (percolation graph, skip graph, Chord...) and they have shown that using this method in small-world percolation networks leads to an optimal routing of $O(\frac{\log n}{\log \log n})$ steps.

Adamic and al. ([19]) have worked on efficient routing algorithm on unstructured P2P networks which have power-law link distribution like Gnutella and Freenet and in which you do not know precisely the location of the target. They have highlighted that the efficiency of the random walker (basically you just randomly forward to one of your neighbor until you find or until you have reached a chain of a given size) was relying on the existence of high degree nodes so they introduced several algorithms similar to the random walker which aims intentionally these nodes. They proved that their algorithm were sub-linear with the size of the network. N. Sarshar, P. O. Boykin, and V. P. Roychowdhury have also tried several strategies for routing in unstructured P2P networks.

Simsek and Jensen ([9]) have studied a network similar to the 1-dimensional grid with the power-law distribution but also with the poisson distribution and used an algorithm using both homophily and degree disparity to increase the efficiency of the routing. They design an algorithm which consists in forwarding the message to the node which minimizes the expected length of the path to the target. In practice, since you can not compute this value exactly they use an estimate of it: if the target is one of your neighbors then you forward to it, else your forward the message to your neighbor which has the highest probability of having one of its links ending at the target. This probability can be computed assuming that you know the precise location of the target and those of your neighbors and the degree of your neighbors. They find out that their algorithm was better than the ones using only homophily (e.g. the algorithm used in Kleingberg's network) or the ones using only degree disparity (e.g. algorithms designed in [19]) but only thanks to simulation, without any theoretical bounds.

B. Kim, C. Yoon, S. Han, and H. Jeong have defined in [29] a new notion on diameter which depends on the strategy used to forward the message and then studied it for several strategies including the one using only local information. They found that using both this strategy and with a global strategy the diameter increases logarithmically with the network size (the network used was the model of Barabási and R. Albert ([31]) which is a network with a power-law link distribution).

Until now all the algorithms used were greedy altough they were not the same, indeed you do not act the same if you have or not knowledge about your neighborhood or not. Giakkoupis and Schabanel ([4]), based on the work done in ([24]), have made an algorithm on Kleinberg network which is more complex than the greedy one but in the other hand it finds paths with optimal expected length $O(\log n)$ while visiting $O(\log^{2+\epsilon} n)$ nodes, $0 < \epsilon$ fixed, with high probability. The result is the same for the other dimensions than 2, except 1 for which they have proven that the length is $O(\log n \log \log n)$. To achieve this, they keep track with the message of all the nodes which have forwarded the message, allowing a backtrack when needed in order to find the shortest path. With this possibility, they go through a BFS tree starting at the source and they have shown that one of the leaves is closer to the target than the source with some non-null probability. They forward to this node and start again the process until it reach the target. With that algorithm, since we are forwarding the message to some nodes which will not be a part of the final path, we are not only interested in the length of the found path but also in the number of nodes "visited" by the message.

In the other hand, Fraigniaud, Gavoille and Paul ([28]) designed a non-greedy routing which improved the length of the path from $O(\ln^2 n)$ (the best we can achieve for greedy routing) to $O(\ln^{1+1/d})$ in the Kleinberg's d-dimensional grid assuming that you have $O(\ln^2 n)$ bits of topological awareness per node. They proved that the bound is tight.

2 Model

The small-world random graph we use is a 1-dimensional grid of size n. Each node is caracterized by an integer $i \in [0..n-1]$ and then the distance between two nodes i and j is $d(i,j) = i - j \ mod \ n$. Every node is linked to the two nodes which are at distance 1 from him, they are called the *local contacts*. We add to this graph, which is until now deterministic, some random directed edges to some further nodes, called *long-rang contacts* : a node has k long-range contacts with a probability proportional to $\frac{1}{k^2}$, and then the k long-range contacts are chosen independently at random with replacement following a distribution that links two nodes i and $j \neq i$ with a probability proportional to $\frac{1}{d(i,j)}$.

More precisely the power-law distribution from which the number of long-range contact of u is drawn, called C_u, is defined as follows :

$$\Pr(C_u = k) = \frac{c}{k^2}$$

where c is a normalizing constant. The average number of long-range contact is $O(\ln n)$ so it leads to :

$$E(C_u) = \sum_{k=1}^{n/2} k \cdot \frac{c}{k^2} = c \cdot (\ln(\frac{n}{2}) + \gamma + o(1))$$

which implies $c = \Theta(1)$ so we finally have $\Pr(C_u = k) = \Theta(\frac{1}{k^2})$.

For the distribution from which the long-range contacts are independently drawn is the same as in Kleinberg's initial network :

$$\Pr(u \to v) = \frac{1}{c' \cdot d(u,v)}$$

where c' is a normalizing constant. The sum of all the probabilities equals to one, so we have :

$$c' = \sum_{v \neq u} \frac{1}{d(u,v)} = \Theta(\ln n)$$

For every edge (u,v) created, u is called an in-contact of v and v an out-contact of u.

3 Protocol

To improve the efficiency of the routing, we ignore the direction of edges : if there is a directed edge from u to v, v will be able to use it to forward a message to u even if there is no edge from v to u as in the model of Fraigniaud and Giakkoupis ([12]). Due to this, the neighborhood is not only composed by the out-going contact of the nodes, but also by all the node for which

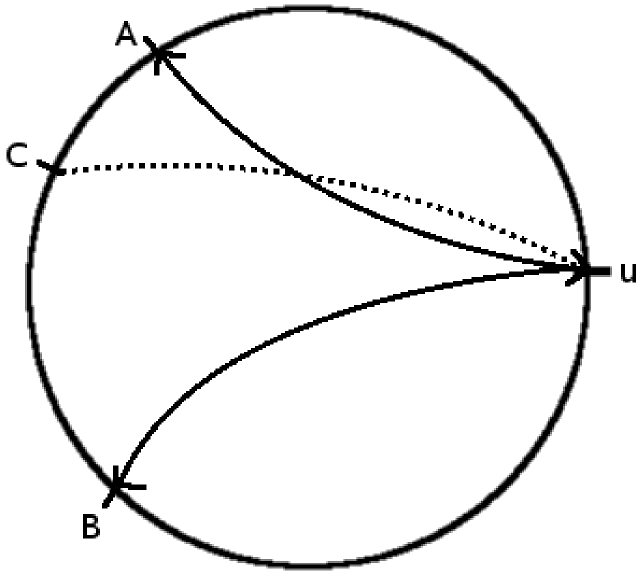


Figure 4: Distant neighborhood of u. Directed links : (u,A) (u,B) and (C,u).

you are an out-contact (e.g. if there is a directed link from u to v, u is in the neighborhood of u as we can see in figure 4, there is a link from C to u but C is a part of the neighborhood u). Neighborhood is then a symmetric notion.

Every node know the structure of the underlying network (1-dimensional grid of size n), the exact location of the target, all its neighbors, the neighbors of its neighbors (and the links between them) in the network. This knowledge allows every node to compute the distance to every node it knows to each other ant to the target.

With such knowledge, instead of forwarding to our neighbor which is the closest of the target, we look at all the neighbors of our neighbors, select the one which is the closest and forward to him via one of their common neighbor. A node will forward to one of its direct neighbors only if it is the source, indeed if it is not, thanks to the local contacts we are sure to the existence of one neighbor of one of its neighbors which is closer from the target. Thus the message is going through the network two steps by two, excepted maybe the last forwarding :

Algorithm 1 The routing protocol to forward a message to node t

```
if Current node ≠ t then
    Let N_1..N_j be the neighbors of the current node.
    if Target = N_k then
        Forward to N_k
    else
        Let N_{k_1}..N_{k_{j'}} be the neighbors of N_k ∀k ∈ [1..j].
        Select the N_{k_i} which is the closest to t.
        Forward the message to N_{k_i} via N_k.
    end if
end if
```

It is exactly the same algorithm called Neighbor-of-Neighbor (NoN) greedy routing algorithm studied by Manku, Naor and Wieder in [7]. See picture 5 to see an example of how the algorithm

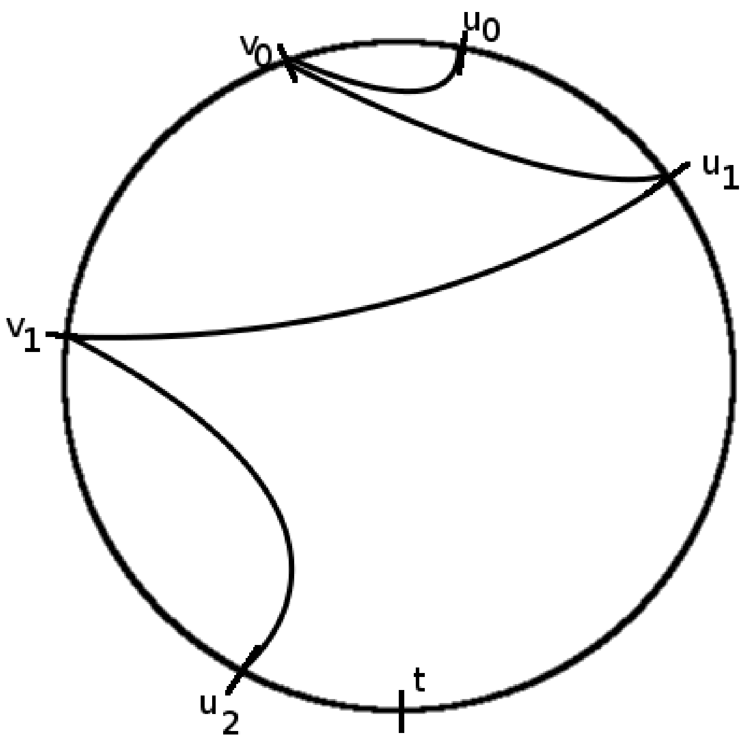


Figure 5: The first two steps of the algorithm : u_0 want to send a message to t, uses v_0 to forward it to u_1 which forward it to u_2 via v_1.

works.

4 Results

Theorem 1. *We found that the expected number of steps needed for a message to reach its target is $O(1)$ which means it is independent of the size of the network. This result is the best we can achieve for a routing algorithm.*

4.1 Notations

Due to the fact that the message is going through the network two steps by two, we have to make a distinction between two kinds of nodes : the nodes u_i which are receiving the message (or the source) and which will execute the main algorithm until it finds the target and the nodes v_i which will only forward to one of their neighbors which has already chosen by the node which has forwarded the message to v_i.
One interesting property is that on the path $u_0..u_i$ (we do not write the v_i but we must not forget that between any u_i and u_{i+1} their exists a common neighbor v_i which is linked to both), for any k we have $d(u_k, t) < d(u_{k+1}, t)$ but we can not say anything about the distance from any v_k to the target except that $v_i < u_{i+1}$ because if it was possible, u_i would have forwarded to one of the local contacts of v instead of u_{i+1}.

4.2 Intuition of the proof

The entire proof works on the fact that the edges are bidirectional : indeed, thanks to that we do not have to consider the edges going from the source but we just have to look at all the other nodes and bound the probability that one of them is linked to both the node which has

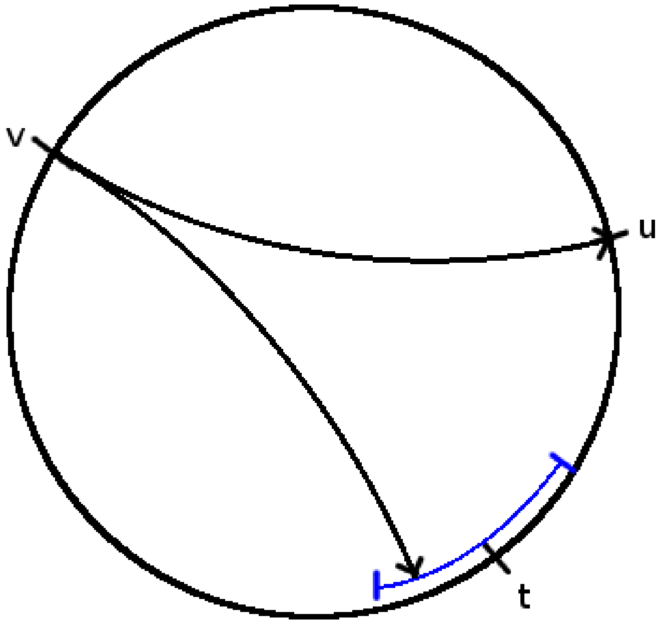


Figure 6: The node v is the kind of node we are looking for.

the message and to a ball of a given size centered on the target. For an example of such a node see figure 6 in which u is the source, t the target and the ball is represented in blue.

The efficiency comes from the fact that there exists several nodes with high-degree which will act like hub although they should be really far from both the nodes which has the target and the source. The 1-lookahead policy allows the nodes to know if they are linked to such a node, which is likely to have an edge to the targeted ball, and select among them the one which will forward the message the closest from the target.

In other words what really matters is not to which nodes the holder of the message is linked to but which nodes are linked to him. The difficulty is coming from the path: if the links were unidirectional (that is not exactly our case since, for an edge (u, v) we can use it as is was (v, u)), the path until a node does not have much influence on what will do the node since its links are independent of the links of the ones of the path, then we have a Markov's chain. But here since we are using the edges of other nodes it is very different : if a node u_0 have forwarded a message to u_1 via v it means that in the following steps the intermediate nodes must be different of v, because if u_1 use v to forward the message to u_2, it means v is link to u_0, u_1 and u_2 so u_0 should have directly forwarded the message to u_2 (see figure 7). This comes from the fact that the u_i are closer and closer from the target, so when u_0 selected the neighbors of its neighbors to which it will forward the message, it would have chose u_2, not u_1. This is not only working for consecutive steps but for more further steps in the execution of the protocol. Our main problem was to get rid of the dependency of the path in the evaluation of the probabilities while keeping interesting results.

We consider two parts during the execution of the algorithm : when the message is at a distance greater than $\sqrt{n}$ from the target and when it is not. When the source of the message is at a distance at least $\sqrt{n}$ from the source we show that the source is able, using an intermediate node, to forward directly the message to a node at a distance lower than $\sqrt{n}$ from the target with high probability. In the other case, we can lower bound the expected number of steps by $O(\ln^2 n)$ ([12]); it will be enough to reach a constant expected number of step since that case

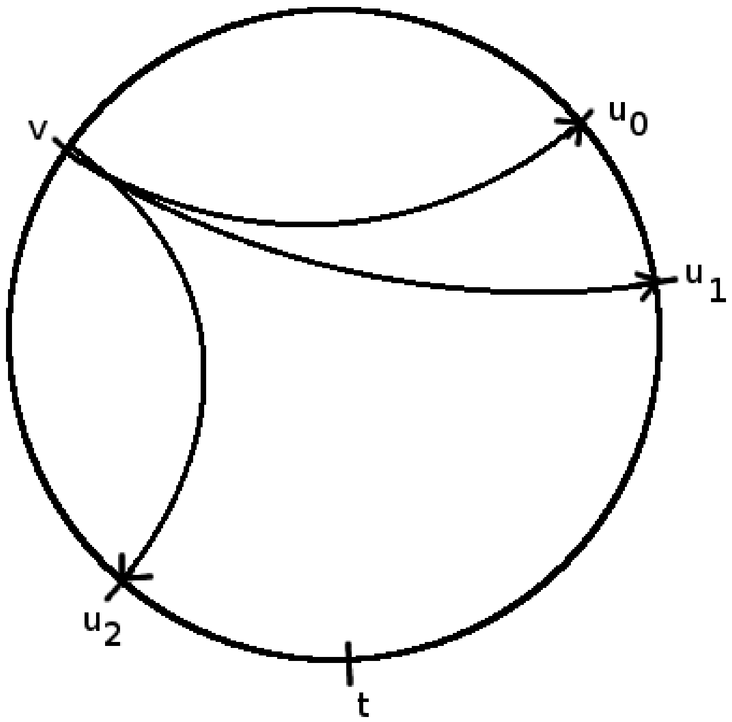


Figure 7: If v is linked to u_0, u_1 and u_2 then u_0 forwards to u_2, not to u_1

happens with really low probability.
Then, once your are close enough of the target, you can reach a node at a distance r with a probability $\frac{1}{r^{1+c}}$ for some fixed c, and then the expected number remaining is $O(\ln r)$ so in average we got a constant expected number of steps.

5 Intermediate Results

5.1 Upper-bound

Lemma 1. *Given a path $u_0, u_1, .., u_i$, the remaining steps from u_i to the target t is at most $O(\ln^2 n)$.*

Proof. Fraigniaud and Giakkoupis ([12]) have studied the routing in the same model excepted that the expected number of long-range contacts is a constant instead of $O(\ln n)$. They have shown that the expected number of steps needed to forward a message for a node u to a node t is $O(\ln^2 n)$ for the *worst* pair (u, t). Then, as having more long-range contacts, more knowledge and making the edges undirected can only improve the efficiency of the routing, $O(\ln^2 n)$ is also an upper-bound for our routing complexity. □

5.2 First step

The first result we got is that the probability that exists a node linked to both a node u and a ball centered on a node t and of radius $\sqrt{n}$ equals to $1 - \frac{1}{n^{\Omega(1)}}$, i.e. the probability that the source forward the message to a node at a distance at most $\sqrt{n}$ from the target is really high. Indeed this probability is computed assuming that no steps have been made before that is why you can only apply it from the source.

Lemma 2. *Assuming that the source u_0 is at a distance $d \geq \sqrt{n}$ from the target t, u_1 is at a distance at most $\sqrt{n}$ from t with probability $1 - \frac{1}{n^{O(1)}}$.*

$$\Pr(\exists v, v \to u_0 \land v \to B_t(n^{1/2})) = 1 - \tfrac{1}{n^{\Omega(1)}}$$

Proof. The probability of the existence of such a v is 1 minus the probability that none of the v is showing these properties :

$$\begin{aligned}
\Pr(\exists v, v \to u_0 \land v \to B_t(n^{1/2})) &= 1 - \prod_v (1 - \Pr(v \to u_0 \land v \to B_t(n^{1/2}))) \\
&\geq 1 - \prod_v e^{-\Pr(v \to u_0 \land v \to B_t(n^{1/2}))} \\
&= 1 - e^{-\sum\limits_v \Pr(v \to u_0 \land v \to B_t(n^{1/2}))} \\
&= 1 - e^{-\sum\limits_v \sum\limits_k \Pr(v \to u_0 \land v \to B_t(n^{1/2}) | C_v = k) \cdot \Pr(C_v = k)}
\end{aligned}$$

Now we need to compute this sum, to achieve this we need to introduce some notations :

- $x = d_{v,u}$
- $y = d_{v,t}$
- $d = d_{u,t}$

With these we can obtain a formula which holds in all cases :

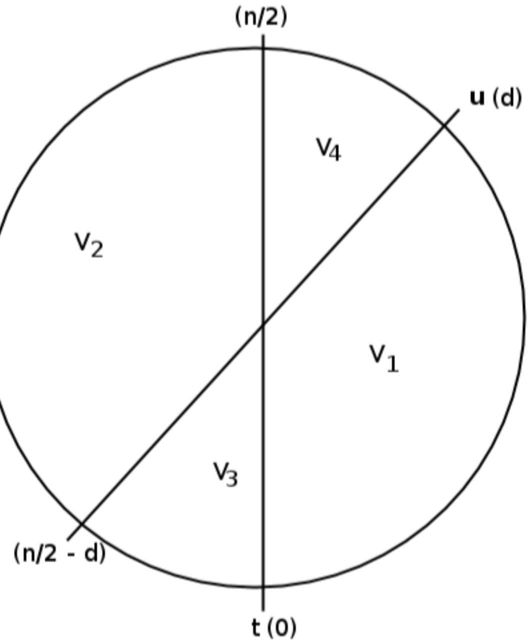
$$\begin{aligned}
&\sum_{k=2}^{k_{max}} \Pr(v \to u \land v \to B_r\{t\} \mid C_v = k) \cdot \Pr(C_v = k) \\
&\quad = \sum_{k=2}^{k_{max}} \Pr(v \to u \mid C_v = k) \cdot \Pr(v \to B_r\{t\} \mid v \to u, C_v = k) \cdot \Pr(C_v = k) \\
&\quad \geq \sum_{k=2}^{k_{max}} \Pr(v \to u \mid C_v = k) \cdot \Pr(v \to B_r\{t\} \mid C_v = k - 1) \cdot \Pr(C_v = k) \\
&\quad = \sum_{k=2}^{k_{max}} (1 - (1 - \frac{r}{y \ln n})^{(k-1)})(1 - (1 - \frac{1}{x \ln n})^k) \frac{1}{k^2}
\end{aligned}$$

We'll have the two cases:

- $x \cdot r \leq y$
- $y \leq x \cdot r$

For each of these, we have 4 cases depending on v:

1. $x + y = d$
2. $x + y = n - d$
3. $x = y + d$
4. $y = x + d$



We will only consider the case $y/r \leq x$ and $\frac{y}{r}\ln n \leq k \leq x\ln n$ and in that case we have : $(1-(1-\frac{r}{y\ln n})^{(k-1)})(1-(1-\frac{1}{x\ln n})^{k})\frac{1}{k^2} \sim \frac{k}{x\ln n}$
Then we can simplify the sum :

$$\sum_{k=\frac{y}{r}\ln n}^{x\ln n}(1-(1-\frac{r}{y\ln n})^{(k-1)})(1-(1-\frac{1}{x\ln n})^{k})\frac{1}{k^2}$$

$$\geq \sum_{k=\frac{y}{r}\ln n}^{x\ln n}\frac{k}{xk^2\ln n}$$

$$= \sum_{k=\frac{y}{r}\ln n}^{x\ln n}\frac{1}{kx\ln n}$$

$$= \Omega(\frac{\ln(\frac{rx}{y})}{x\ln n})$$

Now we will only consider the nodes in the set V_1 which is enough for the probability we want to obtain.
All the nodes v in V_1 have the property that $x+y=d$, so we get:

$\forall v \in V_1, y/r \leq x \leq d-1 \implies \frac{d}{r+1} \leq x \leq d-1$, then:

$$\sum_{v \in V} \Omega(\frac{\ln(\frac{rx}{y})}{x \log n}) \geq \sum_{v \in V_1} \Omega(\frac{\ln(\frac{rx}{y})}{x \log n})$$

$$= \sum_{i=\frac{d}{1+r}}^{d-1} \Omega(\frac{\ln(\frac{ri}{d-i})}{i \log n})$$

$$\geq \sum_{i=\frac{d}{1+r}}^{d-1} \Omega(\frac{\ln(\frac{ri}{d-(\frac{d}{1+r})})}{i \log n})$$

$$= \sum_{i=\frac{d}{1+r}}^{d-1} \Omega(\frac{\ln(\frac{(r+1)i}{d})}{i \log n})$$

$$= \Omega(\frac{\ln^2(\frac{(r+1)(d-1)}{d}) - \ln^2(\frac{(r+1)(\frac{d}{r+1})}{d})}{\log n})$$

$$= \Omega(\frac{\ln^2((r+1)(1-\frac{1}{d})) - \ln^2(1)}{\log n})$$

$$\geq \Omega(\frac{\ln^2((r+1)(1-\frac{1}{2}))}{\log n})$$

$$\sim \Omega(\frac{\ln^2(r)}{\log n})$$

Now we replace this result in the main part with $r = \sqrt{n}$:

$$\Pr(\exists v, v \to u_0 \wedge v \to B_t(n^{1/2})) = 1 - e^{-\sum_v \sum_k \Pr(v \to u_0 \wedge v \to B_t(n^{1/2}) | C_v = k) \cdot \Pr(C_v = k)}$$

$$= 1 - e^{-\Omega(\frac{\ln^2(n^{1/2})}{\log n})}$$

$$= 1 - e^{-\Omega(\ln n)}$$

$$= 1 - \frac{1}{n^{\Omega(1)}}$$

This completes the proof of that the first step reaches the ball centered on t of radius $\sqrt{n}$ with high probability. □

5.3 Second Step

Now we compute the probability for the second steps to reach a ball of radius r. The computation is very similar to the previous one except we do not consider the same v : in this case we consider some v in V_3 instead of V_1. The intuition behind the proof is that all the nodes are verifying $x = y + d$ (because we are in V_3) with a small d compare to x and y (because we are at most at a distance $\sqrt{n}$ from the target) so we have $x \sim y$ and then we have some simplifications we could not have in the previous computation.

An other difference with the previous lemma is that we are not doing the first step and then we must take into account the first step, i.e. there exists a v like to both u_0 and u_1 and, more importantly, no other v has a link to u_0 and a node closer from the target than u_1 (if not, this node would have been selected to receive the message instead of u_1).

Lemma 3. *Assuming that $d(u_1, t) \leq \sqrt{n}$ and $d(u_0, t) \geq \sqrt{n}$, the probability that the second step reaches a ball centered on t of radius r is $1 - \frac{1}{r^{O(1)}}$.*

$$\Pr(\exists v, v \to u_1 \wedge v \to B_t(r) \mid \nexists v', v' \to u_0 \wedge v' \to B_t(d(t, u_1))) = 1 - \frac{1}{r^{\Omega(1)}}$$

Proof.

$$\begin{aligned}
&\Pr(\exists v, v \to u_1 \wedge v \to B_t(r) \mid \nexists v', v' \to u_0 \wedge v' \to B_t(d(t, u_1))) \\
&\qquad = 1 - \prod_v (1 - \Pr(v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t, u_1)))) \\
&\qquad \geq 1 - e^{\sum_v \sum_k \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t, u_1)))}
\end{aligned}$$

So we need to compute this sum, with $V = \{v \mid n^{2/8} \leq d(v, t) \leq n^{3/8}\}$ we have :

$$\begin{aligned}
&\sum_v \sum_k \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t, u_1))) \\
&\qquad \geq \sum_{v \in V} \sum_{k = \frac{d \ln n}{r}}^{d \ln n} \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t, u_1)))
\end{aligned}$$

So we have to simplify $\Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t, u_1)))$ to be able to bound it because we can not compute this probability due to the fact that $v \not\to u_0 \vee v \not\to B_t(d(t, u_1))$.

$$\Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t,u_1))) =$$

$$\begin{aligned}
&= \frac{\Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \wedge (v \not\to u_0 \vee v \not\to B_t(d(t,u_1))))}{\Pr(v \not\to u_0 \vee v \not\to B_t(d(t,u_1)))} \\
&\geq \frac{\Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \wedge (v \not\to u_0 \vee v \not\to B_t(d(t,u_1))))}{1} \\
&\geq \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \wedge v \not\to u_0) \\
&= \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot \Pr(v \not\to u_0 \mid c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \\
&\geq \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot \Pr(v \not\to u_0 \mid c_v = k) \\
&= \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot (1 - \frac{1}{d(v,u_0)\ln n})^k \\
&\geq \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot (1 - \frac{1}{(\sqrt{n} - n^{3/8})\ln n})^{n^{3/8}\ln n} \\
&\sim \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot (1 - \frac{1}{\sqrt{n}\ln n})^{n^{3/8}\ln n} \\
&\geq \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot e^{-n^{-1/8}} \\
&\geq \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot e^{-\frac{1}{2}^{1/8}} \\
&\geq \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot \Omega(1)
\end{aligned}$$

So the sum becomes :

$$\sum_v \sum_k \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t,u_1)))$$

$$\begin{aligned}
&\geq \sum_{v \in V} \sum_{k=\frac{d\ln n}{r}}^{d \ln n} \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r)) \cdot \Omega(1) \\
&= \Omega(\sum_{v \in V} \sum_{k=\frac{d\ln n}{r}}^{d \ln n} \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r))) \\
&= \Omega(\sum_{v \in V} \sum_{k=\frac{d\ln n}{r}}^{d \ln n} \frac{1}{k^2} \cdot \frac{k}{d(v,t)\ln n} \cdot 1) \\
&= \Omega(\sum_{v \in V} \frac{\ln r}{d(v,t)\ln n}) \\
&= \Omega(\frac{\ln r}{\ln n} \cdot \ln(n^{1/8})) \\
&= \Omega(\ln r)
\end{aligned}$$

So we get :

$$\Pr(\exists v, v \to u_1 \wedge v \to B_t(r) \mid \nexists v', v' \to u_0 \wedge v' \to B_t(d(t,u_1)))$$
$$= 1 - \prod_v (1 - \Pr(v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t,u_1))))$$
$$\geq 1 - e^{\sum_v \sum_k \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r) \mid v \not\to u_0 \vee v \not\to B_t(d(t,u_1)))}$$
$$= 1 - e^{-\Omega(\ln r)}$$
$$= 1 - \frac{1}{r^{\Omega(1)}}$$

This completes the proof that, given the path $u_0 u_1$ with $d(u_0,t) \geq \sqrt{n}$ and $d(u_1,t) \leq \sqrt{n}$, we have $d(u_2,t) \leq r$ with probability $1 - \frac{1}{r^{O(1)}}$. □

5.4 Last step

Now we try to bound the probability that the message reaches a ball centered on the target of radius r given a path of length i. We do not need a bound really big so we can afford some simplification to make the computation easy.

Lemma 4. *The probability that exists a node v, both link to u_i and a ball of radius r centered on t, given a path $u_0..u_i$ is $\Omega(\frac{1}{i})$, in other words, given such a path of length $i+1$, the probability that the next step reaches a node at a distance lower than r from the target is $\Omega(\frac{r}{i})$.*

$$\Pr(\exists v, v \to u_i \wedge v \to B_t(r) \mid u_0 u_1..u_i) = \Omega(\tfrac{r}{i})$$

Proof.

$$\Pr(\exists v, v \to u_i \wedge v \to B_t(r) \mid u_0 u_1..u_i) = 1 - \prod_v (1 - \Pr(v \to u_i \wedge v \to B_t(r) \mid u_0 u_1..u_{i-1}))$$
$$\geq 1 - \prod_v e^{-\Pr(v \to u_i \wedge v \to B_t(r) \mid u_0 u_1..u_i)}$$
$$\geq 1 - e^{-\sum_v \Pr(v \to u_i \wedge v \to B_t(r) \mid u_0 u_1..u_i)}$$
$$\geq 1 - e^{-\sum_v \sum_k \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k \mid u_0 u_1..u_i)}$$

So as usual, we need to compute the sum, so we need to simplify $\Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k \mid u_0 u_1..u_i)$ to get rid of the dependency on the path. The fact that the path is $u_0..u_i$ means that any v we consider as a candidate to forward the message must not have links to the couples (u_j, $B_t(d(t,u_{j+1}))$) for all j because if there were such an v linked to a couple (u_j, $B_t(d(t,u_{j+1}))$), instead of forwarding to u_{j+1}, u_j would have transferred the message directly to $B_t(d(t,u_{j+1}))$ via v.

$\Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k \mid u_0 u_1..u_i)$

$$= \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k) \cdot \frac{\Pr(\bigwedge_j (v \not\to u_j \vee v \not\to B_t(d(u_{j+1}, t))) \mid v \to u_i \wedge v \to B_t(r) \wedge C_v = k)}{\Pr(\bigwedge_j (v \not\to u_j \vee v \not\to B_t(d(u_{j+1}, t))))}$$

$$\geq \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k) \cdot \frac{\Pr(\bigwedge_j (v \not\to u_j \vee v \not\to B_t(d(u_{j+1}, t))) \mid v \to u_i \wedge v \to B_t(r) \wedge C_v = k)}{1}$$

$$\geq \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k) \cdot \Pr(\bigwedge_j (v \not\to u_j) \mid v \to u_i \wedge v \to B_t(r) \wedge C_v = k)$$

$$\geq \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k) \cdot \Pr(\bigwedge_j (v \not\to u_j) \mid C_v = k)$$

$$\geq \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k) \cdot \Theta(1)$$

$$= \Omega(\Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k))$$

So now we can compute the entire sum :

$$\begin{aligned}
\sum_v \Pr(v \to u_i \wedge v \to B_t(r) \mid u_0 u_1..u_i) &= \sum_v \sum_k \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k \mid u_0 u_1..u_i) \\
&\geq \sum_v \sum_{k=1}^{\frac{d(v,u_i)\ln n}{i}} \Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k \mid u_0 u_1..u_i) \\
&\geq \sum_v \sum_{k=1}^{\frac{d(v,u_i)\ln n}{i}} \Omega(\Pr(v \to u_i \wedge v \to B_t(r) \wedge C_v = k)) \\
&= \sum_v \sum_{k=1}^{\frac{d(v,u_i)\ln n}{i}} \Omega(\frac{1}{k^2} \cdot \frac{k \cdot r}{d(v,t)\ln n} \cdot \frac{k}{d(v,u_i)\ln n}) \\
&= \sum_v \Omega(\frac{r}{d(v,t)} \cdot \frac{1}{i \ln n}) \\
&= \Omega(\frac{r}{i})
\end{aligned}$$

So we have:

$$\begin{aligned}
\Pr(\exists v, v \to u_i \wedge v \to t \mid u_0 u_1..u_{i-1}) &\geq 1 - e^{-\sum_v \Pr(v \to u_i \wedge v \to t \mid u_0 u_1..u_{i-1})} \\
&\geq 1 - e^{-\Omega(\frac{r}{i})} \\
&\geq 1 - (1 - \frac{1}{2} \cdot \Omega \frac{r}{i})) \\
&= \Omega(\frac{r}{i})
\end{aligned}$$

The last two lines use the fact that $e^{-x} \leq 1 - \frac{x}{2}$ for $0 \leq x \leq 1$.
With this we have proven that $\Pr(\exists v, v \to u_i \wedge v \to t \mid u_0 u_1..u_{i-1}) = \Omega(\frac{r}{i})$. $\square$

5.5 Expected number of steps from u_2

Now we compute the number of remaining steps $X(u_2)$, using the previous lemma, needed to forward the message from u_2 to a constant distance r from the target.

Lemma 5. *Exists a value r for which the expected number of steps, given the path u_0, u_1, u_2 with $d(u_2, t) = d$ is $O(\ln d)$.*

$$E(X(u_2)) = O(\ln d)$$

Proof. Let $\mathcal{E}_i$ the fact that the message reaches ball of radius r centered on the target in exactly i steps. The probability that $X(u2)$ is greater or equal to some constant j is the probability that the message has not reached the target in less than $j+2$ steps, which can be expressed by the $\mathcal{E}_i$.

$$\begin{aligned}
E(X(u_2)) &= \sum_{j=1}^{d-r} \Pr(X(u_2) \geq j) \\
&\leq \sum_{j=1}^{d} \Pr(X(u_2) \geq j) \\
&= \sum_{j=1}^{d} \Pr(\bigwedge_{i=3}^{j-1} \neg\mathcal{E}_i) \\
&= \sum_{j=1}^{d} \Pr(\neg\mathcal{E}_3) \cdot \Pr(\bigwedge_{i=2}^{j-1} \neg\mathcal{E}_i \mid \neg\mathcal{E}_3) \\
&= \sum_{j=1}^{d} \prod_{i=3}^{j-1} \Pr(\neg\mathcal{E}_i \mid \bigwedge_{p\leq i} \neg\mathcal{E}_p) \\
&= \sum_{j=1}^{d} \prod_{i=3}^{j-1} (1 - \Omega(\frac{r}{i})) \\
&= (1-\Omega(\frac{r}{3})) + (1-\Omega(\frac{r}{3}))\cdot(1-\Omega(\frac{r}{4})) + .. + (1-\Omega(\frac{r}{3}))\cdot .. \cdot(1-\Omega(\frac{r}{d})) \\
&= (1-\Omega(\frac{c\cdot r}{3})) + (1-\Omega(\frac{c\cdot r}{3}))\cdot(1-\Omega(\frac{c\cdot r}{4})) + .. + (1-\Omega(\frac{c\cdot r}{3}))\cdot .. \cdot(1-\Omega(\frac{c\cdot r}{d}))\text{for some const}
\end{aligned}$$

Here we have two cases: or $c \geq 1$ and then $r = 1$ is enough, else $r = \frac{1}{c}$. In both cases we have :

$$\begin{aligned}
&\leq \Omega(\frac{1}{1} + \frac{1}{2} + .. + \frac{1}{d}) \\
&= O(\ln d)
\end{aligned}$$

So proven that there exists a value r for which the expected number of steps, given the path u_0, u_1, u_2 with $d(u_2, t) = d$, to reach a ball of radius r centered on t is $O(\ln d)$. $\square$

6 Proof of the result

In this section we will prove that the routing protocol reaches the target in a $O(1)$ expected number of step.

First we have to consider the case that the source u_0 of the message is at a distance $d(u_0, t) \geq \sqrt{n}$ from the target t, from that case we will have two subcases : if the message, after two steps, is close enough of the target or not. The other case is when the source u_0 is at a distance $d(u_0, t) \leq \sqrt{n}$ from t.

If we have $d(u_0, t) \geq \sqrt{n}$, the probability that message fails to reach a distance lower that $\sqrt{n}$ in one step is $\frac{1}{n^{\Omega(1)}}$ from the observation 1. So in that case, from the lemma 1, the expected number of steps to reach the target is $O(\ln^2 n)$. In the other case, which happens with the probability $1 - \frac{1}{n^{\Omega(1)}}$, we have to compute the expected number E of remaining steps to reach t:

$$E = \sum_{r \geq 1} \Pr(d(u_2, t) = r) \cdot E(\text{length of the path from } u_2 \text{ to } t \mid u_0, u_1, u_2)$$

Thanks to the previous lemma, we know that there exists a constant value d for which $E(\text{length of the path from } u_2 \text{ to } B_t(d) \mid u_0, u_1, u_2, d(u_2, t) = r)$ is $O(\ln r)$ and we have such a constant for all the r considered. Thus the maximum value d works for all of them, so we get :

$$E = \sum_{r\geq 1} \Pr(d(u_2,t)=r) \cdot E(\text{length of the path from } u_2 \text{ to } t \mid u_0,u_1,u_2)$$

$$\begin{aligned}
E &\leq d + \sum_{r\geq 1} \Pr(d(u_2,t)=r) \cdot E(\text{length of the path from } u_2 \text{ to } B_t(d) \mid u_0,u_1,u_2) \\
&= d + \sum_{r\geq 2} (\Pr(u_2 \leq r) - \Pr(u_2 \leq r-1)) \cdot O(\ln r) \text{ from Lemma 4} \\
&= d + \sum_{r\geq 2} ((1-\frac{1}{r^\epsilon}) - (1-\frac{1}{(r-1)^\epsilon})) \cdot O(\ln r) \text{ from observation 2} \\
&= d + \sum_{r\geq 2} (\frac{1}{(r-1)^\epsilon} - \frac{1}{r^\epsilon}) \cdot O(\ln r) \\
&= d + \sum_{r\geq 2} (\frac{r^\epsilon - (r-1)^\epsilon}{(r\cdot(r-1))^\epsilon}) \cdot O(\ln r) \\
&= d + \sum_{r\geq 2} (\frac{r^\epsilon \cdot (1-(1-\frac{1}{r})^\epsilon)}{(r\cdot(r-1))^\epsilon}) \cdot O(\ln r) \\
&= d + \sum_{r\geq 2} (\frac{(1-(1-\frac{1}{r})^\epsilon}{(r-1)^\epsilon}) \cdot O(\ln r) \\
&\leq d + \sum_{r\geq 2} (\frac{O(\frac{\epsilon}{r})}{(r-1)^\epsilon}) \cdot O(\ln r) \\
&= d + \sum_{r\geq 2} O(\frac{\epsilon}{r\cdot(r-1)^\epsilon}) \cdot O(\ln r) \\
&\sim d + \sum_{r\geq 2} O(\frac{\epsilon}{r^{1+\epsilon}}) \cdot O(\ln r) \\
&\leq d + \sum_{r\geq 2} O(\frac{1}{r^{1+\epsilon}}) \cdot O(\ln r) \\
&= d + \sum_{r\geq 2} O(\frac{\ln r}{r^{1+\epsilon}}) \\
&\leq d + \sum_{r\geq 2} O(\frac{1}{r^{1+\epsilon'}}) \text{ with } \epsilon' < \epsilon \\
&\leq O(d+1-\frac{1}{n^{\frac{2+\epsilon'}{2}}}) \\
&= O(1)
\end{aligned}$$

So for the entire routing we get $2+O(1)=O(1)$ with probability $1-\frac{1}{n^{\Omega(1)}}$ and $O(\ln^2 n)$ with probability $\frac{1}{n^{\Omega(1)}}$ so the expected number of steps when $d(u_0,t) \geq \sqrt{n}$ is :

$O(1) \cdot (1 - \frac{1}{n^{\Omega(1)}}) + O(\ln^2 n) \cdot \frac{1}{n^{\Omega(1)}} = O(1)$.

In the case $d(u_0, t) \leq \sqrt{n}$ we have to adapt the probabilities : the probability of that first step (which corresponds to the second in the previous case) will reach a ball of radius r centered on t is still $1 - \frac{1}{r^{\Omega(1)}}$, and then we can use the other lemma since the only things that differs is the length of the path traveled which is decreased by 1.

$$\Pr(\exists v, v \to u_1 \wedge v \to B_t(r)) = 1 - \prod_v (1 - \Pr(v \to u_1 \wedge v \to B_t(r)))$$

$$\geq 1 - e^{\sum_v \sum_k \Pr(c_v = k \wedge v \to u_1 \wedge v \to B_t(r))}$$

Then the end of the computation is the same as in the proof of the lemma 3 so we also get that the step reaches a ball of radius r centered in t with probability $1 - \frac{1}{r^{\Omega(1)}}$.

The lemma 4 remains the same and we just have to adapt the proof of the lemma 5: we have one term more in the sum, we just need the term 1 to get the harmonic sum so instead of adding both 1 and 1/2 we just add 1 to get our $O(\ln d)$ result.

Then by the same argument as before we get that the expected number of steps for a source at a distance at most $\sqrt{n}$ from a target to send a message to this target is $O(1)$.

So, in any case, our protocol performs a routing between any source-target pair in $O(1)$ expected number of steps. □

To proove that $O(1)$ is the best complexity we can achieve for a routing algorithm let proceed by contradiction: let $O(C)$ the complexity of a algorithm which performs better than $O(1)$. Then C decreases while n increases. Now we have two cases: or C keeps on decreasing and then we have a value n_0 for which we have a negative complexity which is impossible, or the C stagnates for values of n greater than some value n_0. In the latter case, the only one possible, we ended up having a constant complexity, which means it is no better than $O(1)$.

7 Conclusion and future work

So by taking the 1-dimensional grid graph with a variant number of nodes defined by Fraigniaud and Giakkoupis ([12]), increasing the expected number of long-range contact to $O(\ln n)$ and adding knowledge we achieve a routing in $O(1)$ steps which is optimal for the complexity of a routing algorithm. The fact that we get an optimal complexity also allows us to conclude that additional look-ahead is useless as it was found by Manku, Naor and Wieder in their study over original Kleinberg's model ([7]).

It would be interesting to remove the extra-knowledge of the nodes and try to simulate it: instead of already knowing the neighbors of your neighbors, you send a mail to all your neighbors to ask them about their closest neighbor from the target. By making the edges bidirectional we double the number of edges, so the average number of neighbor of the nodes remains $O(\ln n)$ then it would be tempting to say that this protocol allows to find the short paths by sending $O(\ln n)$ messages since every node of the $O(1)$ nodes of the path asks all their neighbors. Unfortunately that is wrong because thought the average degree of a *random* node is $O(\ln n)$ that is not the case for the one of the path because in average, the average number of friends of your neighbors is higher than yours. This phenomenon is called the "friendship paradox" and is explained ([20]) by the fact when you compute the average number of friends you will count the nodes with a lot of friends once for every friends of its! So node with high degree have an huge influence on the average value compare to the one with few neighbors.

Due to a matter of time we did not study a protocol based on homophily and degree disparity as we intended to. The idea is to limit the knowledge of the nodes concerning their neighborhood: instead of knowing the neighbors of their neighbors, the nodes only know the degree of their neighbors. The underlying algorithm to this knowledge, i.e. not forward to the node which is the closest nor the one with the more neighbors but to the one which is the more likely to forward to a node close to the target (this probability depends on both the degree and the distance from the target), have been studied by Simsek and Jensen ([9]) and was very effective. In our case we should have forwarded the message to the node v which maximizes $1-(1-\frac{1}{d(v,t)\ln n})^{C_v}$, i.e. which maximizes the probability of having at least one edge directed to the target. It depends on both the distance of the node from the target and the degree C_v of the node v.

Computing the diameter is also in the continuation of this internship: the diameter of a network is the longest of the shortest path there are in a network. To compute it you need select, for every pair of nodes, the shortest path between them; then, now you have one path for every pair of nodes, you select the longest one and its size is the diameter of the network. The diameter is the time needed for a message to be forwarded from a source to a target using the shortest path, in the *worst case*. The shortest is it, the best is it for routing assuming you are able to find the right path. It is also interesting for computing a average efficiency for the worst pair; in our case we only compute the average efficiency for *any* pair.

Also it would be interesting to generalize these results to the l-dimensional grid as it was done with most of all the works done on the grid-based networks. As with the other results, we could expect a $O(1)$ expected number of steps with good distributions. Then the study of the influence on the distribution on the efficiency of the algorithm and on the diameter should be a good continuation of this work.

8 References

[1] Martin Dietzfelbinger and Philipp Woelfel. Tight lower bounds for greedy routing in uniform small world rings. In Proceedings of the 41nd ACM Symposium on Theory of Computing (STOC), pages 591–600, 2009.

[2] Pierre Fraigniaud and George Giakkoupis. The effect of power-law degrees on the navigability of small worlds. In Proceedings of the 28nd ACM Symposium on Principles of Distributed Computing (PODC), pages 240–249, 2009.

[3] Pierre Fraigniaud and George Giakkoupis. On the searchability of small-world networks with arbitrary underlying structure. In Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC), pages 389–398, June 6–8 2010.

[4] George Giakkoupis and Nicolas Schabanel. Optimal path search in small worlds: dimension matters. In Proceedings of the 43nd ACM Symposium on Theory of Computing (STOC), pages 393–402, 2011.

[5] Jon Kleinberg. The small-world phenomenon: an algorithm perspective. In Proceedings of the 32nd ACM Symposium on Theory of Computing (STOC), pages 163–170, 2000.

[6] Jon Kleinberg. Complex networks and decentralized search algorithms. In International Congress of Mathematicians (ICM), pages 1019–1044, 2006.

[7] Gurmeet Singh Manku, Moni Naor, and Udi Wieder. Know thy neighbor's neighbor: the power of lookahead in randomized P2P networks. In Proceedings of the 36nd ACM Symposium on Theory of Computing (STOC), pages 54–63, 2004.

[8] Stanley Milgram. The small world problem. Psychology Today, 1:60–67, 1967.

[9] Ozgur Simsek and David Jensen. Decentralized search in networks using homophily and degree disparity. In International Joint Conferences on Artificial Intelligence (IJCAI), pages 304–310, 2005.

[10] Peter Sheridan Dodds, Muhamad Roby, and Duncan J Watts. An experimental study of search in global social networks. Science, 301:827–829, 2003.

[11] Gurmeet Singh Manku. Routing networks for distributed hash tables. In Proceedings of the 22nd ACM Symposium on Principles of Distributed Computing (PODC 2003), pages 133–142, July 2003.

[12] Pierre Fraigniaud and George Giakkoupis. Greedy Routing in Small-World Networks with Power-Law Degrees. Preprint.

[13] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. Reviews of Moderna Physics, 74(1):47–97, 2002.

[14] S. N. Dorogovtsev and J. F. F. Mendes. Evolution of networks: From biological networks to the Internet and WWW. Oxford University Press, 2003.

[15] M. E. J. Newman. The structure and function of complex networks. Society for Industrial and Applied Mathematics (SIAM) Review, 45(2):167–256, 2003.

[16] C. Martel and V. Nguyen. Analyzing Kleinberg's (and other) small-world models. In Proceedings of the 23rd ACM Symposium on Principles of Distributed Computing (PODC), pages 179–188, 2004.

[17] C. Martel and V. Nguyen. Analyzing and characterizing small-world graphs. In Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 311–320, 2005.

[18] J. Kleinberg. Small-world phenomena and the dynamics of information. In Advances in Neural Information Processing Systems (NIPS) 14, pages 431–438, 2001.

[19] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman. Search in power-law networks. Physical Review E, 64, 2001.

[20] Scott L. Feld. Why Your Friends Have More Friends Than You Do. The American Journal of Sociology, Vol. 96, No. 6 (May 1991), pp. 1464-1477.

[21] Duncan J. Watts, Steven H. Strogatz. Collective dynamics of 'small-world' networks. Nature 393, 1998, pp.440-442.

[22] J. Kleinberg. Navigation in a small world. Nature, 406:845, 2000.

[23] S. Lattanzi, A. Panconesi, and D. Sivakumar. Milgram-routing in social networks. In Proceedings of the 20th ACM International Conference on World Wide Web (WWW), pages 725-734, 2011.

[24] E. Lebhar and N. Schabanel. Almost optimal decentralized routing in long-range contact networks. In proceedings of the 31st International Colloquium on Automata, Languages, and Programming (ICALP), pages 894-905, 2004.

[25] D. Liben-Nowell, J.Novak, R. Kumar, P. Raghavan, and A. Tomkins. Geographic routing in social networks. In Proceedings of the National Academy of Sciences of th USA, 102(33):11623-11628, 2005.

[26] D. Coppersmith, D. Gamarnik, and M. Sviridenko. The diameter of a long range percolation graph. In Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 329-337, 2002.

[27] M. Dietzfelbinger, J. Rowe, I. Wegener, and P. Woefel. Tight bounds for blind search on the integers. In Proceedings of the 25th International Symposium on Theoretical Aspects of Computer Science (STACS), pages 241-252, 2008.

[28] P. Fraigniaud, C. Gavoille, and C. Paul. Eclecticism shrinks even small worlds. In Proceedings of the 23rd ACM Symposium on Principles of Distributed Computing (PODC), pages 169-178, 2004.

[29] B. Kim, C. Yoon, S. Han, and H. Jeong. Path finding strategies in scale-free networks. Physical Review E, 65:027103, 2002.

[30] N. Sarshar, P. O. Boykin, and V. P. Roychowdhury. Percolation search in power law networks : Making unstructured peer-to-peer networks scalable. In Proceedings of the 4th IEEE International Conference on Peer-to-Peer Computing (P2P), pages 2-9, 2004.

[31] A.-L. Barabási and R. Albert. Emergence of Scaling in Random Networks, Science 286, 509 (1999).