EFFECTIVE LOAD BALANCING WITH POWER CONSERVATION IN RFID

Vijayakumar.G.Dhas[1], Ramanathan Muthukaruppan[2], Konguvel Balakrishnan[2], Rajarajan Ganesan[2]

[1]Lecturer, Department of Information Technology, Anna University, Chennai
vgdhas@annauniv.edu
[2]Department of Information Technology, Anna University, Chennai
mraman2006@gmail.com, konguvel31@gmail.com, rajarajan248@gmail.com

ABSTRACT

Radio Frequency Identification (RFID) comprises of uniquely identifiable, less expensive tags and readers that monitor these tags through Radio Frequency signals. The information in the tags will be collected with the help of the readers. The load for any reader is the number of tags that the reader has to monitor. For the effectiveness of the RFID system several algorithms like Load Balancing algorithm and Redundant Reader Elimination algorithm were proposed. In these existing schemes the former concentrates on the load of the reader while the latter concentrates on reducing the power consumption by the RFID system. A solution for improving the effectiveness of rfid system is proposed. The system effectiveness can be improved by balancing the load among the readers and turning off readers that do not monitor tags in order to reduce power consumption. System will find if any reader goes down and will wake up the readers that are in sleep state to monitor the tags if any are present in that failed reader's range. If still the tags are not monitored after wake up of redundant readers, the system will prompt for the replacement of that reader.

KEYWORDS

Load Value, Load balancing, Redundant Reader, Timestamp.

1. INTRODUCTION

Radio Frequency Identification (RFID) system consists of Tags and Readers. Tags are uniquely identifiable. Tags can be associated with an object in order to identify the object. Tags can be active or passive. Passive tags do not have any power source. They generate power from the signal sent by the Reader. Active tags have their own power source.

Each tag has memory to store a small amount of data in it. Reader can issue a write signal to store the data in the tag. Reader can monitor the tags that are present in its reading range. Reader periodically reads tags in its vicinity. If there is more number of tags present in its vicinity, then energy of that reader will be depleted more. Also, there is a less probability to monitor all the tags in its vicinity, within the next period. Hence, it becomes necessary to balance the load on each reader.

In the Load balancing scheme in large-scale RFID systems proposed by Q.Dong et. al [3], the tags are allocated to the readers in a fair manner such that no reader in the entire system is allocated to more number of tags. When a set of tags are within the range of each reader, which of these tags should each reader monitor, such that the cost of monitoring tags across the different readers is balanced. Each tag is monitored by at least one reader. They have proposed the localized probabilistic assignment (LPA) scheme, for finding a tag-driven probabilistic assignment of tags to readers. In this scheme, each tag knows which readers are in its vicinity

DOI : 10.5121/iju.2010.1402

and what the load on the readers is and each reader only knows which tags are in its vicinity and its load. In order to achieve a more load balanced assignment, each tag should decide its probability of reporting to some reader based on the load. If a reader in its vicinity has a relatively high load (compared with other readers in its vicinity), the tag should report to it with a relatively low probability. Subsequently, in the LPA scheme, each tag will only consider reporting to its candidate readers instead of all readers that can cover it with maximum transmission range. The candidate readers of a tag are the readers that can reach that tag at the minimum transmission power level.

After each round of data retrieval, each reader and tag automatically obtains up-to-date knowledge about its vicinity. If a reader or tag leaves the system, it will be automatically detected at least after the next round of data retrieval. Therefore, no additional processing is needed to handle reader/tag leaves.

In this scheme, the load gets distributed evenly. There is a possibility for readers to get assigned only one tag each. This leads to more amount of energy depletion. If there are redundant readers, then energy can be saved by switching off redundant readers. Load balancing in large scale RFID systems, follows a tag driven probabilistic assignment of tags to readers. Tag should decide the probability with which to report to the reader. It reports with a low probability to highly loaded reader and with high probability to lightly loaded reader. Tag driven approach leaves more processing on the tag.

The Redundant Reader Elimination in RFID Systems proposed by B.Carbunar et. al [2] addresses the energy conservation by putting redundant readers to sleep state. Redundant readers cover a set of RFID tags which are also in the reading range of other RFID readers. In order to maximize the number of RFID readers that can be simultaneously deactivated, the minimum number of readers that cover all RFID tags needs to be discovered. Redundant Reader Elimination (RRE) consists of two steps. In the first step, each RFID reader attempts to write its tag count (number of covered tags) to all its covered RFID tags. An RFID tag stores the highest value seen and the identity of the corresponding reader. In the second step, an RFID reader queries each of its covered RFID tags and reads the identity of the tag's holder. A reader that locked at least one tag is responsible for monitoring the tag and will have to remain active. However, a reader that has locked no tag can be safely turned off. This is because all the tags covered by that reader are already covered by other readers that will stay active. Each tag is locked by the reader in its vicinity that covers most tags. A reader that locks at least one tag is required to remain active.

There is a possibility for a single reader to get overloaded and all other readers having no load. This results in poor performance. As the tags move in a random manner, the topology of the network changes frequently. When a tag moves from one reader's vicinity to another, then the tag has to be disassociated from the previous reader and it has to be associated with the new reader. In RRE scheme, a tag is allocated to the reader only if the reader has a load value higher than the one that is already in the tag's memory. When a tag moves from a reader's vicinity whose load is 5 to a new reader's vicinity whose load is 3, the new reader will compare its load value with the value in tag which is 5. So, it does not monitor the tag, even though the tag is not monitored by the old reader. To prevent this, the values in the tag are reset at periodic intervals. This interval is determined based on the number of tags that a reader monitors. But, there is a possibility for a tag to go blind until that interval.

2. PROPOSAL

The tags considered are Passive tags. Passive tags have no processing power, i.e. they have no external power source. They generate power from the signal sent by the reader. The tag then sends the information present in its memory. Since the tag is much less expensive and requires no external power source, the tag finds extensive applications. Readers retrieve the information from the tag and decide whether to monitor the tag or not. It is assumed that the Tag collision and reader collision are avoided by RCA algorithm as proposed by B.Carbunar. et al.[2].

The tags are allocated to the readers based on the algorithm described in section 3.1. Tags are initially assigned to the reader which sends the signal first. When a reader's signal reaches the tag, the tag transmits its information to the reader. The reader decides whether to monitor the tag or not based on the algorithm. When a reader decides not to monitor the tag, then the number of tags in its vicinity is reduced by one. When a reader decides to monitor the tag, then it updates the reader id, load value and timestamp values in the tag. Reader updates the entry when the number of tags in its vicinity is more than the load value in the tag and it is lesser than the threshold value. This ensures that a reader does not monitor the tag, which is monitored by another reader, if it is going to be overloaded by monitoring this tag.

Readers that do not monitor any tags are found out by using the algorithm as described in section 3.2. This helps in power conservation by turning off the unwanted readers. Each tag will contain the load of the reader that monitors it and the reader ID. Every reader checks the tags in its vicinity to identify the tags allocated to it. If no tags are allocated to it, then Reader is switched off. This leads to conservation of energy. Redundant readers that are turned off can be turned on, when a reader in the system gets overloaded in order to share the load of the overloaded reader. In this solution for RFID load balancing, when a tag moves from one reader's vicinity to another then the information in the tag will not be considered because it is written by the reader to which it was connected before. Timestamps are used to find out whether the tag has been probed within a certain time period. This ensures that each tag is monitored by at least one reader and prevents the tag from going blind.

If any reader fails or malfunctions, then the system will try to prevent the tag from going blind. This is explained in section 3.4. The system will prompt for either the replacement of that reader or try to monitor the tags by other readers present in the system.

3. ALGORITHM

3.1. Algorithm for allocation of tags

Threshold value= NT/NR + NR/2;

For each reader in the system

{

Find out the number of tags in the reader's vicinity(NOTV)

For each tag in the vicinity

{

Check the timestamp of tag.

If timestamp=NULL then

```
            Create an entry in the tag's memory with the current reader's NOTV value, id,
            and timestamp
    Else if timestamp not old then
            LV=Load value in the entry of the tag
            If LV > Threshold Value and NOTV < Threshold value then
                Update the tag's entry with current reader's NOTV value, id and update the
                timestamp
            Else if LV < Threshold value and NOTV < Threshold then
            value and NOTV> LV)
                  Update the tag's table entry with current reader's NOTV value, id and
                  update the timestamp
            Else
                  Don't update
                  Reduce the NOTV of current reader by one.
            End if
    Else
            Update the tag's entry with current reader's value
    } //for each tag in the vicinity
    }
```

The number of readers and the number of tags that is present in the RFID system are noted. Threshold value is calculated as,

Threshold value = (Number of tags / Number of readers) + (Number of readers / 2)

Each reader finds out the number of tags that are present in its vicinity, which will be denoted as NOTV. Readers send signal to tags in its vicinity. Reader checks the timestamp present in the tag. If the timestamp is null, then the tag is yet to be probed by the reader. Hence, the reader issues a write command to the tag to store its identity and NOTV. Similarly, for all the yet to be probed tags present in its vicinity the reader issues a write command.

When the timestamp is not null, the reader checks the whether the tag has been probed before within a certain period. If it has been probed within the certain time interval, then the reader decides whether to monitor the tag or not based on three criteria.

Criteria 1:
When the load value present in the tag, i.e. the NOTV value written by the reader which previously monitored it, is greater than the Threshold value calculated as above and the NOTV value of current reader is lesser than the Threshold value, then the current reader will monitor it.

Criteria 2:
When the load value present in the tag is lesser than the Threshold value and the NOTV value of the current reader is lesser than the threshold value and NOTV value of the current reader is greater than the Load value of the tag, then the current reader will monitor the tag.

Criteria 3:
If the above two criteria's are not satisfied then, reduce the NOTV value of the reader by one and tag will be monitored by the previous reader itself.

When the tag has not been probed for a specified time interval, then the tag is no longer in the range of the previous reader. Hence, the new reader monitors the tag. After the first period, the readers that are in on state, share the load of other readers if there are tags present in the intermittent range. This enhances balancing of the load among readers.

3.2. Algorithm for turning off redundant readers

```
For each reader in the system
{
Initialize Flag for Sleep = 0;
Find out the number of tags in the reader's vicinity (NOTV)
For each tag in the vicinity
{
Read the reader name in the entry of the tag (RN)
If RN=current Reader then
        Flag for Sleep =1
        Monitor the tag
        Update the Server table (i.e., increment the number of
        tags monitored by the corresponding reader)
Else
        Continue to check other tags
End if
} // for each tag in the vicinity
If Flag for sleep = 0 then
        Put the reader into sleep state
        Update the Number of tags monitored column in server
        table for that reader to 0.
End if
} //for each reader in the system
```

During the next cycle, the reader checks whether any of the tags in its vicinity has its id stored in it. If no tag has its id stored, then the reader has no tags to monitor. Hence, the reader can be turned off. In this way, energy can be conserved by turning off redundant readers.

3.3. Wake up criteria

If during next cycle, new tags are introduced, then there is a possibility for readers to get overloaded. During that time, the redundant readers which were put to sleep are turned on. Overloaded reader creates a trigger that enables the redundant readers to be turned on. If the overloaded reader has tags that are also in the reading range of redundant readers, then the redundant reader shares the load of the overloaded reader.

3.4. Reader Failure

Every reader will report to the system, the information about the tags. If a reader malfunctions, then that reader will fail to report to the system. System will use this to detect the reader failure. If a reader does not report for the next two cycles, then that reader is assumed to have failed. In that case, the system will wake up readers that are in sleep state, if any. The readers that are waked up will share the load, if the region of the failed reader overlaps with it. Still, if there are tags that are not monitored, then the system will prompt for the replacement of that reader. System will find this case, by comparing the number of tags that were reported by the failed reader with the tags monitored by the waked up readers. If they match, then tags will not go blind. Else, the tags will not be monitored. In the later case, the reader needs to be replaced immediately.

4. EVALUATIONS AND RESULTS

Consider the scenario as given in Figure 1.

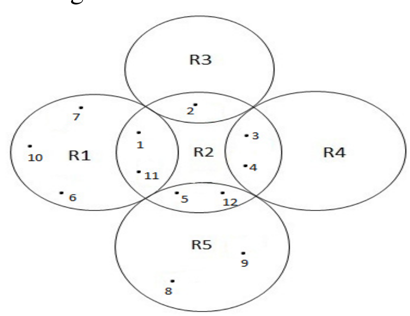

Figure. 1. Scenario containing 5 readers and 12 tags

As per the proposed solution, threshold value is calculated as 4. There are totally 12 tags and 5 readers in the system. Reader R1 monitors 3 tags, R2 monitors 3 tags, Reader 4 monitors 2 tags and R5 monitors 4 tags. R3 is turned off, since it is a redundant reader, i.e. tags in its vicinity are monitored already by another reader. Thus tags are equally distributed among the readers, with no reader overloaded.

For the scenario in Fig 1, as per the load balancing scheme the tags will be allocated as follows: R1 monitors tag t1, t6, t7, t10, t11. R2 monitors t5, t12. R3 monitors tag t2 and R4 monitors t3, t4. R5 monitors t8, t9.

In RRE scheme the readers R3 and R4 are found as redundant readers since R2 is found as the minimum set of reader that can monitor all the tags (t1–t5, t11, t12) in the scenario given in Figure 1. R1 monitors t6, t7, t10. R5 monitors t8, t9.

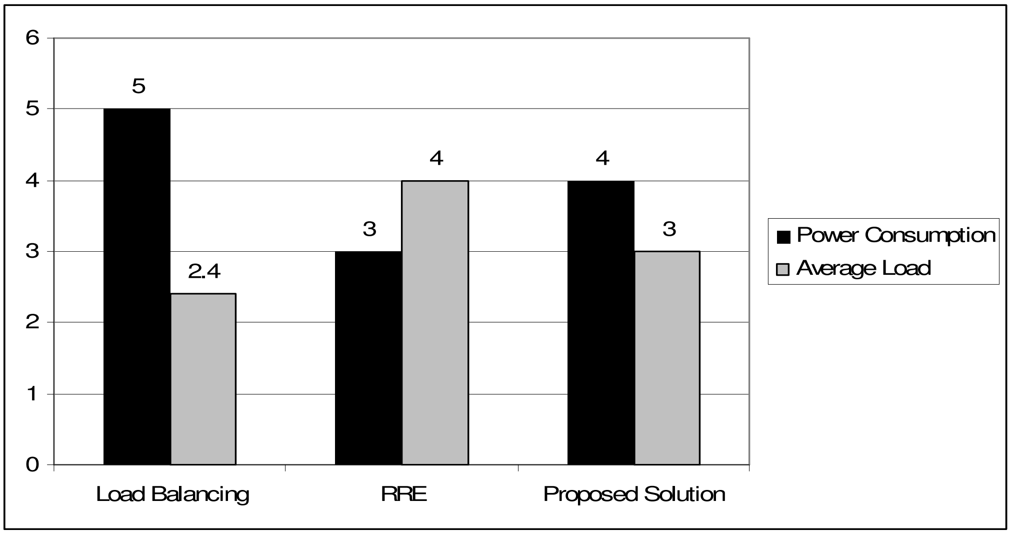

Figure. 2. Comparison of three schemes for the scenario shown in Figure 1.

For the scenario as shown in figure 1, the performance of Load Balancing, RRE and proposed solution is compared in figure 2. As the figure shows, proposed solution allocates tags to readers such that threshold value is not exceeded.

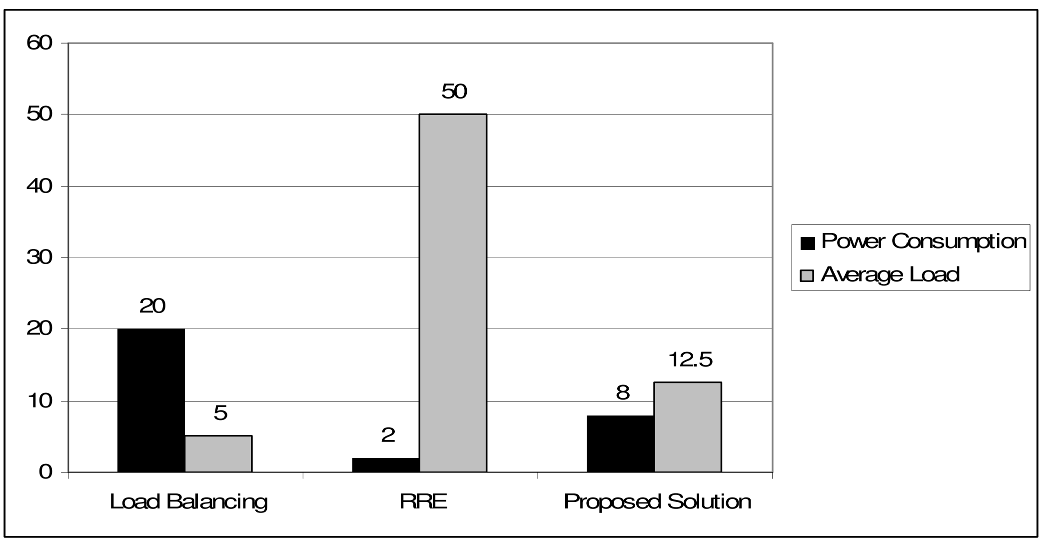

Figure. 3. Comparison of three schemes for the scenario consisting of 20 readers and 100 tags

The performance of the three schemes for complex RFID system consisting of 20 readers and 100 tags is shown in the figure 3. It is observed from the figure that RRE scheme is having the least power consumption, but the average of the readers get increased to a large extent. Proposed solution yields good result when compared to load balancing and RRE schemes.

5. CONCLUSIONS

In this paper, balancing tags among the readers, switching off the redundant readers to conserve power are addressed. Effective Load Balancing with power conservation in RFID performs effectively better. Maximum number of readers that can be switched off are found out and turned off. This reduces the power consumption. Tags in the intersection range of two or more readers are balanced among readers using algorithm proposed in section 3.1. This scheme can perform effectively even in highly dynamic RFID systems. When a reader fails or does not function, then the tags may not get monitored. This situation is handled either by waking up redundant readers or by replacing the faulty reader. Mobility of tags is taken care by using timestamp in tags.

REFERENCES

[1] Y. Bejerano, S.-J. Han, and L. E. Li. "Fairness and load balancing in wireless lans using association control". In ACM MobiCom, 2004.

[2] B. Carbunar, M. K. Ramanathan, M. Koyuturk, C. Hoffmann, and A. Grama. "Redundant-reader elimination in rfid systems". In IEEE SECON*, 2005.*

[3] Q. Dong, A. Shukla, V. Shrivastava, D. Agrawal, S. Banerjee,and K. Kar. "Load balancing in large-scale rfid systems",INFOCOM 2007.

[4] Daniel W. Engels, Sanjay E. Sarma, Stephen A. Weis, and. RFID systems and security and privacy implications. In *CHES '02*, pages 454–469. Springer-Verlag, 2003.

[5] C. Floerkemeier and M. Wille, "Comparison of Transmission Schemes for Framed ALOHA based RFID Protocols", Workshop on RFID and Extended Network Deployment of Technologies and Applications, Phoenix, AZ. Jan 2006.

[6] Jian Feng Cui,Heung Seok Chae, "Mobile Agent based Load Balancing for RFID Middlewares", ICACT '07, Feb2007.

[7] Mamoru Kobayashi, Bin Zhen, Masashi Shimizu. "To read transmitter-only RFID tags with confidence", IEEE 2004.

[8] Raja Jurdak, Antonio G. Ruzzelli, Gregory M.P.O'Hare. "On the RFID wake-up impulse for multihop sensor networks".

[9] Rong Zheng, Jennifer C. Hou, and Lui Sha. "Asynchronous wakeup for ad hoc networks". In MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing, pages 35–45. ACM Press, 2003.

[10] Shweta Jain, Samir R. Das . "Collision Avoidance in a Dense RFID Network". *WiNTECH'06, September 2006.*

[11] Sungjun Kim, Sangbin Lee, Sunshin "An Reader Collision Avoidance Mechanism in Ubiquitous Sensor and RFID Networks", In WiNTECH'06, September 29, 2006, Los Angeles, California, USA. ACM 1-59593-538-X/06/0009.

[12] F. Zhou, C. Chen, D. Jin, C. Huang and H. Min, "Evaluating and optimizing power consumption of anti-collision protocols for applications in RFID systems", ACM ISLPED '04, Aug 2004.

Authors

Vijayakumar G.Dhas

He finished his B.E in 1984. He did his M.E in 2005. He is currently pursuing his Ph.D in Anna University Chennai. His interests include Networking, Web Servers, RFID and Load Balancing.

Ramanathan.M

He did his B.Tech in Information Technology in Madras Institute of Technology Campus, Anna University, Chennai.

Konguvel.B

He did his B.Tech in Information Technology in Madras Institute of Technology Campus, Anna University, Chennai.

Rajarajan.G

He did his B.Tech in Information Technology in Madras Institute of Technology Campus, Anna University, Chennai.