POLITECNICO DI TORINO
Repository ISTITUZIONALE

Introducing probability in Colorwave

Original
Introducing probability in Colorwave / Gandino, Filippo; Ferrero, Renato; Montrucchio, Bartolomeo; Rebaudengo, Maurizio. - ELETTRONICO. - (2011), pp. 1-11.

Availability:
This version is available at: 11583/2423978 since:

Publisher:

Published
DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

24 September 2023

Introducing probability in Colorwave

Filippo Gandino, Renato Ferrero,
Bartolomeo Montrucchio, Maurizio Rebaudengo
Dipartimento di Automatica ed Informatica
Politecnico di Torino, Italy

{*filippo.gandino, renato.ferrero,*
bartolomeo.montrucchio, maurizio.rebaudengo}*@polito.it*

Technical Report n. 01-2011-UC

Abstract

The RFID technology is affected by the reader-to-reader collision problem. Several studies proposes anti-collision protocols. A high throughput adaptation protocol based on time division is Colorwave. A known mechanism, already applied to another time division protocol to improve its performance, is the introduction of a probability factor in the collision resolution routine. In this work, the effects of the introduction of the probability factor in Colorwave are analyzed.

1 Introduction

The Radio Frequency Identification (RFID) is a well known technology for auto-identification. Several applications, from health care [1] to traceability management [2], use many RFID readers in the same area. The presence of readers working close together involves the reader-to-reader collision problem [3].

In order to solve the reader-to-reader collisions several appraoches have been proposed [4]. An high throughput protocol based on time division is Colorwave [5, 6]. It is a distributed adaptable protocol, where each RFID reader periodically try at queering tags, and modify its parameters according to the quantity of collisions. Colorwave is based on DCS [5, 6]. The main difference between these protocols is that DCS does not involve adaptive parameters. When the density of the network is unknown, Colorwave provides a higher throughput than DCS, since each reader modifies its parameters according to the quantity of collisions.

In [7], the introduction of a probabilistic factor in the collision resolution routine of time division protocols has been proposed in order to improve their performance. Probabilistic DCS (PDCS) is a new protocol proposed in [8], which employs the probabilistic factor and is based on DCS.

This work analyzes the introduction of the probabilistic parameter in Colorwave. A new version of the protocol, compliant with this technique, is presented and evaluated.

The remainder of the paper is organized as follows: in Sect. 2 related works are described. In Sect. 3 the proposed scheme is presented. Finally, in Sect. 4, the analyzed approach is evaluated, and in Sect. 5 some conclusions are drawn.

2 Related Works

This section describes the main relevant anti-collision protocols and their requirements.

2.1 Distributed Color System (DCS)

In DCS [5, 6] each communication round is composed of time slots. Each RFID reader can communicate only during its time slot. When a transmission collides, each involved reader stops the communication and randomly chooses a new timeslot that it has reserved, sending a specific signal named *kick*. When the reserved slot is used by some neighbors, they randomly choose a new slot and try using it without reservation.

The communications are divided in rounds. Each round is composed of μ timeslots. Each timeslot is composed of a kick phase and a transmission phase. The identification of a timeslot is called *color*. A color is assigned to a reader, and it works only during the corresponding time slot.

During the kick phase, each working reader that had collided at the previous transmission sends a kick. Each working reader that receives a kick changes color.

During the transmission phase, each working reader that has to read tags executes a transmission. If the transmission collides then the involved readers stop it and randomly choose a new color. At the subsequent round the colliding readers will send a kick, in order to reserve a timeslot.

In the described protocol when more than one reader transmits a kick during the same slot, all the transmitting readers also receive the kick, and they choose a new color.

The kick does not transport any additional information, but is used only to communicate to the neighbors that the channel is busy, so the readers do not need additional hardware. The only additional requirement is represented by the global synchronization, since each reader must initiate a new timeslot simultaneously to its neighbors.

2.2 Colorwave

Colorwave [5, 6] is a protocol based on DCS. This protocol introduces a variable quantity of timeslots that compose a round (μ), differently from DCS where the number of timeslots is fixed. The value is dynamically changed in order to increase the efficiency of the RFID network. When the number of collisions is high, the number of used colors for round rises, while when it is small the number of colors decreases. This protocol requires a special kick transmission, which states the change to a new μ. The kick phase is divided in two subphases, where normal kicks are sent during the first one, and color kicks during the second one.

In order to manage changes of μ, Colorwave introduces two couples of thresholds: one is used to manage the increase of μ, and the other for the decrease. Each couple is composed of a hard threshold, which sets a change, and a soft threshold, which sets a transition state, where the reader changes μ only if a neighbor is already changing. Therefore, each reader counts its percentage of successful transmissions. When the percentage exceeds a hard threshold the reader changes μ and communicates the change to its neighbors during the second kick subphase. If a reader has exceeded a soft threshold and it receives colorkick compliant with the exceeded threshold, then it changes μ and communicates the change to its neighbors.

The adaptable μ allows Colorwave to autonomously find a good configuration. However, in addition to the requirements of DCS, this protocol also requires to manage the special color kicks.

The presence of neighboring readers with different μ can generate additional collisions, as at each round different couples of slots overlap. However, this problem is in part overcome by the improved efficiency introduced by the adaptable μ.

2.3 Probabilistic DCS (PDCS)

Probabilistic DCS (PDCS) [8] is an improved version of DCS. This protocol introduces a new parameter p, which represents the probability to change color after a collision. The goal of this protocol is to reduce the number of

Initialization()

Color$_i$ = (Color$_i + 1$)mod(μ_i)
TimeInColor$_i$ = TimeInColor$_i + 1$;
if reader$_i$ received a request to read tags then
  TransFlag$_i$ = true;
end if
if CTrans$_i$.getCollisions() $>$ UpSafe
AND TimeInColor$_i$ $>$ MinTimeInColor then
  $\mu_i = \mu_i + 1$;
  CTrans$_i$.clear();
  TimeInColor$_i = 0$;
  ColorUpKickFlag$_i = \mu_i$;
end if
if CTrans$_i$.getCollisions() $<$ DownSafe
AND TimeInColor$_i$ $>$ MinTimeInColor then
  $\mu_i = \mu_i - 1$;
  CTrans$_i$.clear();
  TimeInColor$_i = 0$;
  ColorDownKickFlag$_i = \mu_i$;
end if

Fig. 1: Initialization subroutine

collisions, reducing the number of readers that randomly change color. This approach improves the performance of DCS, without additional requirements.

The analysis proposed in [8] shows that after a collision between 2 readers, the color of the collision is probably available, while the other colors could be busy. Therefore, reducing the probability of changing color after a collision, a reader could keep the same color and the other could change, reducing the probability of further collisions.

3 Proposed protocol

In order to present the algorithm, some definitions are preliminary given. The list of the variables used in the probabilistic version of Colorwave is shown in Table 1. The procedures are:

- CTrans$_i$.add() adds a new flag;
- CTrans$_i$.clear() deletes all the flags;
- CTrans$_i$.getCollisions() returns the percentage of flags equal to true;

Tab. 1: List of the variables

.

Variable	**Description**
p	probabilistic factor
UpSafe	highest threshold
UpTrigger	second highest threshold
DownTrigger	third highest threshold
DownSafe	lowest threshold
μ_i	number of used colors for reader_i
MinTimeInColor	minimum number of slots between two subsequent changes of μ_i
TimeInColor_i	number of timeslots spent with the current μ_i
CTrans_i	a First In First Out (FIFO) buffer containing the sequence of flags for the tag interrogations of reader_i among the attempted tag interrogations with the current μ; the flags are true for successful interrogations and false for colliding ones
Color_i	index of the timeslot used by reader_i to query tags
KickFlag_i	boolean flag, true when reader_i has to send a kick
ColorUpKickFlag_i	boolean flag, true when reader_i has to send a colorupkick
$\text{ColorDownKickFlag}_i$	boolean flag, true when reader_i has to send a colordownkick
TransFlag_i	boolean flag, true when reader_i has to query tags

```
Kick_sending()
if KickFlag_i = true AND Color_i = 0 then
  reader_i sends a kick;
  KickFlag_i = false;
end if
```

Fig. 2: `Kick sending subroutine.`

```
Colorkick_sending()
if ColorUpKickFlag_i != 0 true AND Color_i = 0 then
  reader_i sends a colorupkick;
  ColorUpKickFlag_i = 0 ;
end if
if ColorDownKickFlag_i != 0 true AND Color_i = 0 then
  reader_i sends a colordownkick;
  ColorDownKickFlag_i = 0;
end if
```

Fig. 3: Colorkick sending subroutine.

```
Kick_resolution()
if reader_i received a kick AND Color_i = 0 then
  CTrans_i.add(false);
  Color_i = random(μ_i − 1) + 1;
end if
```

Fig. 4: Kick resolution subroutine.

```
Colorkick_resolution()
if reader_i received a colorupkick
AND CTrans_i.getCollisions() > UpTrigger
AND TimeInColor_i > MinTimeInColor
AND ColorUpKickFlag_j > μ_i then
  μ_i = ColorUpKickFlag_j;
  CTrans_i.clear();
  TimeInColor_i = 0;
  ColorUpKickFlag_i = μ_i;
end if
if reader_i received a colorupkick
AND CTrans_i.getCollisions() < DownTrigger
AND TimeInColor_i > MinTimeInColor
AND ColorDownKickFlag_j < μ_i then
  μ_i = ColorDownKickFlag_j;
  CTrans_i.clear();
  TimeInColor_i = 0;
  ColorDownKickFlag_i = μ_i;
end if
```

Fig. 5: Colorkick resolution subroutine.

```
Transmission()
if TransFlag_i = true AND Color_i = 0 then
  the reader_i transmits;
end if
```

Fig. 6: Transmission subroutine.

```
Collision resolution()
if reader_i collides then
  if RANDOM(0,1)<p then
    Color_i = random(μ_i);
  end if
  KickFlag_i = true;
  CTrans_i.add(false);
else
  CTrans_i.add(true);
  TransFlag_i = false;
end if
```

Fig. 7: Collision resolution subroutine.

- Random(a,b) returns a random float between a and b.

In the proposed algorithm, the transmissions performed by the readers are the following:

- interrogation, when a reader communicates with tags,
- kick, when a reader sends the reservation message,
- colorupkick, when a reader send a message, communicating its new higher value of μ_i.
- colordownkick, when a reader sends a message, communicating its new lower value of μ_i.

The algorithm is iterated at each timeslot, and it is organized in three consecutive phases:

- initialization,
- kick,
- transmission.

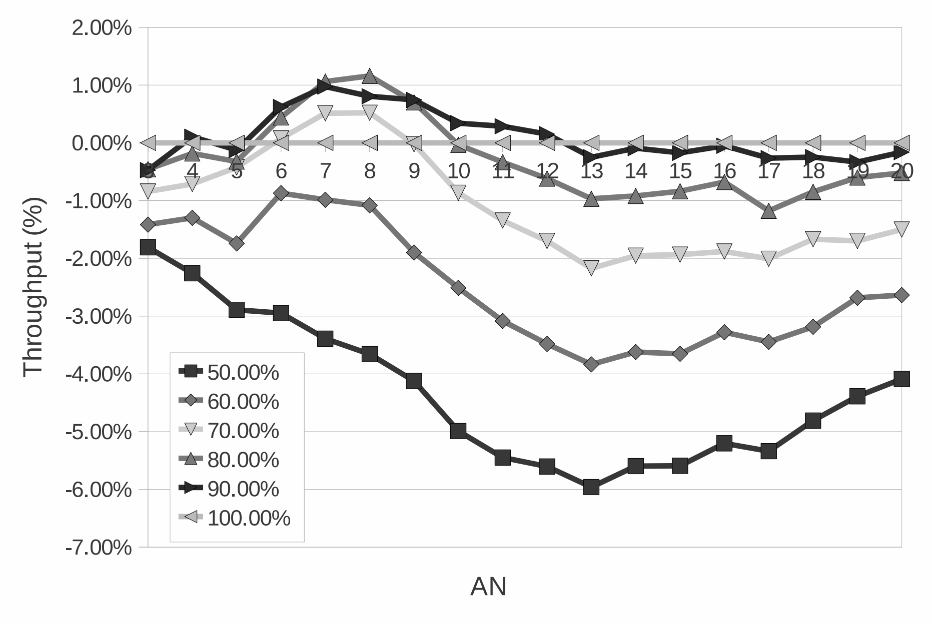

Fig. 8: Percentage Difference of PCW with respect to Colorwave using the configuration proposed in [5]

During the Initialization Phase, which is shown in Fig. 1, the readers update their variables and flags. The reading modification requests of the user are forwarded to the readers. They check UpSafe and DownSafe and eventually change μ_i. The Kick phase, which contains the Kick Sending subroutine (Fig. 2), the Kick Resolution subroutine (Fig. 4), the Colorkick Sending subroutine (Fig. 3), the Colorkick Resolution subroutine (Fig. 5), is used to reserve the current timeslot and to communicate a new μ_i. The readers that have to reserve the slot send a kick. All the readers that receive a kick randomly choose a new color, different from the previous. The readers that change μ_i send a colorupkick or colordownkick, with the new value. The Transmission Phase is used to query the tags. Each reader assigned to the current color can query tags. The colliding readers, with probability p, randomly choose a new color. Fig. 6 and 7 show respectively the Transmission and the Collision Resolution subroutine.

4 Evaluation

The effects of the probabilistic factor in colorwave have been studied with two different sets of thresholds. The former set is 93%, 90%, 2%, 1%, which corresponds to Set1 in [5]. The latter is 66%, 66%, 64%, 64%, which is

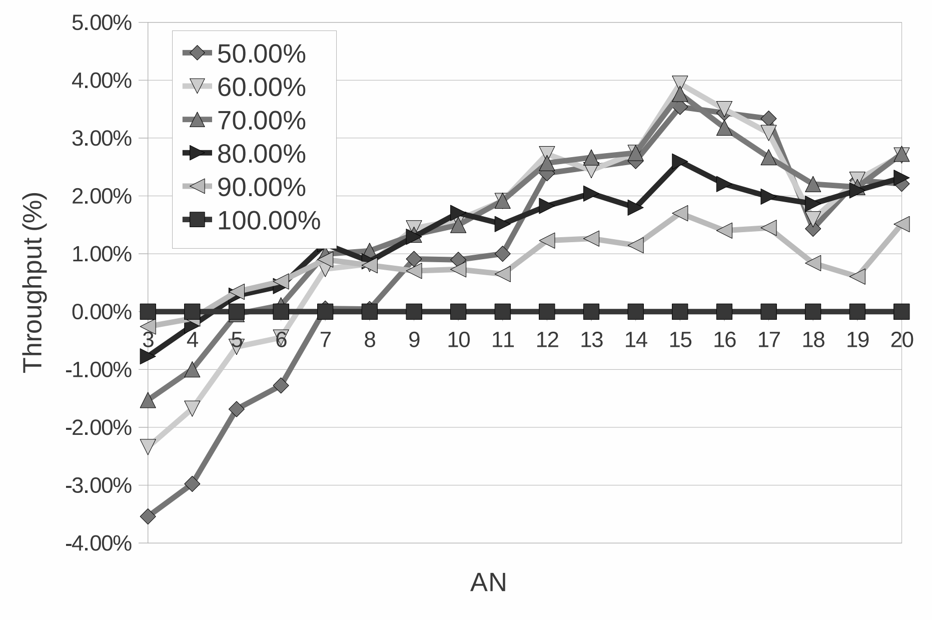

Fig. 9: Percentage Difference of PCW with respect to Colorwave using the configuration proposed in [9]

compliant with the analysis on the colorwave configuration proposed in [9]. MinTimeInColor has been set to 100 slots, and the starting μ to 6. In order to evaluate the technique, the two configurations have been simulated with networks of 250 readers with different density. All the results are calculated as the average of 5 simulations executed on 20 different deployments with the same density.

Fig. 8 shows the comparison of PCW with respect to Colorwave using the configuration proposed in [5]. The results are negative or negligible. The analyzed technique is not able to provide the same effects of DCS, since the presence of neighboring readers with a different total number of colors is too common. In fact, when two neighboring readers have a different total number of colors, at each round their relative color changes. It is equivalent to change color at each round, nullifying the effects of the probabilistic factor. Positive results are reached only when the staring color is proper for the network density, since in these networks, few readers change μ.

Fig. 9 shows the comparison of PCW with respect to Colorwave using the configuration proposed in [9]. With the configuration proposed in [9], the behavior of the readers is different from the previous. A limited number of readers reach high performance, while their neighbors have less opportunities to query tags. It is observed that PCW with a high color change probability provides good results, which are better for high density networks.

5 Conclusion

This work has studied the effects of the introduction of a probabilistic factor in the collision resolution routine of colorwave. On the one hand, using a configuration proposed in [5], this technique does not result compliant with colorwave, since the presence of neighboring readers with a different total number of colors is too common. On the other hand, using a configuration proposed in [9], in high density network PCW provides a throughput higher than Colorwave.

Acknowledgments

This work has been partially supported by the grant "Nano -materials and -technologies for intelligent monitoring of safety, quality and traceability in confectionery products (NAMATECH)" from Regione Piemonte, Italy.

References

[1] M. Chen, S. Gonzalez, V. Leung, Q. Zhang, M. Li, A 2G-RFID-based e-healthcare system, Wireless Communications, IEEE 17 (1) (2010) 37 –43.

[2] P. Bernardi, C. Demartini, F. Gandino, B. Montrucchio, M. Rebaudengo, E. R. Sanchez, Agri-food traceability management using a RFID system with privacy protection, in: Proceedings of the 21st International Conference on Advanced Networking and Applications, AINA '07, 2007, pp. 68–75.

[3] D. Engels, S. Sarma, The reader collision problem, in: Systems, Man and Cybernetics, 2002 IEEE International Conference on, Vol. 3, 2002, p. 6.

[4] M. Bueno-Delgado, J. Vales-Alonso, C. Angerer, M. Rupp, A comparative study of RFID schedulers in dense reader environments, in: Industrial Technology (ICIT), 2010 IEEE International Conference on, IEEE, 2010, pp. 1373–1378.

[5] J. Waldrop, D. Engels, S. Sarma, Colorwave: an anticollision algorithm for the reader collision problem, in: Communications. ICC '03. IEEE International Conference on, Vol. 2, 2003, pp. 1206–1210.

[6] J. Waldrop, D. Engels, S. Sarma, Colorwave: a MAC for RFID reader networks, in: Wireless Communications and Networking. WCNC 2003. IEEE, Vol. 3, 2003, pp. 1701–1704.

[7] F. Gandino, R. Ferrero, B. Montrucchio, M. Rebaudengo, Introducing probability in RFID reader-to-reader anti-collision, in: Proceedings of the 2009 Eighth IEEE International Symposium on Network Computing and Applications, NCA '09, 2009, pp. 250–257.

[8] F. Gandino, R. Ferrero, B. Montrucchio, M. Rebaudengo, Probabilistic DCS: An RFID reader-to-reader anti-collision protocol, Journal of Network and Computer Applications 34 (3) (2011) 821–832.

[9] F. Gandino, R. Ferrero, B. Montrucchio, M. Rebaudengo, Increasing throughput in RFID multi-reader environments avoiding reader-to-reader collisions, in: Consumer Electronics (ICCE), 2011 IEEE International Conference on, 2011, pp. 37 –38.