Optimal Link Deployment for Minimizing Average Path Length in Chain Networks

Zeki Bilgin, Murat Gunestas, Omer Demir, Sahin Buyrukbilen

Optimal Link Deployment for Minimizing Average Path Length in Chain Networks

Zeki Bilgin, Murat Gunestas, Omer Demir, and Sahin Buyrukbilen

General Directorate of Security, Turkey
{zbilgin,mgunestas,odemir,sbuyrukbilen}@egm.gov.tr

Abstract. This study considers chain-topology networks, which has certain inherent limitations, and presents an optimization model that augments the network by the addition of a new link, with the objective of minimizing Average Path Length (APL).

We built up a mathematical model for APL, and formulated our problem as Integer Programming. Then, we solved the problem experimentally by brute-force, trying all possible topologies, and found the optimal solutions that minimize APL for certain network sizes up to 1000 nodes. Later on, we derived analytical solution of the problem by applying Linear Regression method on the experimental results obtained.

We showed that APL on a chain-topology network is decreased by the proposed optimization model, at a gradually increasing rate from 24.81% to asymptotic value of 41.4% as network grows. Additionally, we found that normalized length of the optimal solutions decreases logarithmically from 100 % to 58.6048 % as network size gets larger.

Keywords: Chain networks, average path length, link deployment

1 Introduction

Many industries, today, prefer chain topology on their networks comprised of nodes connected each other consecutively throughout both long and narrow deployment areas like railways [1], highways [2], underground mines [3, 4], as well as in some special type of wireless sensor and mesh networks [3–7] and backbones of telecommunication systems [8]. Similarly, it is also a well-known implementation to connect (wi-fi) routers in daisy-chain topology to provide internet access at each floor in towers or high buildings.

A major disadvantage of chain networks is to have high *Average Path Length* (*APL*) relative to network size, unlike many other types of networks owning the properties of "small-world" networks [9]. *APL* is generally desired to be *small*, and is investigated analytically and numerically in many studies related to network design and optimization [10–18], social networks [15, 17, 18], computing [19], and logic design [20].

In this paper, we examine *APL* for chain networks, and propose an optimization model based on an additional link deployment to the network, with the objective of minimizing *APL*. We derive analytical formulation for *APL* prior to and subsequent to optimization process, as well as obtain numerical results which precisely agreed with analytical analysis.

2 Network Model

Suppose we have a chain-topology network with n nodes, containing bidirectional links between consecutive nodes. This network can be represented as a path graph, P_n, with undirected edges[1] as depicted in Figure 1.

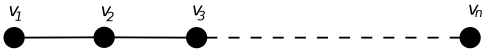

Fig. 1: A path graph representing a chain-topology network with n nodes

Suppose, we aim at augmenting this network further by adding a new performance enhancing link between a certain pair of nodes on the network[2] as illustrated in Figure 2. We should notice at this point that the augmentation process (i.e. adding a new link) is intentionally confined by just one new link in order to keep the optimization cost minimum, and that the implementation cost of such a new link can be assumed fixed regardless of the distance between connected nodes, which is true especially for the leased lines obtained from ISPs.

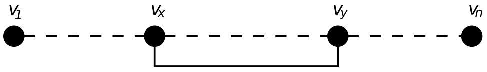

Fig. 2: Adding a new enhancement link (i.e. edge) connecting v_x and v_y

We are now ready to ask our optimization problem:

Main Problem: Which nodes should be connected to reach the objective of *minimizing average path length (APL)* on the network?

Not only does the proposed optimization model minimize APL, but also it improves robustness on chained networks by means of generating alternative routes, as well as reduces cost of packet transmissions.

3 Related Work

3.1 Chain Networks

Given the side effects of unbalanced energy consumptions at nodes in chain networks used in underground mines or on trains, the studies of [1, 3, 4, 6] proposed different protocols or node deployment strategies, aiming to provide balanced energy consumptions at nodes in order to increase network lifetime.

[1] The edges are undirected because of bidirectional transmissions between nodes.

[2] This could be realized via several ways like obtaining a leased line connection from an ISP between the two points of interest, or using a long-range radio link.

Agbinya [2] discussed a specific application of chain networks on highways, and addressed certain characteristics of the network such as interference level, coverage area and path loss; on the other hand, Zhou et al. [5], in a recent study, considered Chain-typed Wireless Sensor Networks (CWSN) deployed in coal mines, and proposed a source-aware redundant packet forwarding scheme for emergency information delivery in CWSN.

Leu and Huang [7] proposed a mathematical model that calculates the maximum throughput of a Wireless Mesh Network in chain-topology, dealing with signal interference, hidden nodes and STDMA time slots among nodes.

Flammini et al. [8] considered the construction of wireless ATM layouts for a chain of base stations, and showed that the problem studied was NP-complete for special instances, and provided optimal solutions for certain cases.

3.2 Average Path Length (APL)

Several researchers derived analytical formulation of APL for different type of networks. For instance, Kleinrock and Silvester [21] considered random graphs; Fronczak et al. [18] and Guo et al. [14] studied a large class of uncorrelated random networks with hidden variables; Zhang et al. [17] examined Apollonian networks; Peng [16] dealt with Sierpinski pentagon; Gulyás et al. [13] focused on the networks with given size and density; Chen et al. [11] investigated Barabási–Albert scale free model; Zhi-guang et al. [10] discussed belt-type networks; and Gao et al. [22] analysed Sierpinski gasket in a recent article.

In the field of logic design, Butler et al. [20] studied APL of binary decision diagrams by deriving the APL for various functions, and showed that the APL for benchmark functions is typically much smaller than for random functions.

Mao and Zhang [19] considered the computation problem of APL for large scale-free networks, and presented a dynamic programming model to solve the load-balancing problem for coarse-grained parallelization. Yen et al. [12] presented an efficient method for updating the closeness centrality of each vertex and the APL of a network, where edges change dynamically as in the case of social networks. In a recent study, Reppas et al. [15] introduced rewiring rules to tune APL on a network while keeping the degree and clustering coefficient distribution unchanged.

To the best of our knowledge, ours is the first study to propose an optimization model aiming to minimize APL for chain networks by optimal deployment of an incremental link.

4 Mathematical Model

4.1 Pure Path

Average path length, APL, of a network is an important parameter showing the efficiency of information transmission on the network, and can be calculated by finding the shortest path between all pairs of nodes, adding their lengths[3] up, and then dividing by the total number of pairs.

To find the mathematical expression for APL of a chain network, let P_n be a path graph including n vertices indexed in sequence from 1 to n, like $v_1, v_2, ..., v_n$, as depicted in Figure 1. It is obvious that the shortest path between a certain pair of nodes on a path graph is the *subpath*, having no alternative, between this pair of nodes. Moreover, the length of such a subpath is equal to the number of edges on itself. Thus, since the vertices are indexed in order, length of a subpath (PL) between vertices of v_j and v_k on P_n can be stated rigorously as follows.

$$PL_{(v_j,v_k)} = |k - j|$$

Then, the Equation 1 gives the sum of path lengths for all (unordered) pairs[4]

$$\sum PL_{(AllPairs)} = \sum_{i=1}^{n-1} \sum_{k=1}^{n-i} k \tag{1}$$

After rewriting the Equation 1, and dividing by the number of all pairs, which is $\frac{n(n-1)}{2}$, we find the APL for the path graph of P_n as given in Equation 2.

$$APL_{P_n} = \frac{\sum PL_{(AllPairs)}}{\frac{n(n-1)}{2}} = \frac{n+1}{3} \tag{2}$$

According to Equation 2, the APL for a chain-topology network is linearly proportional with the length of the chain or the number of nodes, i.e. O(n), and almost equal to one third of network diameter.

4.2 Path with an Additional Edge

Let $P_n^{'}$ be a graph obtained by adding a new edge (v_x, v_y) to the path graph P_n as depicted in Figure 2. Rigorously,

$$P_n^{'} = P_n \cup (v_x, v_y)$$

To built a general mathematical expression for APL on $P_n^{'}$, we first studied on small networks (e.g. around 10 nodes), manually calculated APL, and produced a sketchy formula for APL. Then, we extended our work with larger networks, as repeatedly checking accuracy of the formula, and revised it when needed until the formula persistently gave correct values for all networks investigated. This process yielded Equation 3. Yet we also verified its correctness via experiments as described in the following sections.

$$APL_{P_n^{'}} = \frac{\sum_{i=1}^{t-1} \sum_{k=1}^{t-i} k + (h-1)(\sum_{i=1}^{x} i + \sum_{i=1}^{n-y+1} i) - 1 + R}{\frac{n(n-1)}{2}} \tag{3}$$

[3] The length of a path is measured here in terms of hop count

[4] By all pairs we mean all unordered pairs because the edges are undirected, and for this reason, it is enough to count path lengths in only one direction.

where $h = y - x$, $t = n - h + 1$, and

$$R = \begin{cases} (2n-h-1)\sum_{i=0}^{h/2} i - h(n-h+1) + 2, & \text{if } h \text{ is even} \\ (2n-h-1)\sum_{i=0}^{(h-1)/2} i + \frac{(h+1)^2}{4} - \frac{(h-1)(n-h+3)}{2}, & \text{if } h \text{ is odd} \end{cases}$$

Thus, we obtained analytical expressions for APL prior to and subsequent to additional link attachment into a path graph, as in Equations 2 and 3, which allows us to formulate our problem in the form of *Integer Linear Programming (IP)* as follows:

$$\begin{aligned} \text{minimize} \quad & APL_{P'_n} \\ \text{subject to} \quad & \text{n, x, y are integer} \\ & x < y \\ & 1 \leq x \leq n \\ & 1 \leq y \leq n \end{aligned}$$

where $APL_{P'_n}$ is given in Equation 3.

It is known that *IP* is NP-hard [23], which implies that there is no known polynomial-time solution for *IP* problems. Yet, in the following sections, we will solve certain instances of the problem above by experimentally in the first place, and then, construct a general analytical solution for any value of network size (i.e. n) by means of linear regression method.

5 Finding Optimal Solutions

5.1 Numerical Solutions by Experiment

To find optimal solutions for certain cases of the problem introduced, we prepared an experimental set-up shown as pseudo-code in Figure 3.

Fig. 3: Experimental Setup for Determining APL

```
for $n \leftarrow 3$ to 1000 do                                  ▷ Varying network size
    Enter Adjacency List of the Graph        ▷ Defining network topology for all cases
    function $APL(Graph)$
    Find the minimum $APL$ among all calculated values at each step of $n$
function APL($G$)                                              ▷ Determines $APL$ for $G$
    $c \leftarrow 0$                                                        ▷ Counter
    $APL \leftarrow 0$
    for all Pairs $(v_i, v_j)$ in $G$ do
        function DIJKSTRA$(v_i, v_j)$                                  ▷ Runs Dijkstra
            return Shortest Path (SP) between $v_i$ and $v_j$
        $APL \leftarrow (APL * c + Length(SP)/(c+1)$
        $c \leftarrow c + 1$
    return $APL$
```

In the experiment, we incremented network size from 3 nodes to 1000 nodes, and varied attachment points (i.e. vertices) of the additional link for all possible cases as a brute-force approach. At each step of network size, we first defined network topology by entering adjacency list for the network, for all possible deployment of the additional link as varying variables of x and y, which represent the relative location of vertices v_x and v_y. Then, for each topology, we found the shortest paths for all pairs by implementing Dijkstra's well-known shortest path algorithm [24, 23]. Notice that Dijkstra determines the shortest path between only one pair of nodes, and for this reason, we iteratively employed it for all pairs in the graph (i.e. topology). After calculating lengths (i.e. hop counts) of the shortest paths for all pairs, we took average of them, and thus found APL. Finally, we identified the minimum APL among all calculated APLs yielded as varying locations of v_x and v_y. This experimental process was repeated for certain network sizes of 3, 5, 10, 20, 50, 100, 200, 500 and 1000 nodes.

Table 1: Experimental Results Obtained by Brute Force Computation

Network	APL	APL	Min APL	Optimal Solutions
Size (n)	for *Path*	for *Ring*	(*Optimal*)	(x_{opt}, y_{opt})
	P_n	$P_n \cup (v_1, v_n)$	$P_n \cup (v_{x\_opt}, v_{y\_opt})$	
3	1.33	1.0	1.0	(x=1, y=3)
5	2.0	1.5	1.5	(x=1, y=5)
10	3.66	2.77	2.53	(x=2, y=8) (x=3, y=9)
20	7.0	5.26	4.5	(x=4, y=16) (x=5, y=17)
50	17.0	12.75	10.37	(x=10, y=40) (x=11, y=41)
100	33.66	25.25	20.13	(x=21, y=80)
200	67.0	50.25	39.66	(x=41, y=159) (x=42, y=160)
500	167.0	125.25	98.24	(x=104, y=397)
1000	333.66	250.25	195.87	(x=207, y=793) (x=208, y=794)

Table 1 contains some of the numerical results acquired in the experiments, including optimal solutions that minimize APL as well as results belong to ring topologies (i.e. the cases in which the first and the last nodes of paths are connected each other by the additional link). The first column in the table includes network size in terms of the number of nodes, while the second and the third columns contain APL for *pure path* (P_n) and *ring topology* (i.e. $P_n \cup (v_1, v_n)$) respectively. Notice that *ring topology* occurs when the first and the last nodes on a path are connected each other. The fourth column involves minimum APL which appears when the additional link is placed optimally (i.e. $P_n \cup (v_{x\_opt}, v_{y\_opt})$). The last column shows optimal values of (v_x, v_y) that *minimize* APL.

Figure 4 shows experimental results in the form of 3-dimensional color mapping when network size equals to 100 nodes. As can be seen in the figure, APL has the minimum value (i.e. dark blue color) at around $x = 21$ and $y = 80$, or equivalently, vice versa. Notice that the red area from left bottom corner to right upper corner represent *Path* topology, whereas the points both at the left up corner and at the right down corner produce *Ring* topology.

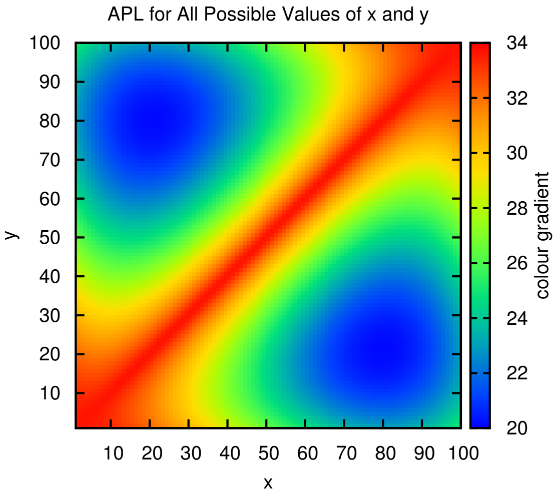

Fig. 4: APL for varying values of x and y, when network contains 100 nodes

Verification of the mathematical model: One might doubt the accuracy of our mathematical model presented in Section 4, i.e. Equation 3. To verify correctness of this mathematical expression, we first computed APL values by using Equations 3 as assigning all possible values to the variables up to network size of 1000 nodes, and then searched out the instances giving minimum APL for each network size. Afterwards, we compared minimum APL values computed in Equation 3 with the APL values yielded from the experimental calculations for certain network sizes as listed in Table 1. We eventually observed that both the mathematical model and the experimental calculations give precisely the same outcomes for APL, which shows the consistency between these two different approaches.

5.2 Analytical Solution by Linear Regression

Table 1 contains numerical results of optimal solutions for certain network sizes. However, to make a comprehensive analysis including asymptotic behaviour of optimal solutions and other variables, we need to establish analytical relations between these variables. For this purpose, we applied a linear regression method on the numerical results at hand, based on least square technique, and consequently, found the following relations.

$$APL_{P'_n}(n) = 0.195331 * n + 0.559447 \tag{4}$$

$$x_{opt} = Round(0.207174 * n - 0.0251311) \tag{5}$$

$$y_{opt} = Round(0.793222 * n + 0.0497688) \tag{6}$$

where $Round(z)$ is a function which returns the *nearest integer* to z.

In fact, Equations 5 & 6 give precise answers to the main problem asked at the beginning of this paper. Equation 4, on the other hand, yields exact outcome for APL when an optimal solution is applied.

6 Discussions

6.1 Average Path Length (APL)

Figure 5 depicts APL for both P_n and $P_n^{'}$ when optimal values of (v_x, v_y) is applied, as network size varies from 3 nodes to 1000 nodes. As seen in the figure, APL linearly increases for both cases as network size grows. However, notice that P_n has higher slope than $P_n^{'}$, which means that adding extra edge reduces APL on a network.

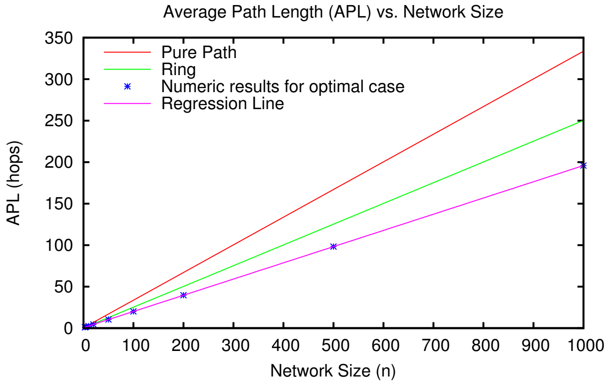

Fig. 5: Average Path Length (APL) for different topologies as network size grows

Notice that there is also model fit (i.e. regression line) which is obtained by linear regression. Goodness of fit can even be visually evaluated in Figure 5, as the fitted line and numerical data exactly matches each other.

6.2 Improvement

Figure 6 exhibits the *Improvement*, i.e. the rate of decrement, on APL when an additional edge is placed to the network at optimal positions. As can be followed in the figure, the improvement rate begins with a slow growth at around 24.81 % when $n = 3$, followed by a period of moderate growth, and then back to a period of slow growth asymptotically approaching to 41.4 %, which is consistent with the analytical analysis below.

$$\begin{aligned}Improvement &= 100 * (APL_{P_n} - APL'_{P_n})/APL_{P_n}\\ &= 100 * \frac{(\frac{n+1}{3} - 0.195331n - 0.559447)}{(n+1)/3}\\ &= 300 * \frac{0.138n - 0.226114}{n+1}\\ &= \frac{41.4n - 67.8342}{n+1}\end{aligned}$$

$$\lim_{n \to +\infty} Improvement = 41.4\%$$

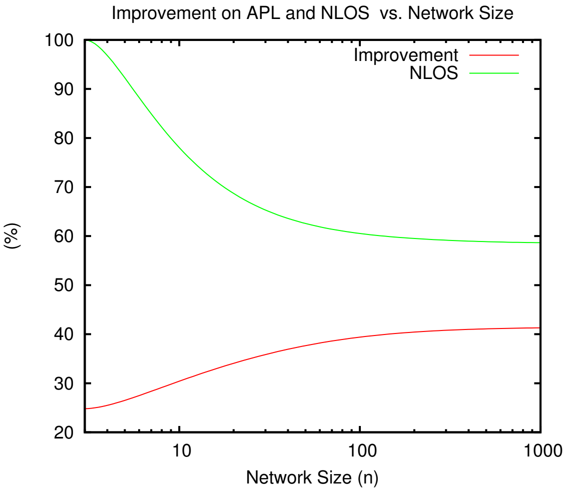

Fig. 6: Improvement on APL and Normalized Length of Optimal Solutions (NLOS) as network size grows (smoothed)

6.3 Optimal Solutions

Equations 5 & 6 are optimal solutions in *analytic form*, while values on the fifth column in Table 1 are *numeric* solutions for certain network sizes. Thanks to Equations 5 & 6, one can readily determine optimal values of (v_x, v_y) for any network size. It is interesting to observe that the optimal solutions, when $n = 3$ and $n = 5$, are two end points of the path (i.e. (v_x, v_y) equals to (v_1, v_3) and (v_1, v_5) respectively) . As network size grows, the optimal values of v_x and v_y slide gradually towards the center of the network. This observation motivated us to investigate normalized length between two end points (i.e. v_x and v_y) of optimal solutions in the next part.

Another observation here is that there is only one (i.e. unique) optimal solution when the network size (n) is odd, whereas there may emerge many optimal solutions when n is even, as can be observed in the fifth column of Table 1. We discovered that alternative optimal solutions for the same network yield isomorphic graphs when they are applied.

6.4 Normalized Length of Optimal Solutions (NLOS)

$NLOS$ represents the normalized distance between two end points (i.e. v_{x_opt} and v_{y_opt}) of optimal link deployments that minimize APL. The normalization process is performed with respect to network size. Figure 6 includes $NLOS$ as network size logarithmically grows. It can be deduced from the figure that the $NLOS$ is reduced logarithmically beginning from 100 % to around 58.6 %, which is consistent with the analytical analysis below. This means that the optimal solutions occur at around two end points of chain-topology when network size is small, whereas attachment points of optimal solutions move away from this end points as network size grows.

$$\begin{aligned} NL &= 100 * \frac{|y_{opt} - x_{opt}|}{n-1} \\ &= \frac{Round(58.6048 * n + 7.48999)}{n-1} \end{aligned}$$

$$\lim_{n \to +\infty} NL = 58.6048\%$$

7 Conclusion

Chain-topology networks performs poorly in certain performance metrics such as throughput, robustness, energy efficiency in data transmissions [1–7]. This is mostly due to fact that average path length (APL) in chain-topology is extremely high, which is almost one third of network size as we showed.

In this study, we aimed at compensating this deficiency by presenting an optimization model in which incremental link deployment was considered, with the objective of minimizing APL on a chain network. For this purpose, we first discovered mathematical expression of the objective, as well as formulated it in the form of Integer Programming (IP). Then, we prepared an experimental setup in order to determine APLs of all possible topologies generated by placing an additional link to varying locations on a chain-topology network. Thus, we found optimal solutions that minimize APL for specific network sizes up to 1000 nodes, and also verified accuracy of our mathematical model. Through the experiments, for each specific network size, we implemented Dijkstra's shortest path algorithm for all pairs, and took average of their lengths in terms of hop count to calculate corresponding APL values. Afterwards, we derived analytical solution by implementing Linear Regression method on the data obtained experimentally, which allowed us to see asymptotic behaviour of the solutions.

Our analyses showed that the optimization model proposed was able to reduce APL on chain-topology networks at a rate of between 24.81% and 41.4%, with gradually increasing ratio as network size grows. Moreover, we found that normalized length of the additional link for optimal solution asymptotically approached to 58.6% of network size.

Besides contribution of such an additional link optimally implanted for minimizing the APL, further research is required to improve other performance characteristics of chain-topology networks, such as ensuring load balancing.

References

1. Kim, S., Kim, J., Yoo, Y.: Container Security Device Chain Network for Safe Railway Transportation. International Journal of Distributed Sensor Networks (2014)
2. Agbinya, J.: Design Considerations of MoHotS and Wireless Chain Networks. Wireless Personal Communications 40(1), 91–106 (2007), http://dx.doi.org/10.1007/s11277-006-9103-0
3. Haifeng, J., Jiansheng, Q., Yanjing, S., Guoyong, Z.: Energy Optimal Routing for Long Chain-type Wireless Sensor Networks in Underground Mines. Mining Science and Technology (China) 21(1), 17 – 21 (2011), http://www.sciencedirect.com/science/article/pii/S1674526410000219
4. Yuan, Y., Shen, Z., Quan-fu, W., Pei, S.: Long Distance Wireless Sensor Networks Applied in Coal Mine. Procedia Earth and Planetary Science 1(1), 1461 – 1467 (2009), http://www.sciencedirect.com/science/article/pii/S1878522009002264
5. Zhou, G., Zhu, Z., Zhang, P., Li, W.: Source-aware Redundant Packet Forwarding Scheme for Emergency Information Delivery in Chain-typed Multihop Wireless Sensor Networks. International Journal of Distributed Sensor Networks (2015)
6. Chen, Y., Ren, T., Liu, Y., Zhou, Y.: Lifetime Maximization Algorithm for Chain Wireless Sensor Networks. Information Technology Journal 12(4), 656–663 (2013)
7. Leu, F.Y., Huang, Y.T.: Maximum Capacity in Chain-topology Wireless Mesh Networks. In: Wireless Telecommunications Symposium, 2008. WTS 2008. pp. 250–259 (April 2008)
8. Flammini, M., Gambosi, G., Navarra, A.: Wireless ATM Layouts for Chain Networks. Mobile Networks and Applications 10(1-2), 35–45 (2005), http://dx.doi.org/10.1023/B%3AMONE.0000048544.06992.f7

9. Watts, D.J., Strogatz, S.H.: Collective Dynamics of 'small-world' Networks. Nature 393, 440–442 (6 1998)
10. Xu, Z.g., Zhu, L.j., Shi, Y.s., Jiang, h.: Research of Scalability of the Belt-type Sensor Networks. Journal of Shanghai Jiaotong University (Science) 17(2), 237–240 (2012), http://dx.doi.org/10.1007/s12204-012-1259-z
11. Chen, F., Chen, Z., Wang, X., Yuan, Z.: The Average Path Length of Scale Free Networks. Communications in Nonlinear Science and Numerical Simulation 13(7), 1405 – 1410 (2008), http://www.sciencedirect.com/science/article/pii/S1007570406002383
12. Yen, C.C., Yeh, M.Y., Chen, M.S.: An Efficient Approach to Updating Closeness Centrality and Average Path Length in Dynamic Networks. In: Data Mining (ICDM), 2013 IEEE 13th International Conference on. pp. 867–876 (Dec 2013)
13. Gulyás, L., Horváth, G., Cséri, T., Kampis, G.: An Estimation of the Shortest and Largest Average Path Length in Graphs of Given Density. ArXiv e-prints (Jan 2011)
14. Guo, D., Liang, M., Li, D., Jiang, Z.: Effect of Random Edge Failure on the Average Path Length. Journal of Physics A: Mathematical and Theoretical 44(41), 415002 (2011), http://stacks.iop.org/1751-8121/44/i=41/a=415002
15. Reppas, A.I., Spiliotis, K., Siettos, C.I.: Tuning the Average Path Length of Complex Networks and Its Influence to the Emergent Dynamics of the Majority-rule Model. Mathematics and Computers in Simulation 109, 186 – 196 (2015), http://www.sciencedirect.com/science/article/pii/S0378475414002456
16. Peng, J.: Average Path Length for Sierpinski Pentagon. International Journal of Advancements in Computing Technology 5(5), 724–732 (03 2013), http://search.proquest.com.mutex.gmu.edu/docview/1400499426?accountid=14541
17. Zhang, Z., Chen, L., Zhou, S., Fang, L., Guan, J., Zou, T.: Analytical Solution of Average Path Length for Apollonian Networks. Phys. Rev. E 77, 017102 (Jan 2008), http://link.aps.org/doi/10.1103/PhysRevE.77.017102
18. Fronczak, A., Fronczak, P., Hołyst, J.A.: Average Path Length in Random Networks. Phys. Rev. E 70, 056110 (Nov 2004), http://link.aps.org/doi/10.1103/PhysRevE.70.056110
19. Mao, G., Zhang, N.: Analysis of Average Shortest-path Length of Scale-free Network. Journal of Applied Mathematics (2013)
20. Butler, J., Sasao, T., Matsuura, M.: Average Path Length of Binary Decision Diagrams. Computers, IEEE Transactions on 54(9), 1041–1053 (Sept 2005)
21. Kleinrock, L., Silvester, J.: Optimum Transmission Radii for Packet Radio Networks or Why Six is a Magic Number. In: Proceedings of the IEEE National Telecommunications Conference. vol. 4 (1978)
22. Gao, F., Le, A., Xi, L., Yin, S.: Asymptotic Formula on Average Path Length of Fractal Networks Modeled on Sierpinski Gasket. Journal of Mathematical Analysis and Applications 434(2), 1581 – 1596 (2016), http://www.sciencedirect.com/science/article/pii/S0022247X15009312
23. Cormen, T.H., Stein, C., Rivest, R.L., Leiserson, C.E.: Introduction to Algorithms. McGraw-Hill Higher Education, 2nd edn. (2001)
24. Dijkstra, E.W.: A Note on Two Problems in Connection with Graphs. Numerische Math 1, 269–271 (1959)