Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 149 (2016) 305 – 312

Procedia Engineering

www.elsevier.com/locate/procedia

International Conference on Manufacturing Engineering and Materials, ICMEM 2016,
6-10 June 2016, Nový Smokovec, Slovakia

Using Bee Algorithm in the Problem of Mapping

Timur Mazitov[a], Pavol Božek[b,*], Andrey Abramov[a], Yuri Nikitin[a], Ivan Abramov[a]

[a]Kalashnikov Izhevsk State Technical University, Department of Mechatronic Systems, Studencheskaya 42, Izhevsk 426069, Russia
[b]Slovak University of Technology, Faculty of Materials Science and Technology, Institute of Applied Informatics, Automation and Mechatronics, J. Bottu 25, Trnava 91724, Slovakia

Abstract

Innovative algorithm for solving simultaneous location and mapping problem in unknown environment is considered. The algorithm is based on the comparison of point clouds with bee swarm algorithm. The algorithm obtained was experimentally tested. The algorithm increases the number of calculations in several times as compared to standard methods of minimization (algorithm of gradient descent and others), but the function calculations in random points are similar operations and it enables the efficient application of parallel computations, thus resulting in increased performance.

Peer-review under responsibility of the organizing committee of ICMEM 2016

Keywords: SLAM; ICP; swarm algoritms; bee algorithm; location; mapping.

1. Introduction

One of the main problems in mobile robotics is simultaneous location and mapping (SLAM) in unknown environment. The first algorithms for the solution of this task were proposed by Leonard and Durrant-Whyte [1]. Their algorithms served for simultaneous estimation of newly perceived landmarks and position of mobile robot while mapping. Many researchers successfully expanded SLAM to indoor [2], outdoor [3], underground [4,5] and underwater [6] environments, in two-dimensional and three-dimensional applications [7,8,9].

Mapping in SLAM algorithms is based on comparison of data from sensors (odometry, lidar or sonar data) at discrete time instants: the object displacement along the basis is calculated with certain accuracy, then the map is supplemented with new data. Data comparison is characterized by high computing complexity. Accuracy of map development depends on accuracy of object position calculation. The existing algorithms of data comparison can be broken down into two types depending on the input data used by them: application of various features (lines, special points [10]) and application of "raw" data [11]. The considered algorithm is related to the second group.

* Corresponding author. Tel.: +79068164399
E-mail address: nikitin@istu.ru

Peer-review under responsibility of the organizing committee of ICMEM 2016
doi:10.1016/j.proeng.2016.06.671

2. Theoretical part

2.1. Mathematical model

The object position in two-dimensional space at each time instant is assigned by matrix R_0, characterized by absolute coordinates x_0, y_0 and driving direction described by angle $\varphi_0[-\pi; \pi]$:

$$R_0(k) = \begin{bmatrix} x_0(k) \\ y_0(k) \\ \varphi_0(k) \end{bmatrix} \tag{1}$$

where k – discrete instant of time.

In order to obtain information about the environment, the HOKUYO utm-30lx-ew sensor is used, providing data in the range from -1350 up to 1350 with the step 0.250 at 40Hz, which can be presented by the following vector:

$$Z(k) = \{z_1(k), z_2(k), ..., z_{1081}(k)\} \tag{2}$$

where $z_l(k)$ – point defined by the polar coordinate at time k, where l =1…1081.

The example of the obtained data is graphically shown in Figure 1.

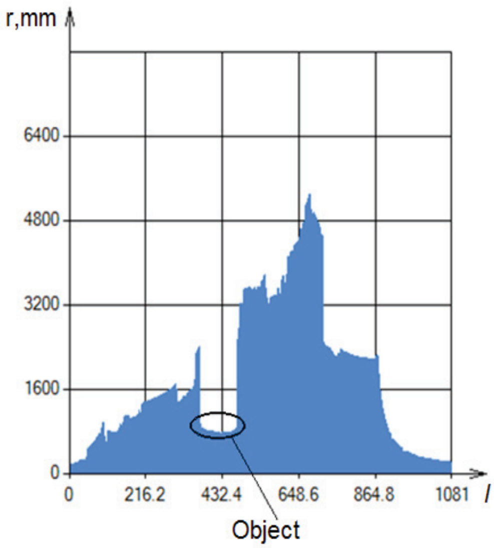

Fig. 1. Histogram showing the lidar data.

In Figure 1 there is the sharp change within relatively smoothly changing data describing the room perimeter. This array of points characterizes the object in the observed area.

The representation of points in Cartesian coordinate system is shown in the following formula:

$$Z^*_{local}(k) = \begin{bmatrix} x_l(k) \\ y_l(k) \end{bmatrix} = \begin{bmatrix} z_1(k) * \cos\varphi_l \\ z_l(k) * \sin\varphi_l \end{bmatrix} \tag{3}$$

where l = 1 … 1081.