A Fault Detection Based Routing Scheme for Google Project Loon

Sourav Kumar Bhoi[1], Sanjaya Kumar Panda[2], ***Member*, *IEI*** and Kalyan Kumar Jena[3]
[1,2,3]*Department of Computer Science and Engineering*
[1,3]*Parala Maharaja Engineering College, Berhampur – 761003, Odisha, India*
[2]*National Institute of Technology, Warangal – 506004, India*
{[1]souravbhoi, [2]sanjayauce, [3]kalyankumarjena}@gmail.com

***Abstract*—Google Project Loon (GPL) is a low cost infrastructure, which is mainly developed to provide Internet connectivity to the rural and remote areas by building a hybrid wireless ad-hoc network using Balloon to Balloon Communication (B2B) and Balloon to Ground Communication (B2G). In this network, balloons are equipped with wireless transceivers and placed on stratosphere over the regions, where Internet connectivity is less. However, the performance of the network may get reduced due to the presence of faulty balloons. Here, the balloons are faulty due to damage of the balloon, out of range, hardware or software faults. In this work, a fault detection based routing scheme is proposed, where the node (user/balloon) while selects a forwarder node, checks the fault status of that node in its local database. If the node is faulty, it selects another forwarder and sends data to the destination in a shortest time. The performance of the proposed scheme over the loon project is evaluated using simulations. From the simulations, it is observed that the data should be forwarded in a fault free path for reliable communication between the user and the Internet Service Provider (ISP).**

***Index Terms*—Google Project Loon, Fault Detection, Balloon to Balloon Communication, Balloon to Ground Communication, Internet Service Provider.**

I. Introduction

Currently, the Internet is a need for the people to connect globally. According to the reports provided by the Google [1], only two thirds of the people in this world use Internet connection. To solve this issue, Google developed a project in 2013 to provide high speed Internet to the remote areas, where reliable Internet connectivity lacks [1]. In disaster situation also, the telecommunication infrastructure gets damaged, which creates problems for the users to connect the Internet. In this situation, Google Project Loon (GPL) [1-17] works well by providing infrastructure to access the Internet. The main concept of the project is that balloons are flying in the stratosphere and they are connected wirelessly. They can connect to a user as well as the Internet Service Provider (ISP). When a user from a rural area wants to access Internet, it sends the request message through the balloon network to the ISP. Then the ISP provides the Internet service by sending the requested packets to the users.

The main problem with the GPL is that the balloons may get damaged due to battery life dead, hardware fault, software fault and out of range. If these conditions arise, then the communication may get affected due to a faulty node (balloon) in the network. The solution to this condition is that while forwarding the data to the next forwarder node the current status of the node should be checked by the source node in its local database. If the node is faulty, then another path should be chosen for reliable communication.

The main contributions of this paper are stated as follows.

(1) A fault detection based routing scheme is proposed, which detects the faulty balloons in the network and sends the data in a short time using a fault free path.

(2) The fault status of a balloon is periodically updated in the local database of other balloons in the communication range.

(3) The performance of the proposed scheme is compared with a shortest path routing scheme. From the simulation results, it is observed that the proposed fault tolerant routing scheme performs better in terms of end to end delay, detection accuracy and false alarm rate.

The rest of the paper is organized as follows. Section II presents the related works. Section III presents the proposed fault tolerant routing scheme. Section IV presents the simulation and results. Section V concludes this work.

II. Related Work

Many ideas have been proposed after GPL is developed. Facebook Drones [1] are developed by Facebook to compete the loon balloons. The drones can fly for more time with high Internet speed and cover more areas. However, it requires expensive parts to setup. The security and privacy are also an important concern in drones. Many routing protocols have been also proposed to send data in a fault free path by knowing the status of the wireless nodes [18-27]. However, from the literature survey, we found that no such work has been done in the area, where the nodes are flying in the stratosphere. Moreover, very less work has been done in the fault management of balloons. Therefore, as Internet connectivity requires high speed, the data should travel in a reliable path to deliver the request of the users at a shortest time, so that they can receive the data packets from the ISP in an easier and faster manner.

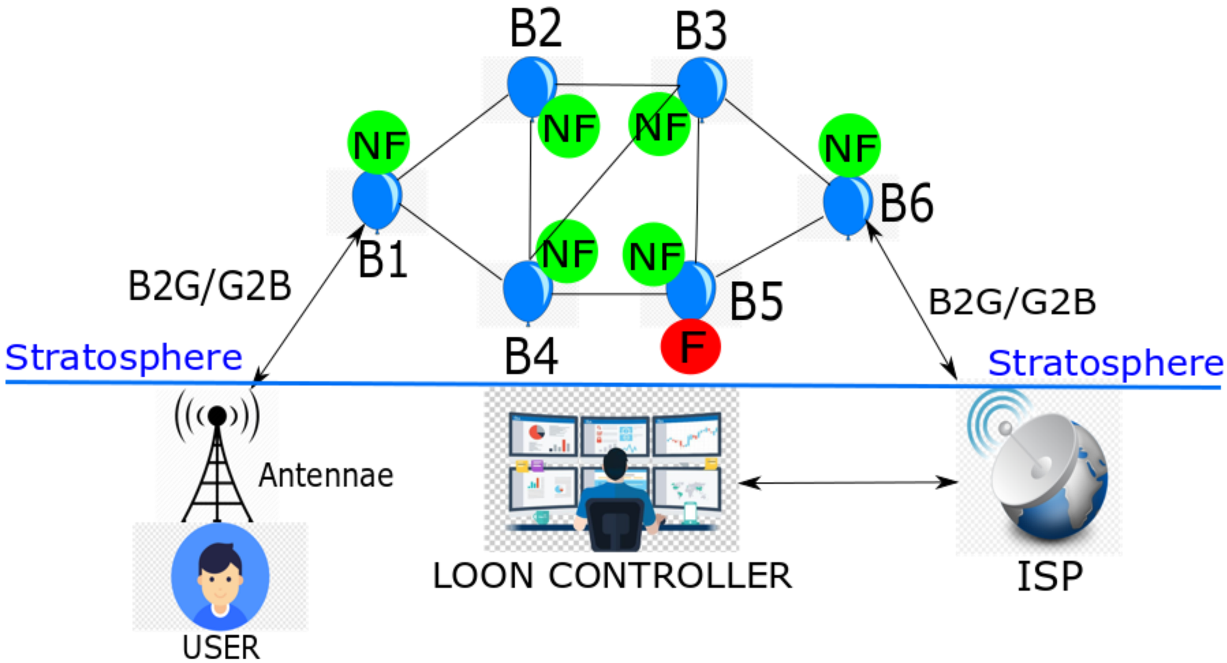

Fig. 1. Proposed system model for the fault detection based routing scheme (*B*: balloon, *F*: faulty and *NF*: not faulty).

III. Proposed Routing Scheme

This section describes about the network model, fault model and working of the proposed scheme. The network model describes about the network topology and the connectivity between the balloons. The fault model describes about the fault behavior of the balloons. The working section describes about the proposed routing mechanism in the balloon network.

A. Network Model

The balloons, in the network, are placed at the stratosphere over the desired location. The balloon has a wireless transceiver for connecting with other balloons or ISP or user in an ad-hoc manner. It also consists of solar panels for charging the batteries. The batteries are connected to the controller unit for power supply. The controller unit is connected to a GPS (global positioning system). A balloon beacons its position at a particular interval of time. The balloon in the communication range stores the position of that balloon. The balloons can be controlled to left or right direction by filling and releasing air. As the air in stratosphere moves in two different directions, if the balloon is moved up, then it moves in the left direction and if it is moved down, then it moves in the right direction. This is performed by the controlling unit. The location of the balloons is known to the project loon handler by receiving signals from the balloons. The balloons are connected to each other in ad-hoc fashion. The balloons, which are in communication range, can be connected to each other and they can send/receive data.

B. Fault Model

The balloon may get faulty due to many reasons, such as out of the communication range, hardware fault, software fault and balloon damages. In this model, if a balloon is faulty, then the other balloons in the communication range could not receive any beacon messages, which they receive periodically. The fault status of the neighbor faulty balloon is shown by *F*: faulty, else *NF*: not faulty with probability 1.

However, if it receives beacons after some time from a neighbor node, then it can set the fault status to *F*/*NF* with a probability of 0.5.

C. Working of Routing Mechanism

From the Fig. 1, it is observed that a balloon network is set at the stratosphere to explain the proposed routing scheme. The network consists of six balloon nodes, namely *B*1, *B*2, *B*3, *B*4, *B*5 and *B*6, where *B*5 is a faulty node in the network. A user requests to access high speed Internet through the balloon network by sending a request message from the specialized antenna. The message is first received at the *B*1 node. The *B*1 nodes check the neighbor table shown in Table I to find its neighbors in the network. It finds *B*2 and *B*4 are the neighbors. Then it checks whether they are fault free or not from its fault status table shown in Table II. If the nodes are fault free, then it finds the fault free node which is farther in the network by using the Euclidean distance *D* shown in Eq. (1).

$$D = sqrt((x2 - x1)^2 + (y2 - y1)^2 + (z2 - z1)^2) \tag{1}$$

Then the data is forwarded to the farthest node in the network using Eq. (2).

$$\text{Next of } B = max(D_{B2}, D_{B4}) \tag{2}$$

This process continues until it reaches to the ISP. From this example, the message is forwarded through USER-*B*1-*B*4-*B*3-*B*6-ISP. The node *B*5 is not chosen as it is faulty. Algorithm 1 (Fig. 2) shows the proposed routing mechanism.

TABLE I. Neighbor Node Status in Communication Range

	B1	*B2*	*B3*	*B4*	*B5*	*B6*
B1	0	1	0	1	0	0
B2	1	0	1	1	0	0
B3	0	1	0	1	1	1
B4	1	1	1	0	1	0
B5	0	0	1	1	0	1
B6	0	0	1	0	1	0

TABLE II. Faulty Node Status in Communication Range (*F*: Faulty, *NF*: Not Faulty)

	B1	*B2*	*B3*	*B4*	*B5*	*B6*
B1	0	*NF*	0	*NF*	0	0
B2	*NF*	0	*NF*	*NF*	0	0
B3	0	*NF*	0	*NF*	***F***	*NF*
B4	*NF*	*NF*	*NF*	0	***F***	0
B5	0	0	*NF*	*NF*	0	*NF*
B6	0	0	*NF*	0	***F***	0

Algorithm: Item-Oriented CF (ICF)

Input: 1) Node id
2) Node coordinate
3) Faulty node status
4) Not faulty node status
5) n: number of balloons

Output: 1) Detection accuracy
2) Delay

1. **for** $i = 1{:}n$
2. **for** $j = 1{:}n$
3. Find D_{i-j} matrix
4. Find fault status matrix
5. **endfor**
6. **endfor**
7. **for** k = user:ISP
8. user = nearer B // sends request message
9. B finds next of B // Eq. (2)
10. B = next of B // sends request message
11. **endfor**
12. $Delay = (td + pd + qd + pd)_{n1-n2} + (td + pd + qd + pd)_{n2-n3} + (td + pd + qd + prd)_{nn-1-nn}$
// td: transmission delay, pd: propagation delay, qt: queuing delay, prd: processing delay
13. $Detection\_Accuracy$ = Number of faulty nodes detected / Total number of faulty nodes

Fig. 2. Fault detection based routing scheme.

IV. Simulation and Results

The performance of the proposed scheme is evaluated by running simulation in MATLAB R2015b. The proposed method is compared with a shortest path routing scheme. The performance is evaluated by considering the following parameters, namely fault detection accuracy, end to end delay and false alarm rate. The number of balloons taken in the simulation is 40. The source node is taken as 1 and the destination node is taken as 1. The simulation is performed in a 3-D three dimensional area of size 100 m × 100 m × 100m (length (l) × breadth (b) × height (h)). The 3-D coordinates of the balloons are randomly deployed in this area. The request packet size is considered to be 100 bytes. The data sending rate is 3 Mbps. The balloon, which is farthest in the communication range, is selected as the next forwarder node. The number of faulty nodes is varied for the performance evaluation. The simulation is carried out for 10 simulation runs and the average data is considered for evaluation. The source and the destination nodes are randomly selected. The simulation parameters are shown in Table III.

TABLE III. Simulation Parameters and Their Values

Sl. No.	Parameter	Values
1	Number of balloons	40
2	Source (user node)	1
3	Destination (ISP node)	1
4	Faulty balloons with probability 1 or 0.5	5% to 25% of n
5	Data rate	3 Mbps
6	Message request size	100 bytes
7	Simulation area ($l \times b \times h$)	100 × 100 × 100
8	Number of simulation runs	10
9	Communication range	20 m

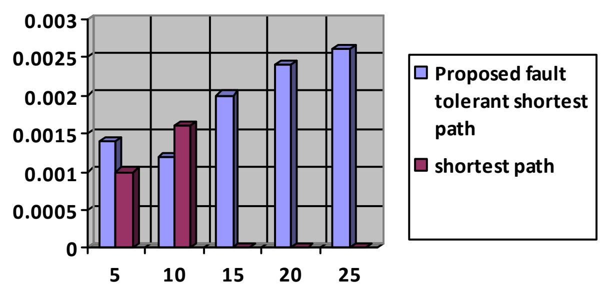

Fig. 3. Comparison of end to end delay (x-axis indicates faulty balloons percentage and y-axis indicates delay in seconds).

Fig. 3 shows the end to end delay comparison for the existing shortest path method and the proposed fault tolerant routing method. From the figure, it is observed that the shortest path method is not able to deliver the request message to the ISP node, when the faulty balloons in the network increase (15% to 25%). The shortest path routing method is unable to deliver the message, because when the data is sent to the forwarder, the forwarder node is dead and we assume that if a node is faulty, then from that route the data is not delivered.

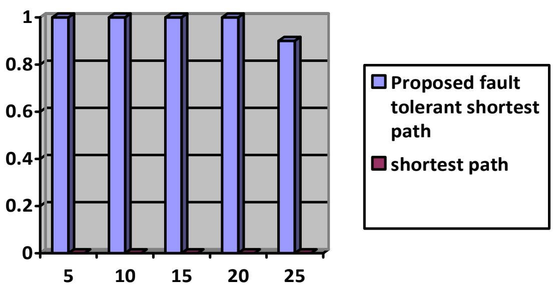

Fig. 4. Comparison of detection accuracy rate (x-axis indicates faulty balloons percentage and y-axis indicates fault detection accuracy between 0 and 1).

From Fig. 4, it is observed that when the fault percentage of balloons is 5%-20%, then the detection accuracy of the proposed routing scheme is 100%, whereas when the fault percentage is 25%, the detection accuracy is 90%, because the data is forwarded through those 10% nodes, which has fault probability of 0.5. The shortest path routing scheme has no fault detection mechanism. Therefore, the detection accuracy is zero.

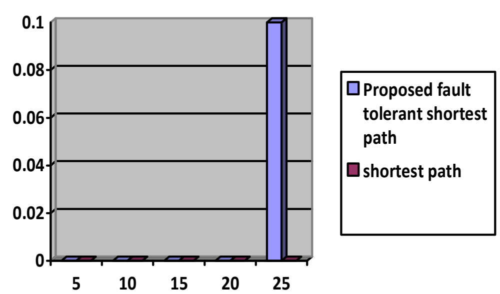

Fig. 5. Comparison of false alarm rate (x-axis indicates faulty balloons percentage and y-axis indicates fault alarm rate between 0 and 1).

From Fig. 5, it is observed that when the fault percentage of balloons is 5%-20%, then the false alarm rate of the proposed routing scheme is zero, because all the nodes are correctly detected. However, when the fault percentage is 25%, the false alarm rate is 10%, because the data may be sent through those 10% nodes which has fault probability of 0.5. The shortest path routing scheme has no fault detection mechanism. Therefore, the false alarm rate is zero. From the above two schemes, it is better to use the fault tolerant routing scheme by sending data (request message) in a fault free path.

V. Conclusion

In this work, a fault detection based routing scheme has been presented to send the data in fault free path. The performance of the proposed scheme over the loon project was evaluated using the simulations. From the simulation, it was observed that the data is forwarded to the ISP in a fault free path with high detection accuracy and in a shortest time. The comparison results have shown that the shortest path algorithm with no fault detection mechanism fails to deliver the data. This routing scheme will be a better routing method to send data in the fastest manner.

References

[1] S. Katikala, “Google Project Loon”, InSight: Rivier Academic Journal, Vol. 10, No. 2, pp. 1-6, 2014.

[2] K. Gayashani, S. Indradasa, K. Kahatapitiya, S. Umaraji, S. Seram and D. Dhammearatchi, “Loon Ballon Plus: Future Implementations of Google Loon Project”, Imperial Journal of Interdisciplinary Research, Vol. 5, pp. 993-997, 2016.

[3] B. Moision, B. Erkmen, E. Keyes, T. Belt, O. Bowen, D. Brinkley, P. Csonka, M. Eglington, A. Kazmierski, N. Kim and J. Moody, “Demonstration of Free-Space Optical Communication for Long-Range Data Links Between Balloons on Project Loon”, Free-Space Laser Communication and Atmospheric Propagation XXIX, International Society for Optics and Photonics, Vol. 10096, 2017.

[4] M. Darokar and D. Astonkar, “Provide High Altitude Network by Using Project Loon”, International Research Journal of Engineering and Technology, Vol. 4, No. 1, pp. 659-662, 2017.

[5] L. Nagpal and K. Samdani, “Project Loon: Innovating the Connectivity Worldwide”, 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, pp. 1778-1784, 2017.

[6] S. Bhatt and K. Ambekar, “Google Loons”, International Journal of Computer Engineering and Technology, Vol. 7, No. 4, pp. 59-66, 2016.

[7] J. Burr, “The Feasibility of Google’s Project Loon”, Australia National University, 2017.

[8] M. Caswell, “Need for Vertical Delineation of Air Space: Can Google’s Project Loon Survive Without It”, Tulane Journal of International and Comparative Law, Vol. 24, pp. 205, 2015.

[9] C. Ahn, “An Applicable 5.8 GHz Wireless Power Transmission System With Rough Beamforming to Project Loon”, ICT Express, Elsevier, Vol. 2, No. 2, pp. 87-90, 2016.

[10] C. Wang, C. Shi and H. Zhang, “Project Loon Based Augmentation for Global Ionospheric Modeling Over Southern Hemisphere”, Scientific Reports, Nature, Vol. 7, pp. 45976, 2017.

[11] D. Revar, S. Mishra, S. Shinde, V. Adkar and A. Somnathe, “Case Study on Project LOON”, 2019.

[12] B. Barritt and V. Cerf, “Loon SDN: Applicability to NASA’s Next-Generation Space Communications Architecture”, IEEE Aerospace Conference, pp. 1-9, 2018.

[13] M. Kuvalekar, “Comparative Study of Project Loon and Wi-Fi”, International Research Journal of Engineering and Technology, Vol. 3, No. 6, pp. 1908-1910, 2016.

[14] S. Kaur and S. Randhawa, “Google LOON: Balloon-Powered Internet for Everyone”, AIP Conference Proceedings, Vol. 2034, No. 1, pp. 20006, 2018.

[15] S. George and K. Babu, “Anywhere at Anytime Internet: Google Loon Balloons”.

[16] S. Kapri and P. Singh, “Internet Penetration and Google Loon as a Last Mile Solution”, International Journal of Aerospace and Mechanical Engineering, Springer, Vol. 3, pp. 41-45, 2017.

[17] R. Pravinkumar, I. Varalakshmi and S. Jayamoorthy, “LOON-A Wireless Network Communication for Providing Fast Internet Access”.

[18] P. Jiang, “A New Method for Node Fault Detection in Wireless Sensor Networks”, Sensors, Vol. 9, No. 2, pp. 1282-1294, 2009.

[19] A. Moustapha and R. Selmic, “Wireless Sensor Network Modeling Using Modified Recurrent Neural Networks: Application to Fault Detection”, IEEE Transactions on Instrumentation and Measurement, Vol. 57, No. 5, pp. 981-988, 2008.

[20] M. Yu, H. Mokhtar and M. Merabti, “Fault Management in Wireless Sensor Networks”, IEEE Wireless Communications, Vol. 14, No. 6, pp. 13-19, 2007.

[21] B. Lau, E. Ma and T. Chow, “Probabilistic Fault Detector for Wireless Sensor Network”, Expert Systems with Applications, Elsevier, Vol. 41, No. 8, pp. 3703-3711, 2014.

[22] J. Chen, S. Kher and A. Somani, “Distributed Fault Detection of Wireless Sensor Networks”, Workshop on Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks, ACM, pp. 65-72, 2006.

[23] L. De Souza, H. Vogt and M. Beigl, “A Survey on Fault Tolerance in Wireless Sensor Networks”, Interner Bericht. Fakultat fur Informatik, Universitat Karlsruhe, 2007.

[24] C. Titouna, M. Aliouat and M. Gueroui, “FDS: Fault Detection Scheme for Wireless Sensor Networks”, Wireless Personal Communications, Springer, Vol. 86, No. 2, pp. 549-562, 2016.

[25] A. Mahapatro and P. Khilar, "Fault Diagnosis in Wireless Sensor Networks: A Survey", IEEE Communications Surveys and Tutorials, Vol. 15, No. 4, pp. 2000-2026, 2013.
[26] S. Panda, P. Khilar and D. Mohapatra, "FTMXT: Fault Tolerant Immediate Mode Heuristics in Computational Grid", International Conference on Informatics and Communication Technologies for Societal Development, Springer, pp. 103-113, 2014.
[27] H. Alwan and A. Agarwal, "A Survey on Fault Tolerant Routing Techniques in Wireless Sensor Networks", Third International Conference on Sensor Technologies and Applications, IEEE, pp. 366-371, 2009.