2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)

Energy Efficient Communication with Lossless Data Encoding for Swarm Robot Coordination

Karthik Narayanan
Intel Corporation
Bangalore, India
karthik1.narayanan@intel.com

Vinayak Honkote
Intel Corporation
Bangalore, India
vinayak.honkote@intel.com

Dibyendu Ghosh
Intel Corporation
Bangalore, India
dibyendu.ghosh@intel.com

Swamy Baldev
NIT Meghalaya
Shillong, India
swamy.baldev@gmail.com

***Abstract*— Energy efficient communication is a key aspect of swarm robot systems working collaboratively towards accomplishing complex missions. We propose a coding scheme inspired by Huffman encoding for lossless data compression optimized for swarm robot communication. In this approach, the perception information on each robot is encoded with known values based on which the algorithm is optimized. The proposed method is implemented and tested on the Intel Edison platform with custom robot chassis. UWB (Ultra-Wide Band) based communication technology is integrated on this platform for enabling robot-to-robot communication. Experiments performed using the proposed approach demonstrate ~46% reduction in power dissipation compared to normal data exchange.**

Keywords—robots, swarm, energy-efficient communication.

I. INTRODUCTION

Swarms of homogenous/heterogeneous robots are increasingly used in complex applications such as, search and rescue, 3D mapping of an unknown environment, inspection, package delivery and surveillance - either as ground robots or as drones. In majority of such complex applications, it is crucial for the robots (ground and/or aerial) to coordinate with each other [1]. In Fig. 1, different elements in a swarm system are depicted. For reliable operation of swarm robots, it is critical to enable real-time and robust coordination. This is achieved using intra-swarm and inter-swarm robot communication [2]. Further, having energy efficiency built into these communications is critical for prolonged operation of mobile robots.

There are many research papers describing different methods and overall framework for communication in swarm systems [3]. However, these methods do not address the power dissipation and optimization aspects of data exchange between the robots in a swarm system. To this end, we propose a framework for a compression based scheme to minimize the data size that needs to be exchanged between the agents in a swarm system. In this approach, the perception data gathered by each robot is encoded with 'R' symbols corresponding to different probabilities and redundancy. Code books are generated from the R symbols. Huffman coding algorithm [4] is optimized for better compression of the information without the loss of data and to reduce the time and bandwidth (power) of the swarm robot communication. The proposed system is designed, implemented and demonstrated on Intel Edison [5] compute platforms hosted on the actual system of swarm robots.

II. PROPOSED METHODOLOGY

In this work, we propose a coding scheme designed for data compression in multi-robot systems. The proposed method takes advantage of modified codebooks, *optimized* for target applications. A generic 'k' bit input data symbol is considered to represent the perceived environmental input - also called as *map*. These maps are typically obtained by processing the gathered sensor data. A sample exploration based task is considered for multiple robots to coordinate and cover an unknown environment. The robots coordinate with each other by exchanging maps. In this representative scenario, we choose an 8-bit representation for each cell in the map. Further, the data is prioritized into three levels - area coverage, perception information and obstacle detection as depicted in Fig. 2. These priorities are assigned at the time of deployment and can be optimized according to the scenario. Note that, the size of the input data symbol is *configurable*. This allows for enabling different data sizes depending on the application. Pre-calculated statistical analysis of the different information included in the map are created beforehand. This information forms the source of the codebook. In Fig. 3, the steps involved in generating the codebook are represented. The codebook generated is read and frequency information is derived according to the given codebook probability, which in turn is decided by the input symbol priorities. Next the map data that needs to be transmitted is loaded. The map information is compressed using encoder algorithm to produce the required code and is transmitted. At the receiver robots, encoded data is read and the decoder algorithm is applied to retrieve the original information without any loss of data.

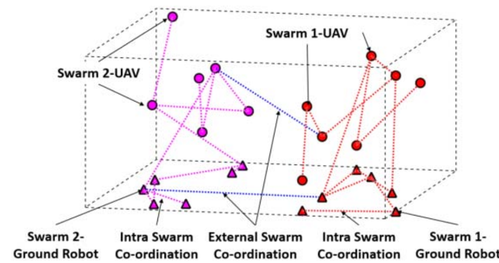

Fig. 1. Example of swarm robot coordination

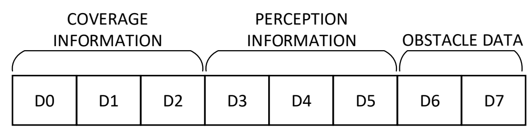

Fig. 2. Proposed input data symbol for area exploration application

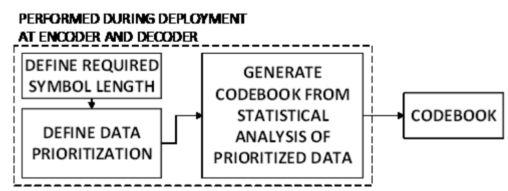

Fig. 3. Generation of codebook

2380-6923/19/$31.00 ©2019 IEEE
DOI 10.1109/VLSID.2019.00118

III. Experiments And Results

The experiments were carried out on a system of swarm robots. Each robot is assembled using third party mechanical components. A UWB transceiver–with DW1000 UWB chip [6] and an antenna–is mounted on each robot. The compute involved is handled by Intel Edison board. The robots are equipped with the required sensors to collect runtime information of its surrounding and are designed to perform a coverage action on a 5m X 5m area.

A fixed symbol length which matches the scenario is chosen, followed by the selection of information priorities (as described in Fig. 2). Using prior information about the distributions of the individual components of the symbol, the statistical distribution of the symbols throughout the entire map is estimated. Two experiments are devised and the results for compression and power analysis are described as below.

Experiment 1: Using randomly generated inputs based on expected statistical distributions

In this experiment, map information is randomly generated based on the expected distributions to fill the swarm robot's map. These results point to the scenario when the inputs exactly match the expected distribution. This dataset will thus act as a control input to rate the performance of any further experiments described. The experiment is carried out using different map sizes (5 X 5 to 1000 X 1000 grids) and corresponding compressed map sizes are noted. It is observed that ~56% compression ratio is achieved compared to the original data.

Experiment 2: Using data collected from a sample exploration application

In this experiment, data generated at runtime based on area exploration application in a 5m x 5m area is considered for compression. There are three different cases considered. Similar to *Experiment 1*, the map size is varied from 5 X 5 to 1000 X 1000. Further, experiments are carried out in 3 steps. First, only coverage information is considered for compression. Next, both coverage and perception information are considered for compression. Finally, the whole symbol is considered for compression. In each of the cases, an overall compression ratio of ~48%, ~51% and ~55% respectively were achieved. The corresponding results are represented in Fig. 4. Note that, detailed experiments were run for each case. However, due to space constraints, the details are presented only for the most generic cases. As more fields are selected for compression, the achieved ratios are observed to trend closer to the compression number achieved using the control input (*Experiment 1*). Details of the comparison with *Experiment 1* are presented in Table 1.

Detailed power analysis for the proposed method was performed on the experimental setup described. For the map data ranging from 5 X 5 to 1000 X 1000 grids, ~50% reduction in execution time was observed using the proposed scheme compared to the un-compressed data transfer. Further, the power component resulting from the data transfer is analyzed. In Fig. 5, power analysis results are plotted, where blue circles represent uncompressed data and orange triangles represent compressed data. It is observed that the proposed method (with compression) results in 46% of reduction in overall power dissipation.

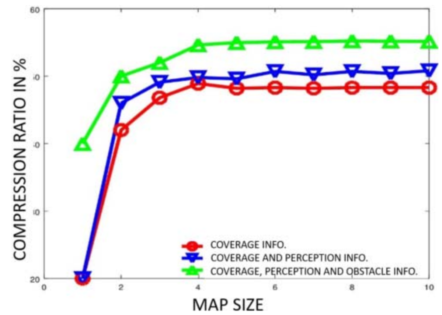

Fig. 4. Results of data compression

TABLE I. Comparison of compression results

Map Size	*Original size (KB)*	*Compressed size for Experiment 1 (KB)*	*Compressed size for Experiment 2 (KB)*
5 X 5	0.025	0.014	0.015
10 X 10	0.100	0.049	0.050
100 X 100	10	4.455	4.503
250 X 250	62.5	27.74	27.90
500 X 500	250.0	110.96	112.04
1000 X 1000	1000.0	443.53	448.44

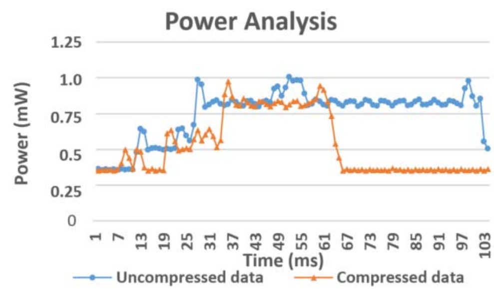

Fig. 5. Comparison of power consumption during data transmission

IV. Conclusion

In this paper, an energy efficient, lossless compression methodology is proposed for data exchange between the agents in a swarm. The method is configurable to cater to different deployment scenarios and data sizes. The proposed method is employed and demonstrated for a representative area exploration application. Experiments demonstrate >50% compression ratio resulting in ~46% reduction in power compared to normal (un-compressed) data exchange. With the increase in the number of agents in the swarm (leading to more data exchanges) and increase in the size of the data, the overall power savings will further improve.

References

[1] B. Pang, C. Zhang, Y. Song, H. Wang, "Self-organized task allocation in swarm robotics foraging based on dynamical response threshold approach," 18th International Conference on Advanced Robotics (ICAR), Hong Kong, China, 2017, pp. 256-261.

[2] M. Li, K. Lu, H. Zhu, M. Chen, S. Mao, B. Prabhakaran, "Robot swarm communication networks: architectures, protocols, and applications," 3rd International Conference on Communications and Networking in China, 2008, pp. 162-166.

[3] X. Wu, J. Sun, "Study on a KQML-based intelligent multi-agent system," IEEE International Conference on Intelligent Computation Technology and Automation, 2010 pp. 466-469.

[4] D. A. Huffman, "A method for the construction of minimum-redundancy codes," Proc. Of the IRE, 1952, vol. 40, pp. 1098-1101.

[5] Intel Edison: https://software.intel.com/en-us/iot/hardware/edison.

[6] DW1000 User Manual: www.decawave.com/products/dw1000.