Computers and Electrical Engineering 000 (2017) 1–8

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

An anti-collision algorithm for RFID-based robots based on dynamic grouping binary trees

Yin Zhang, Fan Yang, Qian Wang*, Qianlin He, JiaCheng Li, Yue Yang

School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan 430073, P.R. China

ARTICLE INFO

Article history:
Received 30 October 2016
Revised 6 March 2017
Accepted 7 March 2017
Available online xxx

Keywords:
RFID-based robots
Anti-collision
Query tree
Dynamic grouping mechanism

ABSTRACT

This paper investigates the conventional algorithms for addressing the issue of tag collision in radio-frequency identification (RFID) systems. To improve the efficiency of RFID reader inquiry, this paper proposes an approach based on dynamic grouping binary trees, named DGBT, for RFID-based robots. Through DGBT, the collision probability and query times of RFID readers are lowered by adaptive anti-collision prefixes, the efficiency of RFID identification is improved via the group-based approach, and the binary-tree-based queries within groups are ensured to be stable, flexible, effective and practicable. Analysis and simulation verify that the query times and throughput rate are significantly improved by DGBT.

© 2017 Published by Elsevier Ltd.

1. Introduction

With the development of electronics, sensors, automation and other techniques, mobile robots have reached high levels of performance in terms of real-time application, accuracy, robustness and compatibility [1]. However, the indoor localization of mobile robots is a challenging and active subject of study because of the complexity of indoor scenarios, the diversity of technologies involved, and the influence of commercial and industrial interests [2]. Although the Global Positioning System (GPS) is a widely accepted solution for outdoor operation, its accuracy is very limited when operating indoors because of limited satellite reception [3].

Over the past few years, a new kind of mobile robot that uses radio-frequency identification (RFID) for localization has received considerable attention because such a system is low in cost, virtually maintenance free and reliable in many harsh environments [4–6]. In many cases, a localization system of this kind is realized by installing a reader on the robot and distributing a certain number of tags in known positions throughout the environment [7]. The integration of robots with RFID readers in this way is one of the most efficient approaches in terms of indoor space resolution [8]. Miah and Gueaieb [9] proposed an RFID-based approach for mobile robot trajectory tracking and point stabilization through on-line neighboring optimal control. DiGiampaolo and Martinelli [10] addressed a global localization problem for a mobile robot that is able to detect the presence of passive RFID tags located in known positions on the ceiling of an environment when traveling below them. In [11], a shared control architecture combining a brain–machine interface with RFID technology was proposed for controlling a robot arm during pick-and-place operations.

However, RFID systems face considerable challenges when attempts to identify multiple tags simultaneously result in collisions among the tag responses to the reader's query, which significantly lower the efficiency of the system [12]. Therefore, anti-collision algorithms have been proposed to solve the RFID tag collision problem, including space-division

* Corresponding author.
E-mail address: qianwang@zuel.edu.cn (Q. Wang).

http://dx.doi.org/10.1016/j.compeleceng.2017.03.003
0045-7906/© 2017 Published by Elsevier Ltd.

multiple-access (SDMA), frequency-division multiple-access (FDMA), code-division multiple-access (CDMA), and time-division multiple-access (TDMA) schemes [13,14]. Because of their operability, TDMA approaches have been especially widely used, and these methods can be divided into two categories: ALHOA-based probabilistic algorithms [15] and binary-tree-based deterministic algorithms [16]. ALHOA-based probabilistic algorithms offer easy operation and rapid access, but it they are not very stable, and consequently, misidentification and lack of identification are unavoidable.

Thus, binary-tree-based algorithms are considered more suitable for addressing this issue, such as algorithms based on query trees (QTs) [17,18], binary queries, quadtree queries, etc. For example, an Adjustive Hybrid Tree approach (AHT) has been proposed for avoiding collisions by means of an adjustive tree based on the two highest bits [19]. An adaptive collision tree approach (ACT) has also been proposed to reduce query times by means of query strings, which are used to dynamically adjust the tree [20]. However, binary-tree-based algorithms are not able to achieve high efficiency because of their time-consuming identification, increased idle timeslots and collision timeslots.

Therefore, this paper proposes an anti-collision algorithm based on dynamic grouping binary trees (DGBT) for RFID-based robots to lower the complexity of the AHT and ACT approaches. In DGBT, an array of prefixes is generated for the grouping of RFIDs, and each group corresponds to a different timeslot. Moreover, in each group, the prefixes of branches without tags are deleted, whereas the prefixes of branches that include multiple tags are extended. Simulations verify that the proposed anti-collision algorithm significantly reduces the query times and transmission delay and increases the system throughput. Specifically, the main contributions of this work are as follows:

- The adaptive anti-collision prefixes reduce the collision probability and the query times of RFID readers.
- The group-based approach improves the efficiency of RFID identification.
- The binary-tree-based queries within the groups are stable, flexible, effective and practicable.

The remainder of this article is organized as follows. Related work is introduced in Section 2. We discuss the design of the anti-collision algorithm in Section 3. Section 4 analyzes the performance of the proposed scheme, and simulation experiments are reported in Section 5. Finally, Section 6 concludes the paper.

2. Related work

2.1. Binary trees and quadtrees

QT algorithms mainly consist of polling and response phases. Specifically, the reader sends a string (i.e., a query prefix) for querying, and only RFIDs that match the corresponding prefix are able to respond to the polling commands [21]. If only one tag responds, it will be successfully identified and send its ID to the reader. However, if there are multiple responding tags, tag collision will occur, and the reader will add 0 or 1 behind the previous query prefix, producing two new prefixes for a new round of polling, to ensure that only one tag can be matched.

Although the AT algorithm can identify all RFID tags, when no tags match the query prefix, there will be an unnecessary idle timeslot. For example, consider two tags that have the same prefixes and differ only by the last bit; collision timeslots and idle timeslots will be unavoidable during their identification. Moreover, this timeslot usage increases the query time and decreases the query efficiency.

Therefore, binary trees and quadtrees have been proposed for improved efficiency. Using binary trees and quadtrees, the tags are compared with the query prefixes, and a response to the polling command is issued in the suitable timeslot corresponding to the result of the comparison. Specifically, a binary tree includes two timeslots, whereas a quadtree includes four. Bai et al. [22] describe the number of timeslots in a multi-tree as shown in Eq. (1).

$$t(m) = l + B\sum_{i=1}^{m} B^{L}[1-(1-B^{-L})^{m}] - mB^{-L}(1-B^{-L})^{m-1} \quad (1)$$

Here, B represents the number of binary trees, m represents the number of RFID tags, and L is the currently accessed layer in the multi-tree. It is obvious that the number of timeslots will change if B is modified. In particular, it has been proven that when $B=3$, the number of timeslots is minimized, which means that the efficiency of tag identification is maximized. However, it is impossible to construct a triple tree, so binary trees and quadtrees are widely used.

2.2. Adjustive hybrid tree

AHT was proposed to adaptively adjust the query prefixes without excessive queries. In AHT, tags are often identified by their prefixes, and the identification process includes the following three kinds of timeslots.

- **Collision timeslots**: In a collision timeslot, multiple tags respond to the reader's query simultaneously, but the reader cannot identify any tags.
- **Success timeslots**: In a success timeslot, only one tag responds to the reader, and it can be successfully identified.
- **Idle timeslots**: In an idle timeslot, no tag responds to the reader.

Download English Version:

https://daneshyari.com/en/article/6883640

Download Persian Version:

https://daneshyari.com/article/6883640

Daneshyari.com