DISTRIBUTED WIRELESS SENSOR NETWORK
SYSTEMS: THEORETICAL FRAMEWORK,
ALGORITHMS, AND APPLICATIONS

by

DONGHWA JEONG

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

Department of Mechanical and Aerospace Engineering

CASE WESTERN RESERVE UNIVERSITY

August, 2015
CASE WESTERN RESERVE UNIVERSITY
SCHOOL OF GRADUATE STUDIES

We hereby approve the dissertation of
DONGHWA JEONG
candidate for the degree of Doctor of Philosophy*.

Committee Chair
Kiju Lee

Committee Member
Roger Quinn

Committee Member
Frank Merat

Committee Member
M. Cenk Cavusoglu

Date of Defense
August, 2015

*We also certify that written approval has been obtained
for any proprietary material contained therein.
Contents

List of Tables v

List of Figures vi

Acknowledgments xi

Abstract xii

1 Introduction 1
  1.1 Motivation ...................................................... 2
  1.2 Research Objectives ........................................... 4

2 Theoretical and Technical Background 8
  2.1 Centralized vs. Distributed Sensor Networks ................... 8
  2.2 Design Considerations in DWSN ............................... 10
  2.3 Taxonomy of DWSN .............................................. 12
  2.4 Time Synchronization ........................................... 14
  2.5 Sensors for Wireless Communication and Localization ........ 15

3 Algorithms 21
  3.1 Self-Synchronization ........................................... 22
  3.2 Hybrid Wireless Network Strategy using IR and RF ........... 25
  3.3 Motion Detection Algorithm ................................... 26
# Primary Application: TAG-Games

## 4.1 Overview

## 4.2 Related Works on Technology-Embedded blocks

## 4.3 Hardware Development

### 4.3.1 SIG-Blocks

### 4.3.2 iSIG-Blocks & Platform

## 4.4 TAG-Game Design

### 4.4.1 Three TAG-Games

### 4.4.2 Memorix

### 4.4.3 TAPware: Tangible Programming Software using iSIG-Blocks

## 4.5 Play Complexity and Scoring Methods

### 4.5.1 Play Complexity

## 4.6 Technical Evaluation

### 4.6.1 Durability

### 4.6.2 Proximity Sensing Failure

### 4.6.3 Peer-to-Peer Synchronization Delay

### 4.6.4 Visualization Delay

## 4.7 Human Subject Evaluation

### 4.7.1 Three TAG-Games

### 4.7.2 Scoring Methods

### 4.7.3 Preliminary Evaluation of Memorix

## 4.8 Summary of TAG-Games

## 5 Extended Application I: Swarm Robots

## 5.1 Overview

## 5.2 Related Works on Swarm Robots
5.3 Hardware Development ........................................... 90
  5.3.1 Mobile Platform with a Corner Reflector ............... 90
  5.3.2 InchBots ................................................. 93
5.4 Algorithms ..................................................... 98
  5.4.1 Directional RSS-Based Localization ....................... 98
  5.4.2 Control Scheme ........................................... 103
5.5 Experiments and Simulations ................................. 110
  5.5.1 Experiment on Mobile Robots with a Corner Reflector .... 110
  5.5.2 Mobility Experiment on InchBot .......................... 112
  5.5.3 Simulations on Dispersion and Line Formation ............ 114
5.6 Summary of Swarm Robots .................................... 118

6 Extended Application II: Wearable Sensors ................. 120
  6.1 Overview ..................................................... 120
  6.2 Related Works on Wearable Monitoring Sensors .............. 121
  6.3 Hardware Development ...................................... 123
    6.3.1 Socio-biosensor V.1 ..................................... 123
    6.3.2 Socio-biosensor V.2 ..................................... 125
  6.4 Embedded Sensors and Measurable Data ...................... 125
  6.5 Preliminary Experiment ..................................... 129
  6.6 Summary of Wearable Sensors ................................ 129

7 Extended Application III: Origami Robots ................. 132
  7.1 Overview ..................................................... 132
  7.2 Related Works on Origami Robots ............................ 133
  7.3 Hardware Development ...................................... 135
    7.3.1 OrigamiBot-I: Robot that Crawls and Manipulates ....... 135
    7.3.2 OrigamiBot-II: Amphibious Robot that Deforms Wheels ... 139

iii
# List of Tables

2.1 Review of frequently used sensors ........................................ 16  
2.2 Comparison of wireless communication protocols .......................... 17  
2.3 Comparison of distance measuring sensors .................................. 18  
4.1 Overview of technology-embedded blocks and SIG-Blocks .................. 38  
4.2 Three TAG-Games and associated cognitive skills ............................ 50  
4.3 Assessment time and number of items in TAG-Games and WAIS-IV .... 71  
4.4 Scoring methods for TAG-Games ............................................. 73  
4.5 Summary of TAG-Games scores and three subtests of WAIS-IV .......... 76  
4.6 Correlations among the three TAG-Games and WAIS-IV subtests ...... 76  
4.7 Parameter sensitivity index ..................................................... 83  
6.1 Components embedded in the socio-biosensors ............................. 127  
7.1 Component cost used in OrigamiBot-III .................................... 144  
7.2 Deflection angle (\(\alpha\)) per loaded weight. ............................... 160  
7.3 Printing or folding time and cost of each part ............................... 164  
8.1 Performance parameters ..................................................... 173  
8.2 Parameter values of the applications ....................................... 173  
8.3 Evaluation of the applications .............................................. 174
List of Figures

2.1 Centralized and Distributed System .................................. 9
2.2 Taxonomy of DWSN at the software level. .......................... 12
3.1 Short-range communication mechanism and self-synchronization ... 23
3.2 IR communication data packets (two bytes of talking) ............. 24
3.3 Hybrid wireless network scheme ...................................... 25
3.4 Tilt angle measurement ............................................... 27
3.5 Pitch and roll angles measured relative to the global horizontal frame using a triaxial accelerometer. .......................... 27
3.6 $x_{acc}$ captured from the triaxial accelerometer versus time while rotating about the $y$-axis. .......................... 28
3.7 Frequency spectrum of the rotating motions in four different frequencies. 29
3.8 Example of $2 \times 2$ assembly configuration seen from the top. .... 30
4.1 SIG-Blocks: interactive creation blocks designed for tangible interactive games. .......................... 33
4.2 Related Works on Technology-embedded blocks ..................... 36
4.3 SIG-Blocks: interactive creation blocks designed for tangible interactive games. .......................... 40
4.4 Six distinctive images with 1-, 2-, and 4- fold rotational symmetry. . 41
4.5 Customized cover layer design .......................... 41
4.6 interactive sensor-integrated geometric blocks, called iSIG-Blocks . . 42
4.7 CAD drawings of iSIG-Blocks and embedded components . . . . . . 44
4.8 Tabletop platform . . . . . . . . . . . . . . . . . . . . . . . . . . . 44
4.9 Pattern generating surface design candidates . . . . . . . . . . . . . 45
4.10 Diffuse reflection mechanism . . . . . . . . . . . . . . . . . . . . . 46
4.11 Connection mechanism of iSIG-Blocks . . . . . . . . . . . . . . . . 47
4.12 Magnetic field analysis on iSIG-Blocks . . . . . . . . . . . . . . . . 48
4.13 Administrator’s graphical user interface . . . . . . . . . . . . . . . . 50
4.14 20 TAG-GameA items. . . . . . . . . . . . . . . . . . . . . . . . . . 51
4.15 10 TAG-GameS items. . . . . . . . . . . . . . . . . . . . . . . . . . 52
4.16 8 TAG-GameM items. . . . . . . . . . . . . . . . . . . . . . . . . . 53
4.17 The administrator screen of Memorix . . . . . . . . . . . . . . . . . 54
4.18 Geometric patterns used in Memorix . . . . . . . . . . . . . . . . . 54
4.19 Memorix demonstration . . . . . . . . . . . . . . . . . . . . . . . . . 56
4.20 Launchpad screen of TAPware . . . . . . . . . . . . . . . . . . . . . 56
4.21 Pattern generator of TAPware . . . . . . . . . . . . . . . . . . . . . 57
4.22 Tangible programming of geometric patterns using TAPware . . . . 58
4.23 An overview of the pattern programming procedure. . . . . . . . . . 59
4.24 Complexity measures of each game: $C^A, C^S,$ and $C^M$. . . . . 61
4.25 Play complexity of patterns . . . . . . . . . . . . . . . . . . . . . . 66
4.26 SIG-Blocks after falling test . . . . . . . . . . . . . . . . . . . . . . 67
4.27 L-shape support for playing games by rotating a single block to match a pattern or play back the remembered pattern. . . . . . . . . . 68
4.28 Histogram of the self-synchronization time for 100 trials (Mean = 300 ms; SD = 150 ms) . . . . . . . . . . . . . . . . . . . . . . . . . . 69
4.29 Human subject evaluation setup . . . . . . . . . . . . . . . . . . . . . 70
4.30 Speed of rotations for three groups . . . . . . . . . . . . . . . . . . . 78
4.31 Number of rotations for three groups. ........................................ 79
4.32 Scores by participants ......................................................... 81
4.33 Number of participants who successfully memorized each pattern ... 81
4.34 Failure rate and pattern complexity variations for patterns (P1-P12) . 82

5.1 Related works on swarm robots ............................................ 87
5.2 CAD drawing and physical prototype of the mobile robot .......... 91
5.3 A corner reflector with a dipole located in-between two plates .... 92
5.4 Experimental results of RSS and directivity ................................. 92
5.5 InchBot, a omnidirectional micromobile robotic platform .......... 94
5.6 CAD drawing and physical prototype of InchBot. ..................... 95
5.7 Four-wheeled robot............................................................... 95
5.8 Flexible spoke wheels ........................................................... 97
5.9 FEA model for simulating the deformation of the flexible spoke wheel 98
5.10 Localization scheme of mobile robot with corner reflector .......... 102
5.11 RSS vs. distance measurements and log_{10}-distance .................. 103
5.12 Schematics of the embedded control method. ......................... 103
5.13 Illustration of directional derivative-based control scheme ........... 103
5.14 Histogram of the self-synchronization time for 50 trials ............. 105
5.15 Inliers extraction from raw RSS data versus distance and log_{10}-distance 106
5.16 Experimental result on Tracking time [sec] vs. distance [in] ........ 110
5.17 Fixed and mobile target localization ..................................... 112
5.18 Distance versus time for three different duty cycles .................. 113
5.19 Forward motion experiment ............................................... 114
5.20 Turning motion experiment ................................................. 114
5.21 Diagonal motion experiment ................................................ 114
5.22 Attractive and repulsive forces between the nodes are modeled with a 
spring and a damper. ......................................................... 115

viii
7.14 Kinematic model of OrigamiBot-II .................................................. 150
7.15 Force model applied to the OrigamiBot-III by wires .............................. 151
7.16 Workspace of OrigamiBot-III .............................................................. 153
7.17 Overall system architecture with internet-based control ......................... 154
7.18 Graphical user interface for internet-based teleoperation ......................... 155
7.19 Loading and unloading durability testing ............................................. 157
7.20 Performance result of 1000 cycles of loading and unloading origami structure .......................................................... 158
7.21 Stiffness changes induced by fatigue .................................................... 159
7.22 Video captured images of bending motion and loading test ..................... 159
7.23 Forward movement in top view (Oscillation frequency is 0.25Hz) .......... 160
7.24 Directional passive flaps ................................................................. 161
7.25 Obstacle navigation with sequential control scheme .............................. 162
7.26 Locomotion on road and grass .......................................................... 163
7.27 Locomotion on water ................................................................. 163
7.28 Five objects used in the manipulation test ......................................... 165
7.29 Manipulation experiment for five different objects .............................. 166
7.30 CAD drawing and physical gripper platform ........................................ 167
7.31 FSR value versus applied force (N) .................................................. 167
7.32 Current consumption and FSR value versus stroke ................................ 168
8.1 Evaluation of four applications ......................................................... 175
Acknowledgments

During my doctoral research, particularly in writing to my dissertation, I looked back and thought about what I learned from this long journey in the USA. I felt I was very lucky that I met many lovely people that have guided and helped me. My wife, best friend, and soul mate, Kyunghee dedicated her five years to cheer me up and motivate me whenever I am frustrated or decided to give up. You deserve the best and I promise to give this luck to you. I would like to express my sincere gratitude to my advisor, Dr. Kiju Lee, for providing the opportunity to work on research and for her support during my graduate study. Your confidence and insight in research inspired me and nourished my growth as a successful educator and engineer. In addition, I deeply appreciate my committee members, Dr. Roger Quinn, Dr. Francis Merat, and Dr. Murat Cavusoglu. Also, I would especially like to thank Dr. Elizabeth Short for guiding me on clinical studies. Also, I would like to say thanks to my lab mates: Ken Hornfeck, Endri Kerci, Yan Zhang, Beatrice Floyd, and Tao Liu. Special thanks go to Chris Puehn and Nick Ang for providing feedback and proofreading for this dissertation. Thanks to my Korean friends Jungkyu, Hyungjin, and Myeongseop who shared my concern and refreshed me with good comments. Thanks to Shiela and Carla for helping me all of these years. Lastly, I would like to thank my mother, father, brother, mother-in-law, father-in-law, sister-in law and Lena. They always assured me that I am loved, which gave me the strength to keep moving forward.

Abstract

by

DONGHWA JEONG

This dissertation presents a theoretical framework and algorithms for distributed wireless sensor networks and their physical implementations on tangible geometric games (TAG-Games) to assess cognitive and motor skills automatically using sensor-integrated geometric blocks (SIG-Blocks) with an interactive graphical user interface. To allow for a variety of possible game designs, single and multiple blocks were used to form different geometrical configurations. To resolve communication issues between the blocks as well as reliability issues with sensing, the following distributed wireless sensor network algorithms were developed: 1) self-synchronization technique in a local network, 2) a hybrid wireless network, 3) an assembly detection algorithm, and 4) a motion sensing algorithm. The theoretical framework of this paper is that distributed wireless sensor network is fault tolerant, scalable, and a dynamic solution to complex multi-agent systems. To evaluate the validity of the TAG-Games with the developed algorithms and proposed theoretical framework, 86 participants were recruited to the human subject evaluation aging from 18 to 30. The results confirmed that the TAG-Games is suitable as an automatic assessment tool. The developed algorithms and theoretical framework are applied to three extended applications, including swarm robots, wearable sensors, and origami robots. First, two swarm robotic platforms were developed and the wireless algorithms were implemented. One of them
is a group of mobile robots with corner reflectors to localize using directional received signal strength. The other one is the InchBots, a cubic inch-sized omnidirectional swarm robotic platform with a stackable hardware feature for customization. The robotic swarm aims to achieve a global goal by using a large number of robots, each with limited capabilities. In this application, the hybrid wireless network, assembly detection algorithm, and the self-synchronization algorithm were utilized and a geometry-based global shape formation algorithms were also implemented. The simulation results showed that the simple robotic agents can achieve global shape formation, i.e., line formation and radial dispersion, by using only received signal strength without centralized control. Second, wearable sensors were prototyped aiming to measure person-to-person interactions as well as biological and environmental data. Integration of the received signal strength from the proposed hybrid wireless network algorithm and motion detection algorithms were embedded in the system. Lastly, robots with a novel origami design were developed to achieve amphibious locomotion and articulated manipulation using the semi-soft origami structures. In this application, the data traffic resulting from real-time video streaming was controlled by distributing the hardware and internet protocol efficiently. The presented applications were then validated for the system efficiency, cost, reliability, and showed the great potential in the field of robotics, clinical studies, and in education.
Chapter 1

Introduction

The first project during my doctoral research was to develop tangible geometric games (TAG-Games) for automatic assessment of cognitive and developmental disabilities through a play session with sensor-integrated geometric blocks (SIG-Blocks). To reduce the system cost and hardware complexity while ensuring scalability for a variety of clinical applications, a distributed wireless sensor network (DWSN) system was adopted. As the system was intended for clinical applications, the system must operate robustly with little to no error. Also, some procedural and technical issues remain unsolved, such as selection of the most appropriate wireless protocol, efficient communication without data collision, and the level of sensor integration and collaboration among the sensors. One of the main difficulties is in hardware implementation. Multiple design criteria including universality and scalability often conflict with constraints in cost, power dissipation, safety, reliability, and overall size. For example, the universal design of SIG-Blocks requires various embedded sensors which increases the cost, power dissipation, and overall size of the blocks. Similarly, scaling up the system in wirelessly connected devices causes increased data traffic which makes the network system unreliable. Design is also relatively subjective so multiple modifications are expected during the development process such as addition of new blocks or
The DWSN developed for the TAG-Games project was found to have potential for various engineering applications, specifically in multi-agent systems. The theoretical framework proposed in this study is that DWSN offers fault tolerant, scalable, and dynamic solutions to complex multi-agent systems. Therefore, three additional hardware systems were developed and the DWSN was applied in parallel to validate the theoretical framework and to demonstrate the feasibility of the self-synchronization, implemented hybrid wireless network, motion detection, and assembly detection algorithms. The three extended applications include 1) swarm robots, 2) wearable sensors, and 3) origami robots. Each hardware platform faced design challenges to meet the technical constraints and aesthetic and functional demands. Throughout the study, the scope of this research is focused on the distributed system by defining the DWSN as a wireless sensor network with spatially distributed sensors and actuators to monitor physical or environmental conditions and to control the behavior of each agent to accomplish a global goal. In this context, the agent is defined as a tangible object with limited sensing, communication, and actuating capabilities. The research described in this dissertation also covers the methods and results associated with the DWSN in each application to overcome traditional drawbacks, enable new control methodologies, and deliver meaningful outcomes to our society where applicable.

1.1 Motivation

Recent manufacturing technologies and wireless network development have enabled low-cost and large scale production of numerous products ranging from portable devices to sensor-networked autonomous monitoring systems. For example, a microprocessor such as the Intel pentium used to cost around the same as a small car in 1991, but for the same processing capability nowadays the price is as low as a candy bar.
enabling a large scale study with multi-agent systems is possible by integrating and embedding low-cost sensors, actuators, and communication modules [1]. These multi-agent systems have similar characteristics because their overall tasks are achieved by coordinating the sensors, actuators, processing units, power sources, and communication modules in an efficient manner. For such distributed systems to achieve their global goal, the overarching question is that how a large number of agents are programmed to achieve a global goal while being connected wirelessly. There is a need for versatile algorithms that can be applied to a broad range of applications to achieve the goal while maintaining robustness, fault tolerance, and reliability.

DWSN is useful when gathering real-time information from spatially dispersed sensors or mobile platforms [2, 3]. Redundancy in robotics often refers to kinematic redundancy and actuation redundancy. The kinematic redundancy occurs when extra active joints and links are added and actuation redundancy is attained when passive joints are replaced with active joints. These redundancies consume more energy and need more controlling variables, but have benefits such as singularity reduction and robustness with respect to mechanical failure [4]. Similarly, the redundancy of the network channels, employed sensors, and processed information in a DWSN can cause inefficiency in energy consumption and structural cost, however, this redundancy may be necessary to ensure reliability and robustness in the system for the particular applications [5].

A DWSN is essential for systems requiring multi-agents to sense from sparsely distributed locations, to communicate between spatially uncoordinated agents, and to localize the agents in an unknown environment. The introduced applications utilize the theoretical framework and DWSN algorithms as follows: First, development of an assembly detection and self-synchronization algorithm for multiple blocks allowed automatic assessment of cognitive and fine motor skills in TAG-Games. To employ more blocks without data traffic, a hybrid wireless network is also applied.
Second, the hybrid wireless sensor network with infrared and radio frequency ensured wireless connectivity and distance estimation in swarm robots. The swarm robots deploy multiple robots with limited capabilities to achieve global goals such as disaster monitoring or rescue operations in various environments where a single integrated robot would not be applicable. As programming each robot with different behaviors costs vast time and energy, a single algorithm to control all robots is applied on the DWSN. Third, wearable sensors are developed to collect environmental, biological, and social data. Due to the limited infrastructure in communication or operating time with small batteries, a DWSN is utilized for data acquisition in this application. Lastly, a DWSN is applied to an origami robot for an affordable, easy-to-make robot that can be monitored and controlled in real-time. This feature makes it ideal as an educational kit. To avoid using an expensive laptop-like system, Raspberry Pi and Arduino based opensource platforms were fully utilized for processing, data storing, and networking in a distributed manner.

1.2 Research Objectives

The overall goal of the presented research is to develop distributed wireless network algorithms and hardware systems for a broad range of healthcare and robotic applications. Specific objectives for each project are as follows.

TAG-Games

TAG-Games aims to address the following limitations in the traditional cognitive assessment methods: 1) high cost due to clinician/administrator time, 2) limited amount and types of measurable data by a human administrator, 3) subjectivity and human measurement errors, 4) limited accessibility to professional services for people living in hard-to-reach areas, and 5) difficulty in addressing individual differences in
cognitive skills, age, and developmental status due to lack of flexibility in test items. To resolve these limitations, the major goal of this research is to develop a system consisting of modular blocks equipped with wireless sensing capabilities accompanying distributed algorithms with friendly games for automated assessment of cognitive and fine motor skills. To achieve this goal, the following specific aims were pursued:

- **Aim 1:** Implement sensor-integrated geometric blocks with an interactive graphical user interface.
- **Aim 2:** Develop computational measure of play complexity.
- **Aim 3:** Evaluate if the developed system and proposed play complexity can perform quantitative assessments by testing on young adults and analyzing the collected behavioral data with standard assessment methods.

**Swarm robots**

Collective behavior in swarm robotics explores various scenarios involving many robots communicating, sensing, and running simultaneously. This strategy aims to reduce the time and energy required and to improve the efficiency of completing complex tasks which are typically difficult to accomplish by an individual robot. To accomplish this goal, the following specific aims are made:

- **Aim 1:** Investigate the distance estimation and localization techniques with a sensor based approach.
- **Aim 2:** Develop the swarm robot hardware in a small scale that is capable of sensing and wireless communication.
- **Aim 3:** Develop and simulate shape-forming algorithms based on local interactions between robotic agents.
Wearable sensors

Focusing on the healthcare applications with a DWSN, the major challenges are 1) creating an ergonomic and aesthetic design, 2) ensuring reliable sensing during daily activity, and 3) selection of appropriate network protocols to collect behavior. In an attempt to address these challenges, the main goal of this application is to develop a low-cost wearable hardware platform for environmental, biological, and social monitoring. To accomplish this goal, the following specific aims were pursued:

- **Aim 1**: Design ergonomic stand-alone device to collect data on human behavior and daily activity, as well as social interactions.

- **Aim 2**: Development and validation of socio-biosensors to monitor biological, environmental, and social data with a DWSN.

Origami robots

Considering an educational application with low-cost material and a safe design, paper-folded origami robots are developed. Origami folding requires dextrous hand manipulation and attention skills as well as the ability to follow instruction carefully. The main advantage of the origami robot is the semi-flexible hardware that can solve some challenging engineering problems associated with locomotion across rugged terrain and manipulation of fragile objects. The main goal of this project is to develop a low-cost educational platform for improving motor skills and learning programming skills using an opensource embedded system. To address the goal, three specific aims were addressed:

- **Aim 1**: Develop a series of origami robot platforms to be able to perform a specific task utilizing the advantages of the structure.

- **Aim 2**: Build a kinematic model of the structure and perform a simulation using the opensource simulator, Gazebo.
• **Aim 3:** Configure the network architecture to allow teleoperation using a sensor network and real time video streaming.
Chapter 2

Theoretical and Technical Background

2.1 Centralized vs. Distributed Sensor Networks

The selection of whether to use centralized or distributed communication and control highly depends on the target applications. Both centralized and distributed systems are discussed in various domains such as political science, supply chain, and informatics [6, 7]. In sensor networks with centralized processing, as illustrated in Fig.2.1(left), the sensor values obtained by the individual sensors across the network are transmitted directly from the sensors to a central agent, where it is processed and distributed directly to the end users. Thus, each sensor functions simply as an information gathering device, while all the processing such as pattern recognition, inference, and segmentation is performed using complex centralized algorithms at the back-end. The benefits of a centralized sensor network are that there is no redundancy in collected data as all data will be sent to a single central node and the hierarchical structure makes for easy scheduling and synchronization within the network [8]. However, this process is energy-intensive and prone to communication constraints, making the net-
work non-scalable and non-robust. It may put undesirable limits on the amount of data that can be collected by a sensor. Also, in query processing, this may involve transmission of unnecessary information, resulting in communication overhead and response delays. Finally, in some situations, transmission of information to and from a central station is not feasible because of accessibility or security concerns. Centralized processing is thus time-intensive, inefficient and insecure, and it reduces the lifetime of the network. It works well for small systems in limited applications, but for large systems, scalability and other issues limit its applicability.

Figure 2.1: Centralized and Distributed System

In decentralized/distributed processing, as shown in Fig.2.1(right), each agent not only gathers data, but also carries out computations locally. It collaborates with its neighbors and derives local inferences by processing the information. Thus, new global behavioral patterns emerge from local interactions between the agents through the process of self organization. The agents are designed to create an impromptu network, assemble the network on their own and adaptively respond to degradation, device failure and other changes in the structure of the network. Distributed networks
are particularly time efficient compared to centralized networks when the system is expected to respond to local phenomena. Since local measurements are highly correlated, the dimensionality of data is reduced through data aggregation. However, care should be taken that no pertinent data is lost in the process, while still accurately representing the system state.

2.2 Design Considerations in DWSN

Computation Constraints

Each agent has limited memory and computational capability to send from a light data such as tri-axial data to a relatively heavy data like high quality video frames. Computing and processing video frames at the microcontroller would be challenging unless the microcontroller has a graphical processing unit (GPU). The video can be transferred to the host computer for more analysis such as object detection, classification, or segmentation using an OpenCV library but collected frames contain delay during streaming and sending them back to the agent will be inefficient. A wireless sensor network has to divide the task optimally, understanding computation constraints and allocating the task is important in a wireless sensor network.

Communication and Sensing Coverage

Communication coverage has to be considered depending on the applications. For instance, reliable monitoring and analysis of unspecified motion and untested environment will require agents to reach the other agent immediately when there are dynamic changes in environment. Also, the communication is more energy-intensive compared to the coverage in that much more energy is needed to cover a larger communication area. Besides the communication coverage, the sensing coverage is also a critical factor as it will decide the number of necessary agents to be deployed as well
as the cost of the sensors. For instance, a long distance RFID detection is much more costly compared to the regular RFID system detecting within a couple of inches by harvesting RF energy.

**Scalability**

A large number of agents deployed would create challenging issues such as time synchronization and information traffic. In order to cover a larger sensing and network area, deployment of multiple agents is essential when the network scalability is guaranteed. To be a scalable system, the size and cost has to be minimized while keeping the functionality of proposed applications. However, there is a tradeoff between the size and the power as well as the motors, i.e., the battery power and motor torque are relatively proportional to the size. The primary issue for scalability is the network traffic due to the limited bandwidth and the constrained number of connected agents. For instance, data rate of the ZigBee limits is about 250kbps. Even though the network size of the ZigBee is 64K, it cannot be deployed with a complex configuration due to the data traffic.

**Energy Contraints**

Each agent is generally powered by a battery with a finite amount in energy. The sensors, actuators, processors, wireless modules, and other electronics draw energy from the battery and the improvement of energy storing in a small size is slow which means that Moore’s law does not apply to batteries. Therefore, efficient and optimal power management is essential in wireless sensor networks to make the system function well until a battery replacement or recharge is needed.
2.3 Taxonomy of DWSN

To understand the systems following the design criteria under DWSN, three major aspects of DWSN, i.e., data input, communication, and access control for multiple modular blocks, are addressed. Fig. 2.2 shows the taxonomy of the DWSN elaborated at the software level rather than the hardware level such as structure, package, and functions. The hardware can be chosen based on the selected DWSN features.

![Image of DWSN taxonomy diagram]

Figure 2.2: Taxonomy of DWSN at the software level.

Data input

The data input is divided into periodic data transfer and event-driven data transfer. Periodic data transfer can deliver an adequate amount of information but it is difficult to avoid data traffic on a large scale. On the other hand, data traffic is negligible in event-driven data transfer assuming enormous events do not occur at the same moment. Also, the event driven data transfer reduces the amount of information by not sending redundant or repetitive information.
Synchronization

The synchronization in a network aims to provide a common timescale for local clocks of nodes in the network. However, all hardware clocks are imperfect, therefore the local clocks of nodes may drift away from each other over time, so observed time or durations of time intervals may differ for each node in the network. To deal with this problem, master-slave and peer-to-peer synchronization method are considered. A master-slave synchronization assigns one node as the master and the other nodes as slaves. In general, slave nodes send burst data with a short time period and the master node receives them. However, the master node requires CPU resources proportional to the number of slaves. Mock et al. [9] have adopted the IEEE 802.11 clock synchronization protocol due to its simple, non-redundant, master/slave structure. Ping’s Protocol [10] also adheres to the master-slave synchronization. On the other hand, peer-to-peer structure can communicate directly with every other node in the network. Without assigning master-slave or hierachical structure, communicating between homogeneous modules are challenging as there is no priority for communication order. With variable-length codes, self-synchronization was enabled for image transmission [11]. However, a common problem with this variable-length source code is that channel errors that may cause synchronization slippage, and self-synchronizing Huffman codes were designed that resynchronizes the decoder regardless of the synchronization slippage [12].

Network type

Network type is important in terms of security and energy efficiency. Point to point, broadcast, and hybrid communications are allowed in many existing wireless modules and can be selected for specific needs. This network type can refer to network topology or structure of network. Point to point is a unimpeded communication between the two endpoints, broadcast transmits a message to all recipients simultaneously, and
hybrid communication stands for all other possible networking types mixing different
network topologies.

Multiple access control

Utilizing the API (Application programming interface) mode, dynamic channel selection is enabled and hence the multiple targets are accessed and controlled simultaneously. The advantage of the static channel is the reliable communication with fast response time in sending and receiving data whereas the dynamic channel requires time for linking and stabilizing for channel updates. The benefits of the dynamic channel is its resistance to a communication failure in large scale by allocating a new channel dynamically.

2.4 Time Synchronization

In DWSN, the time synchronization is necessary to allow reliable communication, collective signal processing, sensor and source localization, data aggregation, and distributed sampling. For example, synchronization has to be established in shared communication channels to integrate motion sensors collected from each agent to draw the motion data based on the global clock. Unlike the centralized wireless sensor network, the distributed sensor network does not share a certain clock but rather each agent uses their own internal clock in the embedded microprocessor. Synchronization for a given application can be determined by the following: 1) the resources of available communication, 2) the degree of accuracy, 3) the time budget for achieving the synchronization, and 4) the longevity of synchronization (whether synchronized all the time or just when needed). First, the resources of available communication include optical, RF, or ultrasound signal transceivers which can establish the synchronization in different communication sources. For instance, the RF-based synchronization
is well developed as smartphone-like mobile platform technology has been developed innovatively along with the communication protocols such as 3G, 4G, and LTE. One must determine whether to use the built-in synchronization techniques or to develop a new synchronization protocol. However, optical or ultrasound signal-based synchronization protocols are not well developed. Second, the degree of accuracy needs to be considered whether it produces errors at the range of milliseconds or microseconds. Third, the time allowed for synchronization must be investigated for each specific application. For instance, when two agents need to synchronize frequently, e.g., mobile robots, the time for synchronization has to be short. On the other hand, the time for synchronization does not have to be as short for applications with multiple agents manipulated by humans because the time frame for data collection is much smaller than the actual human motion, e.g., TAG-Games. Lastly, the longevity of synchronization needs to be decided. That is, it must be decided whether the synchronization method will be applied continuously or only when needed.

2.5 Sensors for Wireless Communication and Localization

From a few cents to a thousand dollars, there exist various sensors with different functions and capabilities on the market. To measure distance, infrared-based, laser-based, and ultrasonic-based sensors are frequently used. Each method has advantages and disadvantages when used alone, and therefore, sensor fusion or integration is often applied [13]. Although laser-based distance measurement is more precise, the price for system integration is much more expensive than an infrared or ultrasonic based method. While both infrared and ultrasonic sensors are relatively inexpensive, infrared sensors are smaller and therefore more frequently used than ultrasonic sensors in small sized systems. Regarding orientation detection, three types of sensors are
widely used: accelerometer, gyroscope, and magnetometer. To measure the absolute rotational angle, a combination of an accelerometer and a magnetometer can be used as they detect gravitational direction and polarity of the earth, respectively, which are constant. However, each sensor has inherent limitations at measuring the rotational angle. The rotation perpendicular to the gravitational direction is not measurable by an accelerometer. Gyroscopic drift is also caused by numerical integration over time. Therefore, a sensor fusion or integration method, such as a Kalman filter or complementary filter, is often used to compensate the weakness of a single sensor. Global Positioning System (GPS) and vision systems are also frequently used for system localization and position detection.

Table 2.1: Review of frequently used sensors

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Measurement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Infrared</td>
<td>Distance</td>
<td>Measures the time of infrared signal reflected by object</td>
</tr>
<tr>
<td>Laser</td>
<td>Distance</td>
<td>Measures the time of laser signal reflected by object</td>
</tr>
<tr>
<td>Ultrasonic</td>
<td>Distance</td>
<td>Measures the time of ultrasonic signal reflected by object</td>
</tr>
<tr>
<td>Accelerometer</td>
<td>Orientation</td>
<td>Measurement of acceleration forces in the 3D space</td>
</tr>
<tr>
<td>Gyroscope</td>
<td>Orientation</td>
<td>Measurement of angular velocity in the 3D space, numerical integration is need to get rotational angles</td>
</tr>
<tr>
<td>Magnetometer</td>
<td>Orientation</td>
<td>Measurement of magnetic field</td>
</tr>
<tr>
<td>GPS</td>
<td>Location</td>
<td>Provides location and time information using satellites</td>
</tr>
<tr>
<td>Vision</td>
<td>Location</td>
<td>Provides the position in the camera view and location using stereo vision</td>
</tr>
</tbody>
</table>

There exist several communication protocols for establishing a wireless network
Table 2.2: Comparison among different wireless communication protocols in terms of bandwidth, transmission range, power consumption, network size/type, cost, and system complexity.

<table>
<thead>
<tr>
<th>Properties</th>
<th>ZigBee</th>
<th>Bluetooth</th>
<th>Wi-Fi</th>
<th>NFC</th>
<th>IrDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission Range</td>
<td>&lt;300 ft</td>
<td>&lt;30 ft</td>
<td>&lt;300 ft</td>
<td>&lt;1ft</td>
<td>&lt;30 ft (line of sight)</td>
</tr>
<tr>
<td>Power Consumption</td>
<td>Very low</td>
<td>Medium</td>
<td>High</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Network Size</td>
<td>64000</td>
<td>1 to 7</td>
<td>10</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Network Type</td>
<td>Ad-hoc</td>
<td>Ad-hoc</td>
<td>Point to hub</td>
<td>Point to point</td>
<td>Point to point</td>
</tr>
<tr>
<td>Cost</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>System Complexity</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Signal Type</td>
<td>RF</td>
<td>RF</td>
<td>RF</td>
<td>RF</td>
<td>IR</td>
</tr>
</tbody>
</table>

using radio or IR, including ZigBee, Bluetooth, Wi-Fi, NFC, and IrDA. Table 2.2 summarizes these protocols in terms of bandwidth, transmission range, power consumption, network size and type, cost, system complexity, and signal type.

Most localization methods first estimate distances or angles between unknown sensors and anchor sensors. Then, the locations of the unknown sensors are calculated with some geometric algorithms. Thus, the most important elements for sensor localization are distance measurement, angle measurement, and geometric constraints. In order to determine the best sensors to detect the object or estimate the distance between objects, three sensor concepts have been analyzed in different aspects such as cost, range, accuracy, and robustness to environmental factors (Table 2.3). Ul-
Table 2.3: Comparison of distance measuring sensors

<table>
<thead>
<tr>
<th>Pros</th>
<th>Ultrasound</th>
<th>LIDAR</th>
<th>RADAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pros</td>
<td>Robust to target or object color (robust to light)</td>
<td>High accuracy</td>
<td>Can estimate the distance to an object although line of sight is screened.</td>
</tr>
<tr>
<td>Cons</td>
<td>Shape of the target affects measurement.</td>
<td>Cannot detect object out of sight.</td>
<td>Requires transmitter and receiver both, needs filtering.</td>
</tr>
<tr>
<td>Range</td>
<td>&lt;10m</td>
<td>&lt;10m</td>
<td>&lt;100m</td>
</tr>
</tbody>
</table>

Ultrasound uses sound waves above the range of audible sound to humans to measure the position of objects within a 2-3 m range. LIDAR (LIght Detection And Rang-ing) uses light such as infrared or laser for positioning and measuring distances by measuring the difference between the transmitted pulse and reflected pulse. RADAR (RAdio Detection And Ranging) uses long wavelength electromagnetic waves for detecting, locating, tracking, and identifying moving and fixed objects at a long range. Localization techniques with wireless communication and distance measuring sensors include 1) received signal strength indication (RSSI), 2) time of arrival (ToA), 3) time difference of arrival (TDoA), 4) angle of arrival (AoA), 5) triangulation, and 6) trilateration.

- **Received Signal Strength Indication**: The power of the received radio signal falls off exponentially as distance increases, and the receiver can measure this attenuation based on Received Signal Strength Indication (RSSI) in order to estimate the distance from the transmitter. This method has been used primarily for radio frequency signals.

- **Time of Arrival**: The distance between the transmitter and the receiver may be estimated based on the speed of the wave propagation and the measured
time for a radio signal to travel between two sensor nodes. This method can be applied not only for the radio frequency signal but also acoustic, infrared and ultrasound signals. ToA offers a high level of accuracy, but also requires relatively fast processing capabilities in sensor nodes to resolve the timing differences for fine-grained measurements.

- **Time Difference of Arrival**: Time Difference of Arrival (TDoA) uses different communication media at different speeds. For example, a radio signal is very fast compared to ultrasound or other signal types, therefore, the radio signal is used for synchronization between the transmitter while the receiver and the ultrasound signal is used to estimate the distance between them.

- **Angle of Arrival**: Angle of Arrival can be used to estimate the relative orientation of a transmitter and receiver combined with smart antenna arrays to measure the angle at which the signal arrives. Although the system with smart antenna arrays can enhance the capacity of wireless links through the combination of array gain, it may not be applicable to sensor networks due to the high cost of a complex antenna array and interference in a large number of nodes.

- **Triangulation**: Triangulation is a geometric technique that uses the angles of arrival to determine the location of sensors. With the angle of each anchor sensor known, the locations of unknown sensor nodes are estimated with trigonometry laws.

- **Trilateration**: Trilateration uses distances between three anchor sensors and one unknown sensor to determine the unknown sensor’s location. An unknown sensor is uniquely located when at least three reference points are associated with it in a two-dimensional space. The location of the unknown sensor is estimated by calculating the intersection of three circles.
RSSI only gives the distance without orientation information, hence, the location can be identified by the trilateration method with three distance estimates from the RSSI of three anchors. ToA, TDoA, and AoA require high speed processing capabilities to have high resolution accuracy. Therefore, these methods would not be adequate for the low processing capability of a microcontroller.
Chapter 3

Algorithms

A DWSN system adopts a large number of agents and the overall networking system becomes complex. Since the DWSN system addresses peer-to-peer topologies to connect many agents reliably, a broadcasting network is considered as one of the fundamental operations in wireless network. However, transmitting the messages to all agents after receiving the messages can impose a large amount of redundancy in traffic and a significant waste of limited bandwidth. This is not only applicable for sensor networks, robotic applications such as self-reconfigurable or self-organizing robotics also require communication coordination between robotic modules. This communication coordination can be realized with existing network protocols such as IrDA but it requires additional hardware and control by request to be synchronized. In an effort to develop low-cost, reliable, and scalable synchronization, a self-synchronization algorithm has been proposed and applied to the applications in this study. For heterogeneous features in the network (i.e., IR for local network and RF for global network), a hybrid wireless network was developed to reduce data traffic and maximize network coverage with given resources. In addition to the network algorithms, motion and assembly detection algorithms are developed in this study. The details of the algorithm are described in the following sections.
3.1 Self-Synchronization

Unlike the RF-based synchronization protocols, optical or ultrasound signal-based synchronization protocols are not well developed. For short-range wireless technologies with infrared, IrDA technology has been used widely \[14\]. However, the cost of IrDA (§3-5) and the added complexity of electronical design require additional efforts. Also, the ultrasound transceivers can easily interfere with each other when multiple modules are used, and the module is not small enough to be embedded in the proposed applications.

The synchronization algorithm requires two IR sensors and two photo transistors for bidirectional signal transfer and receiving. Assuming two IR sensors, A and B, are communicating with each other, B listens while A talks, and A listens while B talks if two ports are fully synchronized. However, as shown in Fig. 3.1, transmitted signals by A and B may be conflicted if they both talk or listen at the same time. The self-synchronization method uses an internal clock, transmitted pulse, and received pulse. By comparing these three, one port is determined to listen while the other talks. This bit-shift strategy is performed as follows, for the port A:

\[
I_A = [1, 2, 3, \cdots, 30, 31, 32]; \quad (3.1)
\]
\[
T_A = [0, 0, \cdots, 0, 0, 1, 1, \cdots, 1, 1]; \quad (3.2)
\]
\[
R_A = [\ast, \ast, \cdots, \ast, \ast, \ast, \ast, \cdots, \ast, \ast]. \quad (3.3)
\]

\(I_A\) is the 32-bits cyclic bit number index generated by the internal clock, \(T_A\) is the transmitted binary data, and \(R_A\) is the received signal from the other port. The pulse train from \(T_A\) is generated from the IR photo diode according to the bit number of \(I_A\). When the two ports, A and B, are within the local communication range, \(R_A\) receives the data from \(T_B\) as well as the reflected signal from \(T_A\). Due to the pull-up resistors attached to the photo transistors, \(R_A\) is set as one initially or when no IR signals
are received and changes to zero when IR input is detected. The decision for shifting
either forward or backward for synchronization can be made stochastically; however,
this process can take a long time. For every 31st bit of the 32 bits (4 bytes), the binary
train moves one step forward or two steps forward based on the received data. If the
$\sum_{n=bit1}^{bit8} R_A(n)$ is smaller than the $\sum_{n=bit9}^{bit16} R_A(n)$, then the pulse train proceeds one
step forward whereas the pulse train moves two steps forward when $\sum_{n=bit1}^{bit8} R_A(n)$
is bigger than the $\sum_{n=bit9}^{bit16} R_A(n)$. When the two transmitted signals are the same, a
random shift is applied to each port so that it makes partial synchronization. When
the synchronization is completed, IR communication data packets are replaced in the
last two bytes of the $T_A$. Fig. 3.1 illustrates this process.

Figure 3.1: Short-range communication mechanism via IR sensors and self-
synchronization scheme between the ports A and B. The gray bar indicates talking
bytes while the white bar refers to listening bytes.

Algorithm 1 is implemented in each agent so in the case that A encounters
Algorithm 1 Self Synchronization Technique for IR Sensors

1: while Not synchronized
2:   if bit n=31 then
3:     if $\sum_{bit1}^{bit8} R_A(n) < \sum_{bit9}^{bit16} R_A(n)$ then
4:       bit n ← bit n+1
5:     else if $\sum_{bit1}^{bit8} R_A(n) = \sum_{bit9}^{bit16} R_A(n)$ and $\sum_{bit1}^{bit16} R_A(n) = 16$ then
6:       bit n ← bit n+ rand(1,2)
7:     else if $\sum_{bit1}^{bit8} R_A(n) = \sum_{bit9}^{bit16} R_A(n)$ and $\sum_{bit1}^{bit16} R_A(n) = 0$ then
8:       Synchronization is done!
9:   end if
10: end if
11: bit n ← bit n+1
12: if bit n = 33 then
13:   bit n ← bit 1
14: end if
15: end while

$\sum_{bit1}^{bit8} R_A(n) > \sum_{bit9}^{bit16} R_A(n)$, B will encounter $\sum_{bit1}^{bit8} R_B(n) < \sum_{bit9}^{bit16} R_B(n)$. In this case, agent B would shift one bit forward and agent A would not shift. When the two transmitted signals are the same, a random shift is applied to each port so that it makes partial synchronization. When the synchronization is completed, IR communication data packets are replaced in the last two bytes of $T_A$. When the synchronization is completed, IR communication data packets are replaced in the last two bytes of $T_X$ (Fig. 3.2).

![Figure 3.2: IR communication data packets (two bytes of talking)](image-url)
3.2 Hybrid Wireless Network Strategy using IR and RF

To establish a reliable and efficient wireless network among multiple agents, a hybrid algorithm was developed that uses both IR and RF technologies. The IR-based method is adopted for precise short-distance communication within less than an inch for synchronization and the RF-based ZigBee technology is used for localization and long-distance communication as shown in Fig. 3.3. The general communication range of ZigBee is up to 300 feet whereas the range for localization by distance measurement using RSSI is limited to 60 inches. Within this range, a linear relationship between the RSSI data and physical distance is observed [15]. For IR-based communication, pairs of an IR emitting diode and a phototransistor (QRD1114) were used instead of commonly used IR data association (IrDA) devices. These optical IR sensors are cheaper and easier to implement without requiring many additional hardware components such as the case with IrDA devices.

![Figure 3.3: Hybrid wireless network scheme](image)

Fig. 3.3 demonstrates the network scheme for a mobile robot. Short-distance communication using IR sensors and long-distance communication with RF modules: inch-scale and ft-scale. The RF signal can reach up to 300 ft while reliable distance measurement for localization can be made within a distance of 60 inches. The infrared
and RF modules can be selected depending on the dimension of the experimental set-up. More powerful RF modules or low-frequency modules such as 433MHz RF modules can transmit and receive the signal as well as estimate at a longer distance. Also, the more energy intensive infrared LED can also reach farther than a few inches. To compensate the errors of the RF signal due to the interference by the robots, infrared can be used. Likewise, RF-based distance sensing can compensate for a longer distance which infrared cannot reach.

3.3 Motion Detection Algorithm

In order for a triaxial accelerometer to be used to detect motion, it needs to filter out the noise caused by hand movements. The low-cost capacitive accelerometer used in our study features signal conditioning, a 1-pole low-pass filter, temperature compensation, and g-selection which allows for the selection among 4 sensitivities: 1.5g, 2g, 4g, and 6g. The highest sensitivity of this accelerometer is 800mV/g occurs when 1.5g is selected. There are three output ports on the accelerometer which provide analog acceleration signals along the $x$, $y$, and $z$-axes. These analog signals are then transformed to digital signals through the analog-to-digital converter in the microprocessor (ATmega328). When an accelerometer is in a static state, it can detect angular displacements by measuring static accelerations due to gravity which range between $-g$ and $+g$. As shown in Fig. 3.4, an arbitrary tilt angle can be obtained using a single-axis acceleration data. Using a triaxial accelerometer (Fig. 3.5), tilt angles along the $x$- and $y$-axis can be calculated based on three acceleration data ($x_{acc}$, $y_{acc}$, $z_{acc}$) by [16]:

$$
\theta_{pitch} = \tan^{-1}\left(\frac{x_{acc}}{\sqrt{(y_{acc})^2 + (z_{acc})^2}}\right);
$$
\[ \theta_{\text{roll}} = \tan^{-1}\left( \frac{y_{\text{acc}}}{\sqrt{(x_{\text{acc}})^2 + (z_{\text{acc}})^2}} \right). \]

If Matlab\textsuperscript{TM} is used for numerical computations, two built-in functions compute the inverse of the tangent, ‘atan’ and ‘atan2’. For a real \( x \), ‘atan(x)’ is in the range \([-\pi/2, \pi/2]\). ‘atan2(y, x)’ gives the value of \( \theta \), such that \( \sin \theta = y \) and \( \cos \theta = x \). The value of \( \theta \) lies in the interval \([-\pi, \pi]\).

Figure 3.4: Tilt angle measurement: an accelerometer can detect an arbitrary tilt angle by measuring the static acceleration due to gravity.

Figure 3.5: Pitch and roll angles measured relative to the global horizontal frame using a triaxial accelerometer.

Detecting tilt angles while the accelerometer is in motion is challenging. To analyze the effect of dynamic acceleration, a simple block test was conducted by manually rotating a block 90° and −90° about the \( x \)-axis repeatedly at different frequencies.

Fig.3.6-(a) is generated by rotating the block 5 cycles in approximately 50 seconds (0.1 Hz). Fig.3.6-(b), (c), and (d) show the results when the block is rotated at 0.5
Figure 3.6: $x_{acc}$ captured from the triaxial accelerometer versus time while rotating about the $y$-axis.

Hz (5 cycles per 10 seconds), 1.0 Hz (10 cycles per 10 seconds), and 2.0 Hz (20 cycles per 10 seconds) respectively. According to these experimental results, the overall influence of dynamic acceleration is trivial for slow motions ($\leq 1.0$ Hz) while the error increases and becomes significant in higher frequency domains. This is corroborated by a study by Bernmark which showed that the accelerometer is not influenced by dynamic motion within 0.75 Hz motion [17]. For fast motions (Fig. 4.24(d)), $x_{acc}$ from the accelerometer exceeds $\pm 1g$ when block reaches the turning points ($\pm 90^\circ$) showing the influence of dynamic accelerations.

Fig. 3.7 represents the amplitude in unique frequency domains. Although every user behaves uniquely and has different frequency responses, humans have a limited speed and range of motions for manipulating rigid objects. Moreover, the target end-users of TaG-Games include preschool-aged children with developmental disabilities, the elderly, and patients with traumatic brain injuries or mental illnesses. Therefore,
the blocks will be mainly used within the low frequency ranges while successfully retrieving their rotational information and providing reliable data on how they are manipulated by a user. By applying a digital low-pass filter at the cutoff-frequency, which can be empirically determined through calibration, the animation of the blocks can be further improved. Using this algorithm, SIG-Blocks could provide additional information on behaviors, such as fast and repetitive motions, shaking, and hand vibrations through the accelerometer data.

3.4 Assembly Detection Algorithms

The assembly detection algorithm is mainly effective for TAG-Games but swarm robots can utilize the algorithm to detect the configuration of the robot in a global view.

When any two blocks are assembled, the assembly configuration is captured by
a pulse train via local inter-block communication using IR sensors. The pulse train involves the following information: (ID1, SIDE1, TOP1, ID2, SIDE2, TOP2). ID1 and ID2 are the ID numbers of the two blocks assembled; SIDE1 and SIDE2 are the surfaces on the blocks facing each other; and TOP1 and TOP2 are the surface of each block that is facing upward. Considering the low response and signal reading time in the photo transistor and the microcontroller’s input port, 1 millisecond per bit for the pulse train was selected.

For each assembly detection, the configuration matrix, $A$, is generated by labeling the block’s IDs and side IDs. For the example shown in Fig. 3.8, $A$ is a $6 \times 6$ matrix consisting of four $3 \times 3$ matrix blocks where each matrix block contains the SIG-Block ID in the (2,2) entry and four sides information in the (1,2), (2,1), (2,3), and (3,2)
entries. Before and after assembly, $A$ changes as follows:

$$
A := \begin{bmatrix}
* & 0 & * & 0 & * \\
0 & 1 & 0 & 0 & 2 & 0 \\
* & 0 & * & 0 & * \\
0 & 4 & 0 & 0 & 3 & 0 \\
* & 0 & * & 0 & * 
\end{bmatrix} \rightarrow \begin{bmatrix}
* & 0 & * & 0 & * \\
0 & 1 & 3 & 1 & 2 & 0 \\
* & 0 & * & 0 & * \\
0 & 4 & 1 & 3 & 3 & 0 \\
* & 0 & * & 0 & * 
\end{bmatrix}
$$

For an $n \times m$ assembly configuration in 2D, $A \in \mathcal{R}^{3n \times 3m}$ consisting of $n \times m$ matrix blocks. It can be further extended to 3D configurations by defining $A$ as a 3D matrix consisting of $3 \times 3 \times 3$ matrix blocks.
Chapter 4

Primary Application: TAG-Games

4.1 Overview

Skilled cognitive performance is a highly complex process that involves perception, memory, reasoning, problem solving, and executive functions [18]. A fine-tuned orchestra of cognitive skills joins in concert to produce general intellectual and learning capabilities [19, 20, 21]. The ability to measure these components efficiently is essential for academic programming and occupational placement. Additionally, precise measurement of these components may enable a better understanding of the cognitive impairments associated with many disorders, including brain injuries, intellectual disabilities, dementia, psychiatric/mental disabilities, and other neurological conditions [22, 23]. Finally, given the increased longevity of human beings, assessment of age-related cognitive decline, whether mild or severe, in a variety of cognitive and motoric components is of critical importance [24].

The face of cognitive assessment has evolved and become more diverse over the past several decades. The Wechsler tests are among the earliest and most widely accepted instruments in the field [25, 26]. When verbal constraints are in place or the clinician chooses to test non-verbal intelligence, the Raven’s Progressive Matrices
is often selected [27]. The Raven’s Progressive Matrices rely on a series of perceptual, analytic reasoning problems presented in a matrix format [28]. Sometimes, cognitive assessments include both cognitive processing and motor responses in the procedures with physical objects often employed in the assessment tasks [29]. Geometric blocks, such as cubes, are well defined for observing manipulation patterns and developmental transformations, and therefore often employed as an evaluation tool for understanding how children develop spatial cognition and fine motor skills [30]. As an example, the Block Design subtest in the Wechsler tests employs a set of cubes to examine how quickly and accurately the examinee assembles the cubes to match the item configuration. Recent studies found that achievement of certain developmental milestones can be captured by performance on manipulative tasks, such as piling and assembling blocks and inserting objects in holes [31, 32]. Previous studies also found that the blocks can be used for cognitive rehabilitation of patients with cognitive and/or fine-motor skill deficiencies due to traumatic brain injury (TBI), stroke, cerebral palsy, Huntington’s disease, and dementia [33, 34, 35, 36, 37]. Using geometric blocks, or any other physical objects, in cognitive assessment requires the administrator’s close observation of the person’s performance while administering the
test and recording measurable data (e.g. accuracy and speed for each test item). Compared to a paper-pencil test, this process can be more labor intensive and error prone.

This study presents the SIG-Blocks (Sensor-Integrated Geometric Blocks) technology developed for tangible, computerized assessment of cognitive skills using physical blocks (Fig. 4.1). The computer-based games using SIG-Blocks are called TAG-Games (Tangible Geometric Games). SIG-Blocks detect overall manipulative motions applied by the player and communicate with each other to detect assembly configuration. The system features independent wireless communication between each SIG-Block and a local host computer using radio frequency and block-to-block communication using infrared (IR) signals. For reliable and efficient block-to-computer (global) and block-to-block (local) communications, DWSN algorithms, including self-synchronization and hybrid-sensing techniques, were implemented. The SIG-Blocks technology is designed for automated, real-time, and remote assessment of cognitive skills as well as hand-eye coordination and fine-motor control proficiency. It aims to address the following limitations in the traditional assessment methods: 1) high cost due to clinician/administrator time, 2) limited amount and types of measurable data by a human administrator, 3) subjectivity and human measurement errors, 4) limited accessibility to professional services for people living in hard-to-reach areas, and 5) difficulty in addressing individual differences in cognitive skills, age, and developmental status due to lack of flexibility in test items.

Using SIG-Blocks, a variety of TAG-Games can be designed at the software level targeting specific areas of cognition and executive functions. For example, SIG-Blocks can be used simply to automate an existing test, such as the Block Design subtest in the Wechsler tests by simply changing the block cover images and designing the TAG-Games accordingly. In this study, three types of TAG-Games were designed and evaluated with target assessments of cognitive problem-solving, hand-eye coordina-
tion, and working memory skills: TAG-Game\textsuperscript{A} (block assembly game), TAG-Game\textsuperscript{S} (shape-matching game), and TAG-Game\textsuperscript{M} (memory game). For the preliminary utility evaluation, TAG-Games was tested on 86 university students between the ages of 18 and 30. Reliability was tested via split-half and test-retest reliability evaluations. Preliminary utility of TAG-Games for assessing target cognitive skills was validated by comparing the TAG-Games performance data with the scores from three subtests of the Wechsler Adult Intelligence Scale 4th Edition (WAIS-IV) (i.e., Block Design (BD), Matrix Reasoning (MR), and Digit Span (DS)). For each TAG-Game, a computational measure of play complexity ($C_{\text{play}}$) was defined that quantifies the difficulty associated with each test item controlling for the number of blocks and the geometric complexity of the images used on the blocks. $C_{\text{play}}$, once fully validated, can be used to customize the test items in order to address the individual/group differences and potentially provide more sensitive assays when combined with outcomes from the standardized methods. $C_{\text{play}}$ and participants’ performance data on each type of TAG-Games were compared to validate the proposed complexity formulas.

### 4.2 Related Works on Technology-Embedded blocks

Owing to the recent advancement in sensors and wireless communication technologies, cognitive assessment using technology-embedded block systems has been receiving considerable attention for the past few decades. In this section, the available technologies will be briefly reviewed, what each system attempts to accomplish, and how SIG-Blocks technology differs from them and addresses the limitations of the previous systems. For starters, there have been several technology-embedded block systems developed for a broad range of education, entertainment, and research purposes. Beginning with the earliest variant, AlgoBlock was developed in 1995 for children to learn programming via tangible manipulation of the blocks [38]. Every
AlgoBlock has special semantics that enables children to write their own programs while effectively facilitating children in interactive learning of computer programming. A second system is called the Tangible Programming Bricks with a built-in microprocessor that was developed for children to learn programming language via tangible manipulation of the bricks [39]. The Electronic Blocks is a third system that was developed for young children to build simple computer programs by stacking the blocks with embedded electronic circuits [40].

A fourth system is the Cognitive Cubes developed for cognitive and constructional assessment [47]. The Cognitive Cubes are equipped with male-female connectors for forming a 3-D assembly and a network topology with a special base cube that is wired to the host computer. The roBlocks is a fifth system that was developed to expose users to a variety of advanced engineering concepts, such as kinematics, feedback, and distributed control by using the blocks [43]. The working principle of roBlocks is quite similar to Electronic Blocks. A sixth system is Cubelets, a modular robotic kit consisting of sensor, logic, and actuator blocks [48]. By assembling different types of Cubelets, a user can build a robot with specific communication and
sensing capabilities. An eighth system is called the Learning Cube which is a digitally augmented system enriched with LED displays and an internal speaker [44]. The Learning Cube is capable of gesture recognition using an accelerometer in order to create an interactive learning interface. As a ninth system, Sifteo is another sensor-embedded block system with a small LCD screen on its top face [49]. They can sense and communicate with each other and detect tilt and shake motions. A tenth system the multi-agent system developed for educational games for children with autism [46]. It displays its active state through color and light intensity by detecting the status of neighboring blocks with infrared (IR) communication.

Table 4.1 compares the existing sensor-embedded blocks and SIG-Blocks in terms of the global (block-to-computer) and local (block-to-block) communication strategies, embedded sensors, measurable data, and online feedback capability. The technology embedded in SIG-Blocks is not very different from other work done, however, SIG-Blocks has very unique features which distinguish it from other available systems. First of all, the SIG-Blocks system is operated in a fully decentralized manner without a hierarchical or master-slave structure as used in Cognitive Cubes and therefore is fault-tolerant. Another strength of our system is that SIG-Blocks is very universal and target oriented. The cover images of the SIG-Blocks can be easily replaced for each target application. Further, most existing systems are designed for tangible gaming or educational purposes but appear to lack the necessary recording functions which are essential for the assessment functions. SIG-Blocks with TAG-Games are specifically designed to provide feedback on the player’s performance and behavior via real-time, wireless, and bidirectional communications. Only a few existing systems, such as Learning Cube [44] and Sifteo [49], demonstrated wireless block-to-computer communication capability, both based on radio frequency (RF).

Among these existing systems, Sifteo has the most comparable features to SIG-Blocks in terms of sensing and communication capabilities. Sifteo adopts a gateway
Table 4.1: Communication strategies, sensing capabilities, and homogeneity in the design of existing technology-embedded blocks and SIG-Blocks and their application domain.

<table>
<thead>
<tr>
<th>Block systems (Year)</th>
<th>Block-to-computer communication</th>
<th>Inter-block communication</th>
<th>Embedded sensors</th>
<th>Measurable data</th>
<th>Data display</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlgoBlock [38] (1995)</td>
<td>Wire</td>
<td>Electro-mechanical</td>
<td>-</td>
<td>Binary assembly detection</td>
<td>Online (PC)</td>
</tr>
<tr>
<td>Electronic Blocks [40] (2001)</td>
<td>-</td>
<td>Electro-mechanical</td>
<td>Light, sound, movement</td>
<td>Light, sound, touch</td>
<td>Online (LED)</td>
</tr>
<tr>
<td>Multi-agent system [46] (2009)</td>
<td>-</td>
<td>Optical</td>
<td>Infrared</td>
<td>Assembly detection</td>
<td>Online (LED)</td>
</tr>
<tr>
<td>SIG-Blocks</td>
<td>Wireless (RF)</td>
<td>Optical</td>
<td>Triaxial accelerometer, infrared, infrared</td>
<td>Triaxial accelerations, assembly, orientation</td>
<td>Online (PC)</td>
</tr>
</tbody>
</table>
called Sifteo-base which can be connected to a host computer using a USB cable for downloading new games and then wirelessly communicates with the rest of the Sifteo blocks. The major differences in SIG-Blocks compared to Sifteo includes: 1) the low-cost infrared communication method, 2) direct wireless communication between each block and a host computer without requiring a gateway, 3) continuous sensor-data collection from each block, and 4) applicability for both 2-D and 3-D construction tasks. First, while Sifteo adopts four IrDA transceivers ($3-5 each), SIG-Block uses six pairs of a IR LED and a photo transistor ($0.50 each). Second, bi-directional wireless communication between each SIG-Block and a host computer is established. The computer can wirelessly control the mode of the blocks for different games and each block sends the collected sensor data including orientation and assembly information to the computer. Third, continuous data collection for assessment purposes can be made. Lastly, the cube design of SIG-Blocks enables 3-D construction games. The LCD display on Sifteo makes it appealing and suitable for entertainment and educational applications. While the current version of the SIG-Blocks was designed for cognitive assessment and thus clear geometric patterns were selected, different versions of SIG-Blocks (e.g. iSIG-Blocks [50]) with augmented sensory feedback mechanism can be developed while keeping the core internal electronics module.

4.3 Hardware Development

4.3.1 SIG-Blocks

Our initial prototype consists of four homogeneous cubes (Fig. 4.3) [13]. Each block contains a triaxial accelerometer (MMA7260), a microprocessor (ATmega328), six reflective optical sensors, an XBee wireless module, and four rechargeable AAA batteries. The block is $2.7 \times 2.7 \times 2.7$ inch$^3$ and weighs 355g including batteries. The triaxial accelerometer provides two angular accelerations along the roll and pitch axes.
The optical sensors, which are installed on each contact surface of the block, are pairs of an infrared emitting diode and an infrared phototransistor which detects the reflected signal. By combining the data from the accelerometer and optical sensors, assembly configurations of the blocks can be reconstructed and displayed in the GUI. Each block has a unique ID which enables independent and simultaneous communication of all the blocks with the host computer. By attaching different shapes or colors on each surface of the blocks, the SIG-Blocks can be used for various types of play or tests with varying complexity. Each block is powered with four 1.2 Volt rechargeable batteries for which the running time is approximately 4-5 hours.

The newly developed SIG-Blocks have the following improved features: 1) newly added infrared communication, 2) bidirectional XBee communication, 3) modified printed circuit board (PCB) design, and 4) improved durability. The block’s size and weight are reduced to 2.05 inches in length along each side and 82g. Each block contains a triaxial accelerometer (ADXL335), a microprocessor (ATmega328), six reflective optical sensors, and an XBee wireless module. It is powered by two 3.7 Volt, 350mAh lithium-ion batteries which are rechargeable through a USB connector. It

Figure 4.3: SIG-Blocks: interactive creation blocks designed for tangible interactive games.
takes less than 30 minutes to recharge one cube and the operating time is approximately 3.5 hours. Core electronics are embedded in three stacks of PCB layers and are encased in a cube block made of mat-board. The reflective optical sensor installed at the center of each block’s contact surface detects the reflected infrared signal. By combining the data from the optical sensors and the accelerometer, assembly configurations of the blocks can be reconstructed for real-time feedback [13]. Also, the local communication between the assembled blocks is enabled through these optical sensors with proposed protocols as described in Chapter 3.1. These optical sensors are relatively fragile and may require replacement periodically. Female sockets were used to place these optical sensors to ease the replacement process.

![Figure 4.4: Six distinctive images with 1-, 2-, and 4-fold rotational symmetry.](image1)

Figure 4.4: Six distinctive images with 1-, 2-, and 4-fold rotational symmetry.

![Figure 4.5: Customized cover layer design using geometric patterns, colors, and emoticons.](image2)

Figure 4.5: Customized cover layer design using geometric patterns, colors, and emoticons.

Fig. 4.4 shows six distinctive geometric images used for the block surfaces. These images were selected to represent 1-, 2-, and 4-fold rotational symmetry where each pair has the same shape with reversed colors [13]. In keeping with the core module, cover layers can be customized for a different purpose. Fig. 4.5 shows SIG-Blocks with different cover images, including geometric patterns, distinctive colors, and simplified facial expressions (emoticons). Users can also change the cover layers with basic mathematical operations, numbers, or letters. Although printed paper images were
initially used for the cover layers, the blocks may be covered with digital display
devices, such as LED matrices or LCD screens, while the overall cost and durability
may be of a concern.

4.3.2 iSIG-Blocks & Platform

Fig. 4.6: interactive sensor-integrated geometric blocks, called iSIG-Blocks

Figure 4.6: interactive sensor-integrated geometric blocks, called iSIG-Blocks

Fig. 4.7 shows the CAD drawings of iSIG-Blocks and the embedded electronic
components. Each block is equipped with sensors for detecting assembly and ma-
nipulative motions, a wireless communication module for data collection and repro-
gramming, and sensory feedback mechanisms. The embedded sensors include three
gyroscopic sensors, a tri-axial accelerometer, and electromechanical contact sensors.
Sensory feedback to the user is enabled by reprogrammable LED patterns on the
block surfaces (visual), a piezo buzzer (auditory), and a vibration motor (tactile).
Each block is 2.75 inches in length along each side and weighs about half a pound.
An 8-bit microcontroller (ATmega1280) controls and executes the entire application
program and handles sensor data. The block is also equipped with 128 KB ISP flash
memory card, 4 KB EEPROM, 8 KB SRAM, and 86 general-purpose I/O lines. The
I/O lines are used to collect inputs from the sensors and execute sensory feedback
functions. The block is 2.75 inches in length along each side and weighs about 200 gram.

Each cube is built from seven printed circuit boards (PCBs), six forming a cube and one assembled at the center. Each of three circuit boards are soldered together forming a half cube and share or distribute the electrical signals. Two pre-soldered pieces are assembled to the main circuit board in the middle of the cube through a slide-in connector. The outer surface is then covered by black acrylic sheets, each with eight right-angled triangular shapes for defining geometric patterns. Finally, it is covered by clear acrylic sheets with a teflon film layer for diffusing LED lights and encapsulating all electronic components except for the contact sensors and a USB port for charging batteries.

An 8-bit microcontroller, ATmega1280, controls and executes the entire application programs and handles sensor-collected data. It is primarily responsible for A/D conversion for the triaxial accelerometer and three gyroscopic sensors and signal control for LEDs, a vibration motor, and a piezo buzzer. It is equipped with 128Kbytes ISP flash memory, 4Kbytes EEPROM, 8Kbytes SRAM, and 86 general purpose I/O lines. These 86 general purpose I/O lines are used to collect the sensor inputs, and execute the feedback simulations. Each block contains 96 LEDs (16 LEDs on each surface) controlled by LED sink driver (STP16CP05 from STMicroelectronics) with a low voltage, low current power 16-bit shift register. The block is powered by two 7.4V Lithium-ion batteries which are rechargeable through a USB connector. The operating time depends on how long the LEDs are turned on as they consume most of the power. It runs approximately one hour when all the LEDs are turned on continuously. However, it runs more than 2 hours when only half of the LEDs are turned on at the same time.

Fig. 4.8 shows the tabletop platform that can interact with iSIG-Blocks. This platform functions as a game interface by displaying various geometric patterns and
Figure 4.7: CAD drawings of iSIG-Blocks and embedded electronic components including a wireless module, gyroscopic sensors, triaxial accelerometer, LED driver, spring-loaded connectors, piezo buzzer, and ball magnets.

Figure 4.8: Tabletop platform. A $3 \times 3$ grid structure is used to display patterns with multiple geometric shapes (sub-patterns) for various games which may involve assembly, matrix reasoning and/or working memory.

requiring the player to manipulate the blocks to match, memorize, and/or complete the patterns. Force sensing resisters (FSRs) are installed underneath the top surface to identify location of the blocks and thus the assembly configurations. Rare-earth
magnets in the blocks and the platform pull each other when the blocks are close to the platform and result in the activation of the FSRs. To detect when the blocks are placed on the platform, a pre-calibrated threshold level from the FSRs together with a neighboring detection algorithm is implemented.

**Pattern generating surface design**

The surface of the iSIG-Block was equipped with a reprogrammable LED display consisting of eight right-angled triangles that can be selectively illuminated generating various geometric patterns in one of three distinctive colors as shown in Fig. 4.9. At the initial design stage, both LCD and LED displays were considered. Most widely used LCD and LED displays that are commercially available were not appropriate for our design concept due to fragility for the frequent impact expected during the user manipulation and the difficulty for designing inter-block communication since mechanical or electrical sensing components are blocked by this flat panel display. Pointing this out, a bundle of LEDs is used to make pattern by arranging LEDs efficiently. The aim of geometric design is to create generic patterns that can represent the geometrical relationships such as geometric complexity and symmetry.

![Pattern generating surface design candidates](image)

Several geometric shapes were considered for the surface design and three final candidates are shown in Fig. 4.9. Triangles are the simplest polygon that can generate a large number of polygons, and combinations of triangles can also be used to create quite complex configurations. Among the three, the right design was selected for this study due to the fact that three square shapes in different sizes can be displayed by illuminating two, four, or eight triangles and also due to simplicity in matching with
PCB layout and contact design.

![Diffuse reflection mechanism between PCB surface and Teflon film with comparison of diffusion properties using no diffusive layer, Teflon, and Acetal.](image)

Figure 4.10: Diffuse reflection mechanism between PCB surface and Teflon film (top) and comparison of diffusion properties using no diffusive layer, Teflon, and Acetal (bottom).

The number of LEDs is directly linked to the entire power consumption. To reduce the number of LEDs for the given surface design, a light diffusion film layer was considered to distinctively visualize geometric patterns with a fewer number of LEDs as shown in Fig. 4.10. The contact surface contains eight right-angled triangular shapes and therefore the minimum number of LEDs would be eight. Since each block allows only a small gap between the film layer and the LEDs, the selection of the film material and the number/location of LEDs were critical. For a light diffusion layer, the polytetrafluoroethylene (Teflon) and polyoxymethylene (Acetal) were considered because these materials have a great property on diffusing light from emitting diodes [51], [52]. Between the two, polyoxymethylene was selected as it showed a better diffusion property for the given design constraints as shown in Fig. 4.10. To achieve energy efficient, wider viewing angles, and sufficient brightness, various SMD type LEDs are considered. White, blue, and green LEDs (LITEON) were tested and embedded to the surface. In general, office light intensity is 320-500 lux [53], to make distinctive pattern in the office environment, the desired intensity is targeted more
than 2000 lux. This is also the general brightness of the backlight of a smartphone [54]. 2000 lux with 0.5 mm, the distance between the LEDs and the surface material, is 35 mcd with equation, \( I_v = E_v \times D^2 \). Considering the transmissivity of the Teflon which is approximately 80\%, about 40 - 45 mcd is required for the LED light intensity.

**Contact surface design and magnetic field analysis**

![Connection mechanism](image)

Figure 4.11: Connection mechanism: When two cubes get close together, the rare-earth magnets rotate freely and pull each other.

To enable iSIG-Blocks for a variety of 2D and 3D geometric games, each block was equipped with eight sphere-shaped, rare-earth magnets at the vertices for assisting alignment and securing electrical and mechanical connections among the blocks (Fig. 4.11). An aligned and firm connection between the blocks also guarantees reliable assembly detection and inter-block communication. When two blocks are assembled, spring-loaded pins are triggered. Unlike electromagnets or motor-driven docking mechanisms, permanent magnets do not consume power and are therefore preferable. To allow magnets to freely rotate and pull each other regardless of how the blocks are attached, sphere magnets were housed in small cubicles slightly larger than the magnet’s diameter. The magnet pulls the blocks towards each other with approximately 1.6 lbs of pulling force. However, the acrylic layers between the two magnets are 0.25 inches in thickness, and thus the pulling force is measured as less than 0.25 lbs. That is, no more than 0.25 lbs is required to disassemble the blocks.
Figure 4.12: The contour of magnetic flux density (top) and the contour of magnetic field (bottom) of tSIG-Blocks

For the spherical permanent magnets polarized along its diameter, the flux density, $B$, is calculated by \[55\]

\[ B = \mu_0(H + M) \] (4.1)

where $\mu_0 = 4\pi10^{-7} Tm/A$ is the permeability of free space, $M$ is the magnetization
measured of the net magnetic dipole moment per unit volume, and $H$ is the magnetic field strength. By using the separate variable method to solve the partial differential equation of the Laplace equation, the flux density of the magnetic field is obtained as [55]

$$H = \frac{1}{3} M \left( \frac{a^3}{r^3} \right) \left[ 2\cos(\theta)\hat{r} + \sin(\theta)\hat{\theta} \right]$$ (4.2)

where $a$ is the radius of sphere magnets, $r$ is the distance from the center of sphere magnets, $\theta$ is the angle between the point in the field and bipolar direction, $\hat{r}$ is the unit vector for radial direction, and $\hat{\theta}$ is the unit vector for angular direction. This formula tells us that the magnetic flux density loses its density as the distance gets bigger. In fact, the magnetic flux density at 1cm distance is 125 times stronger than one at a distance of 5cm. Fig. 4.12 shows the magnetic flux density and magnetic field intensity when two iSIG-Blocks are assembled. As shown in the figures, the magnetic field is mostly concentrated at the corners of the blocks and therefore does not affect the inner electronic components. It is also noted that magnets begin pulling each other physically when they are close ($\leq 1.5\text{cm}$).

### 4.4 TAG-Game Design

Three types of TAG-Games, TAG-Game$^A$ (block assembly game), TAG-Game$^S$ (shape matching game), and TAG-Game$^M$ (sequence memory game), are designed aiming for assessing cognitive skills including problem solving, working memory, visual-motor integration, and attention span. Table 4.2 lists cognitive skills that are hypothetically associated with each game. The GUI is developed based on Microsoft Visual C# with an open graphics library (OpenGL) for graphical visualization. TAG-Game involves two GUIs, one for the game player serving as a game display and another for the administrator to administer the game and monitor the player’s performance and behavior in real-time and remotely if applicable. As shown in Fig. 4.13, the admin-
The administrator’s GUI displays the test/game item, current configuration of the blocks, time and accuracy at each manipulative step, communication status of the blocks used in the game, and triaxial accelerations in real time. It also allows the administrator to type comments. While the administrator’s GUI displays measurable data comprehensively, the player’s GUI may simply display the test item and/or the current block configuration depending on the game type. The layout and display elements of the GUI can be easily customized.

Table 4.2: Three types of games and cognitive skills expected to be associated with each game.

<table>
<thead>
<tr>
<th>Type</th>
<th>Associated cognitive skills</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAG-GameA</td>
<td>Fine-motor proficiency, visual-motor integration, low-level working memory, attention span</td>
</tr>
<tr>
<td>TAG-GameS</td>
<td>Fine-motor proficiency, visual-motor integration, low-level working memory, cognitive problem solving, attention span</td>
</tr>
<tr>
<td>TAG-GameM</td>
<td>Fine-motor proficiency, visual-motor integration, high-level working memory, attention span</td>
</tr>
</tbody>
</table>
4.4.1 Three TAG-Games

TAG-Game\textsuperscript{A}: Assembly

![Figure 4.14: 20 TAG-Game\textsuperscript{A} items.](image)

This is an assembly construction game in which the user recreates a displayed image using the SIG-Blocks. The displayed image is an arrangement of geometric images on the SIG-Block faces. The user must rotate and rearrange the blocks in order to create the displayed pattern. The difficulty and novelty of this task comes from the manipulation of the blocks in order to find the correct face images, the size of the pattern, and the discriminability of the block face images within the pattern image. The discriminability of the patterns can be decreased in order to make them more difficult by removing the separating lines between the pattern components. This change results in the big picture being more prominent in the patterns than the individual block face images. Fig. 4.14 shows 20 assembly items.
TAG-Game$^S$: Shape-Matching

As shown in Fig. 4.15, this test consists of 10 items for which an assembly pattern with a missing image is displayed and the participant is prompted to fill in the missing image by placing a SIG-Block with the missing image face up. It is designed to assess fine-motor control, visuospatial reasoning, and problem-solving skills. Each of the patterns has a calculated play complexity value and they are ordered by increasing complexity. For all items, the system records the time it takes for the subject to complete the pattern and whether it is completed correctly. The game requires the participant to figure out the relationship among individual images in each pattern and complete the relationship by supplying the missing piece. As with TAG-Game$^A$, fine-motor control is reflected in the coordination with which the block is manipulated. Visuospatial reasoning is reflected in the participant’s ability to see the relationships between block rotations and face images. Problem-solving skills are reflected in the participant’s ability to find the relations within the pattern and predict the missing image.

TAG-Game$^M$: Memory

This test requires the subject to remember a sequence of images and repeat it back using a SIG-Block. The images within the sequence are flashed one at a time on the
screen and the sequence is played back by placing the SIG-Block with the correct image face up in the order that they appeared on the screen. As shown in Fig. 4.16, two different sets of images are used to make up the sequences; the first is a set of 6 different colored squares and the second is a set of images with the same black and white geometric shapes that were used in the other two games. The two different sets of images are used to see if the complexities of the images within the sequence affect the difficulty of remembering the sequence. This game assesses a participant’s fine motor control, working memory, and attention span. Fine motor control is reflected in the speed and accuracy with which the participant can rotate the block to find the correct face image. Working memory is reflected in how well the participant remembers the sequence of images. Finally, attention span is reflected in the ability to maintain focus when the sequences become longer.

4.4.2 Memorix

Memorix is a TAG-Game created for assessing player’s spatial and working memory span. Geometric patterns are displayed on the platform for a specified duration (e.g., 1 to 5 seconds) and a player is required to memorize the patterns and rebuild them by
assembling iSIG-Blocks. Various types of sensory feedback can be provided indicating correctness at each step, such as generating melodies or vibrations.

Figure 4.17: The administrator screen of Memorix. The GUI displays sensor data, game items, real-time assembly configurations in 3D, correctness notifications, and performance score. The layout and display elements can be customized based on needs.

Figure 4.18: Geometric patterns used in Memorix. Group I patterns (P1-P4) with mono color/shape (top); Group II patterns (P5-P8) with mono color and two distinctive shapes (middle); and Group III patterns (P9-P12) with three distinctive colors and two different shapes (bottom).
The 3 × 3 platform can generate more than $2.8 \times 10^{17}$ distinct patterns. Each pattern consists of multiple geometric shapes arbitrarily placed on 9 possible locations and each of the geometric shapes may be differently colored. For our preliminary evaluation, 12 geometric patterns were designed as shown in Fig. 4.18. Three groups of items each containing four patterns resulting in twelve patterns total. Group I displays uni-color and uni-shape patterns, Group II shows multi-color and uni-shape patterns, and Group III flashes multi-color and multi-shape patterns. Fig. 10 shows the administrator screen of the completed Memorix game. The following elements are displayed for real-time monitoring: 1) user-defined items (12 geometric patterns), 2) real-time 3D animation of the blocks, 3) sensor data (e.g., accelerations, velocities, etc.), 4) data transferred from the platform (e.g., block IDs, assembly detection, time, etc.), and 5) correct/incorrect notifications for each item. For each pattern, play complexity is automatically calculated and shown under each of the pattern images.

The platform displays each item for two seconds and then the player is asked to manipulate and place the iSIG-Blocks to match the pattern as demonstrated in Fig. 4.19.

### 4.4.3 TAPware: Tangible Programming Software using iSIG-Blocks

Following the tangible manipulation protocols, two blocks are randomly picked and shaken for 5 seconds to launch TAPware. Fig. 4.20 shows the initial launchpad screen of TAPware. Once this window opens, one of the blocks is continuously shacked to scroll through the choices until a desired game type is selected. The selection changes every 5 seconds of shaking motion and visualized by darkening the color on the image. Selection can also be done by clicking the button on the GUI. After selecting the game “Memory”, the pattern generator is initiated as shown in Fig. 4.21. The
Figure 4.19: Memorix: The platform displays a pattern consisting of multiple geometric shapes for 2 seconds. When the platform turns off the LEDs, the player recite the pattern by choosing blocks with matching shapes and colors and placing them (facing up) on exact locations.

Figure 4.20: Launchpad screen of TAPware. When two blocks are shaken simultaneously for 5 seconds, TAPware begins and is initialized with the launchpad screen showing the TAG-Game selection table. The pattern being designed is displayed on the left in real time, and the pattern number and play complexity are displayed on the right. Previous patterns are also displayed at the bottom so that the user can review the created items. Pattern design is made by “stamping” the geometric shapes of iSIG-Blocks on the platform. The image on
Figure 4.21: Pattern generator. Any pattern can be created by placing the blocks on the platform. Designed patterns can be reset or saved. Also saved patterns can be loaded. User can play assembly game or memorix game with patterns designed by this pattern generator.

the top surface of the block is imprinted on the platform as demonstrated in Fig. 4.22. Designed patterns are saved by shaking an iSIG-Block for 5 seconds or clicking “Save All Patterns” on the screen. TAPware also has template patterns so that a user can simply load predefined sets of patterns for the game. This function is enabled by clicking “Load Patterns.” “Reset All Patterns” deletes created patterns and allows the user to start over. Once all patterns are designed, shaking two blocks simultaneously turns off the TAPware application. Otherwise, by clicking “Play Assembly Game” or “Play Memorix Game,” the game using the designed patterns can be played.

TAPware provides a unique interface for creating customized sets of geometric patterns for TAG-Games by combining the pre-programmed GUI application with a tangible programming scheme. It also enables the user to create a set of patterns with specific desired difficulty by computing the pattern complexity based on an information-theoretic approach. TAPware does not require learning specific syntax
Figure 4.22: Tangible programming of geometric patterns using TAPware. It demonstrates the design process for a pattern (P12) used in Memorix by “stamping” the geometric shapes of the block on the platform.

of programming languages, but instead allows programming through physical manipulations of the blocks. Although a wide range of games or tests can be made to use the iSIG-Blocks and its associated platform, the current version of TAPware has three initial types of games for which users can create geometric patterns: TAG-Game\textsuperscript{A} for assembly, TAG-Game\textsuperscript{S} for shape-matching, and TAG-Game\textsuperscript{M} for memory. The assembly game simply requires the player to assemble the blocks to match the pattern displayed on the platform. For the shape-matching game, a pattern involves a missing piece and the player is asked identify what block image completes the pattern. The memory game requires the player to remember a geometric pattern or a sequence of patterns within a specific time frame and recite them back by manipulating the blocks.
As illustrated in Fig. 4.23, the overall architecture of TAPware consists of four layers: 1) tangible programming, 2) hardware abstraction, 3) interpretation, and 4) verification. Detailed descriptions for each layer is provided below. Physical manipulation of the blocks involves rotating, shaking, and assembling. Each of these manipulation types can be detected and distinguished by analyzing the sensor data, and therefore, can be used as a specific input command for game programming. Tangible manipulation protocols for designing a TAG-Game are follows:

- **Initialization of TAPware**: Shake any two blocks simultaneously for at least 5 seconds at which point the GUI application will launch TAPware.

- **Selection of a game type**: Shake a block continuously to scroll through the game types and stop shaking when the desired game type among the three (TAG-GameA, TAG-GameS, TAG-GameM) is selected on the GUI. Wait for 5 seconds.

- **Design of geometric patterns**: Design a pattern by placing the blocks with desired geometric shapes on the platform. Once each pattern is completed, shake a block for 5 seconds. Then, the platform displays the pattern image imprinted
from the geometric shapes on the blocks and the GUI also displays the pattern
with a corresponding play complexity value. Repeat this process until the entire
set of patterns are created. To revise the pattern, place a block with a different
geometric shape on the top of previously imprinted one.

• Finalization of TAPware: Shake any two blocks simultaneously for at least 5
seconds at which point TAPware will automatically turn off.

The hardware abstraction layer takes the sensor data received from the iSIG-
Blocks and platform, and transforms it into an interpretable form. The transformed
data includes the ID of the blocks being manipulated, the type of physical manipu-
lations, and the positions and orientations of the blocks placed on the platform. It is
then transmitted to the interpretation layer. During this time, gyroscopic sensors in
stationary blocks are calibrated in order to register a threshold value for distinguish-
ing shaking motion. Interpretation of tangible programming is based on Visual C#
and OpenGL, and hardware programming is done in Java. These programming lan-
guages are selected because they provide a good balance between performance, cross
language compatibility, and rich graphical visualization. According to the tangible
manipulation protocols, manipulation types are distinguished and interpreted into
script language by Visual C#. Once interpretation is completed, visual feedback is
provided through the platform for verification that the pattern is properly imprinted
into the algorithm. If the pattern needs to be modified, the user can redesign the
pattern by placing the blocks again.
4.5 Play Complexity and Scoring Methods

4.5.1 Play Complexity

Difficulty is a relative attribute that can vary significantly depending on each person’s developmental status, age, or health conditions. For an assessment test to produce reliable and sensitive assays, it must be neither too difficult nor too easy. For TAG-Games to be potentially used as a dynamic assessment tool that can be personalized for each individual, a computational measure of play complexity associated with each TAG-Game, $C$, is defined and evaluated (Fig. 4.24).

\begin{align*}
C^A &= H_{\text{initial}} - H_{\text{final}} 
\end{align*}

(4.3)

where $H_{\text{initial}}$ indicates the amount of randomness/uncertainty for the set of geometric blocks used in play and $H_{\text{final}}$ is the remaining randomness/uncertainty after
successfully accomplishing a given task (e.g. assembling the blocks to form a specific assembly configuration) measured by the discrete entropy. Therefore, the difference between $H^{\text{initial}}$ and $H^{\text{final}}$ indicates the amount of uncertainty that is reduced by the person assembling the blocks.

**Play Complexity of TAG-Game$^S$**

The play complexity of TAG-Game$^S$ takes into account the variety of pattern properties that increase and decrease the complexity of the item. The factors that increase the complexity are the total number of blocks, the number of unique blocks, and the length of the embedded pattern. The factors that decrease the complexity, or make it easier to solve, are the number of times the pattern repeats and the number of symmetry axes contained in the pattern. The play complexity of TAG-Game$^S$ is then calculated as

$$C^S = \frac{N \cdot N_d \cdot L}{R \cdot (S + 1)}$$ (4.4)

where $N$ is the total number of blocks, $N_d$ is the number of face images used in the pattern, $L$ is the pattern length, $R$ is the number of pattern repeats, and $S$ is the number of symmetry axes.

**Play Complexity of TAG-Game$^M$**

How difficult it is to remember a sequence of images is related to complexity of the employed images, the number of images and the number of repeating elements within the sequence. The complexity measure is defined by the configurational entropy of the entire sequence computed by

$$C^M = \log_2 Q + \log_2 L$$ (4.5)
where $Q$ is the number of all possible arrangements for the given images and $L$ is the sequence length. For example, if the sequence involves four different colors, the number of possible arrangements of four colors in four distinctive locations is $Q = 4! = 24$ and the corresponding $C^M = \log_2 24 + \log_2 4 = 6.585$. If the sequence item uses the same colored image four times, $Q = 4!/4! = 1$, such that $C^M = \log_2 1 + \log_2 4 = 2$. If the geometric images used in TAG-Game$^A$ and TAG-Game$^S$ are used, it is multiplied by the number of distinctive orientations for each image so that the complexity of the geometric pattern is taken into account.

### Play Complexity of Memorix

Most existing tests have a fixed set of problems that are difficult to modify for different groups based on specific ages or health concerns. In order for TAG-Games to be employed for assessing one’s intellectual, cognitive, and motor capabilities, the game items must be neither too difficult nor too easy. In existing assessment instruments, such as Wechsler Intelligence Scales [56], the difficulty of the test or individual items is analyzed via a large-scale human subject study involving thousands of study participants. Therefore, any modifications in the test requires another round of costly evaluations.

TAPware incorporates a computational measure of play complexity that quantifies difficulty associated with a geometric pattern based on the concept of discrete and configurational entropy [57]. This feature helps users create a TAG-Game with a target difficulty level. Discrete entropy as a measure of play complexity of assembly games was presented in [58, 59]. Extending this approach for all types of TAG-Games, play complexity is represented by complexity of geometric patterns involving multiple shapes and colors, such that, for the $i^{th}$ geometric pattern,

$$C_i = H^L_i + H^C_i + H^S_i$$

(4.6)
where

\[ H_{i}^{L} = \log_{2} B(N, n_{i}^{L}); \quad H_{i}^{S} = \log_{2} O(n_{i}^{L}, n_{i}^{S}); \]

\[ H_{i}^{C} = \log_{2} O(n_{i}^{L}, n_{i}^{C}). \]

\( N \) is the total available locations for placing blocks, \( n_{i}^{L} \) is the number of geometric shapes in the pattern (i.e., the length of the pattern), \( n_{i}^{S} \) is the number of distinctive geometric shapes, and \( n_{i}^{C} \) is the number of distinctive colors used in the \( i^{th} \) pattern.

\( B(n, k) \) is the binomial coefficient which counts the number of ways of choosing \( k \) unordered elements from an \( n \)-element set, computed as [60]

\[ B(n, k) = \frac{n!}{(n-k)!k!}. \]

\( O(n, k) \) denotes the number of “onto” functions, which counts the number of ways to distribute \( n \) labeled objects among \( k \) labeled boxes such that each box contains at least one object, given by [60]

\[ O(n, k) = k! S(n, k) = \sum_{i=0}^{k} (-1)^{i} B(k, i)(k-i)^{n}, \]

where \( S(n, k) \) is the Stirling number of the second kind.

It is important to note that \( C_{i} \) defined here does not take account of complexity of the overall geometry of the pattern. For example, if the pattern contains three geometric shapes along a single column, a row, or a diagonal, memorizing such pattern would be easier than when three shapes are randomly located. Alternatively, complexity associated with locations can be defined by

\[ \tilde{H}^{L} = H^{L} + H^{L_{p}} \quad (4.7) \]

where \( H^{L_{p}} \) captures symmetry properties of the entire pattern while \( H^{L} \) only con-
siders the number of geometric shapes in the pattern.

Fig. 4.25 shows the corresponding play complexities for the designed patterns using Eq. 4.7. Computational steps for three selected patterns are provided below:

- Pattern 1:
  \[ C_1 = H^L_1 + H^S_1 + H^C_1 = 5.17, \]  
  where
  \[ H^L_1 = \log_2 B(9, 2) = 5.17; \]
  \[ H^S_1 = \log_2 1 = 0; \quad H^C_1 = \log_2 1 = 0. \]

- Pattern 6:
  \[ C_6 = H^L_6 + H^C_6 + H^S_6 = 8.98, \]  
  where
  \[ H^L_6 = \log_2 B(9, 3) = 6.39; \]
  \[ H^S_6 = \log_2 O(3, 2) = 2.59; \]
  \[ H^C_6 = \log_2 1 = 0. \]

- Pattern 12:
  \[ C_{12} = H^L_{12} + H^C_{12} + H^S_{12} = 19.12, \]  
  where
  \[ H^L_{12} = \log_2 B(9, 5) = 6.98; \]
  \[ H^S_{12} = \log_2 O(5, 3) = 4.91; \]
  \[ H^C_{12} = \log_2 O(5, 2) = 7.23. \]

In the last pattern (P12), the complexity induced by memorizing the locations of geometric shapes is computed by the entropy on the probability of choosing one.
from all possible ways of picking 5 locations among 9, i.e., $B(9, 5)$. The complexity associated with geometric shapes is computed based on the number of different ways to populate the 5 different locations with 2 distinctive geometric shapes such that each shape is used at least once, which is counted by $O(5, 2)$. Lastly, there are $O(5, 3)$ ways for coloring 5 elements using 3 different colors, such that each color is used at least once.

### 4.6 Technical Evaluation

In order for SIG-Blocks to be implemented as an automated assessment tool in a variety of serious games with either clinical or educational purposes, the blocks should be reliable in collecting the performance and behavior data and communicating either between block and block or between the blocks and host computer. Sensing and communication performance of the blocks is evaluated in terms of durability, proximity sensing failure, self-synchronization delay, and visualization delay.
4.6.1 Durability

In order to test the ability to withstand damage from falling, SIG-Blocks were dropped in free-fall thirty times from desk height (about 3 feet) and 10 times from 7 feet. After being dropped 21 times from desk height, no evident damage was found. When the block was dropped on its 22nd trial, it was found that the corners were dulled by the impact (Fig. 7.19). However, severe damage was not detected until the 30th free-fall drop trial. However, when the block is dropped from 7 feet in height, it began breaking the covers (breaking edges, pulling out the connectors) as shown in Fig. 4.26. Although the covers were broken on impact, the electronics were fine for the 10 times it fell from 7 feet.

4.6.2 Proximity Sensing Failure

Proximity sensing is accomplished by the reflective optical sensor using a binary input. For the test set-up, three blocks are tested for proximity sensing as the three blocks are used for shape matching, sequence memory, face matching, and face memory games which require proximity sensing to detect the orientation of the block towards the L-shaped support (Fig. 4.27). Each port is tested thirty times (six ports per
block), for a total of 540 trials of proximity sensing for the three blocks (30 times x 6 ports x 3 blocks). Over the 540 trials, proximity sensing failure was not detected. Compared to the proximity sensing failures in previous SIG-Blocks, new SIG-Blocks strengthened the intensity pulses of the photodiode to generate high intensity and increased the cutoff voltage of the phototransistor to be robust to the sensitivity of inherent feature of phototransistors.

Figure 4.27: L-shape support for playing games by rotating a single block to match a pattern or play back the remembered pattern.

4.6.3 Peer-to-Peer Synchronization Delay

To evaluate the self-synchronization algorithm, two blocks are programmed to start the internal timer when a port receives photo input and stop the timer when synchronization is completed. By performing 100 trials of the assembly process with two blocks, synchronization failure was not detected. For 100 trials, the self-synchronization time was recorded as 0.36 seconds on average with a standard deviation of 0.31 seconds. Fig. 4.28 shows the histogram of the self-synchronization time. The simulation revealed a 0.3 sec synchronization time with an average 0.15 standard deviation for 100 trials.
4.6.4 Visualization Delay

Although the assembly detection is done in 0.3 seconds on average, total time between physical assembly and visualization takes a little bit longer than 0.3 seconds (about 0.5 seconds). This delay comes from the executing time for visualization with OpenGL and serial reading time from the host computer. The visualization delay and serial reading time are dependent on the processor capabilities of the computer. Our testbed processor is an Intel Core i3-2310M CPU with 2.10GHz, 8GB Ram and a Windows 7 OS. For the monitor, a 1366 x 768 administrator’s display and a 1920 x 1200 player’s display are used with a dual monitor set-up. The serial reading and graphic rendering delays are detected with empirical observation but were not major causes for the total delay and this can be minimized by better processor and memory capabilities of the testbed computer. Also, one possible reason for this delay comes from the wireless data transfer lag in ZigBee (about 30ms) [61]. Taking this delay into account, lag compensation is applied to the final decision of time taken. The lag compensation is calculated as: lag-compensation = wireless data transfer lag + execution time of
functions in C#. Heuristically found lag-compensation is 0.2 seconds and is applied to the algorithm to compensate for delay from the wireless device and executing time in the host computer.

4.7 Human Subject Evaluation

4.7.1 Three TAG-Games

The human-subject study was focused on evaluating 1) preliminary reliability and consistency of the TAG-Games data, 2) the computational complexity measures, and 3) preliminary validity of TAG-Games measures on measuring target cognitive skills. This study was reviewed and approved by Case Western Reserve University’s Institutional Review Board. Fig. 4.29 shows the experimental setup.

![Test setup: A player is given a set of SIG-Blocks and plays TAG-Games while the game items are displayed on a computer screen. The administrator can monitor the player’s performance and sensor data in real-time.](image)

Figure 4.29: Test setup: A player is given a set of SIG-Blocks and plays TAG-Games while the game items are displayed on a computer screen. The administrator can monitor the player’s performance and sensor data in real-time.
Participants

For our preliminary evaluation study, 86 students enrolled at CWRU including 44 females and 42 males, ages between 18 and 30 (Mean = 21.59, SD = 3.13), were recruited and participated in this study. 47 of the students were from the Engineering School and 39 students were from the School of Arts and Science. The demographics are skewed in that the majority of students from the engineering department were male and a larger proportion of the arts and science students were female ($\chi^2 = 12.16, p < 0.01$). For this preliminary testing, gender, major, and age distribution are not controlled.

Protocol

Table 4.3: The test involves six subtests (60 min) including three TAG-Games and three subtests of WAIS-IV.

<table>
<thead>
<tr>
<th>Order</th>
<th>Task</th>
<th>Time</th>
<th>No. Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TAG-Game$^A$</td>
<td>15 min</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>TAG-Game$^S$</td>
<td>5 min</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>TAG-Game$^M$</td>
<td>15 min</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>WAIS: Block Design</td>
<td>5 min</td>
<td>14</td>
</tr>
<tr>
<td>5</td>
<td>WAIS: Matrix Reasoning</td>
<td>10 min</td>
<td>26</td>
</tr>
<tr>
<td>6</td>
<td>WAIS: Digit Span</td>
<td>10 min</td>
<td>48</td>
</tr>
</tbody>
</table>

All three types of TAG-Games and three subtests of the Wechsler Adult Intelligence Scale 4th Edition (WAIS-IV) (i.e., Block Design, Matrix Reasoning, and Digit Span) were performed by the participants (Table 4.3). The three subtests of WAIS-IV were selected for our preliminary validity evaluation because they are expected to assess similar cognitive skills to those tested by TAG-Games. Each participant was given a brief oral instruction at the beginning of each TAG-Game with a sample
item. The three subtests from the WAIS-IV were administered using the standardized protocol (i.e., instruction, prompts, time limits, and discontinuation rules) outlined in the WAIS-IV manual. The three subtests were administered in the order of Block Design → Matrix Reasoning → Digit Span immediately following TAG-Games. Block Design involves 14 items requiring subjects to manipulate a given set of blocks to match the assembly configuration presented in a booklet. It discontinues with two consecutive scores of 0. Matrix Reasoning contains 26 items of colored matrix or visual patterns requiring subjects to guess and select the missing item in the pattern among 5 possible answers. The test discontinues after 3 consecutive scores of 0. Digit Span consists of three parts with 16 items each and 48 items in total - Digit Span Forward (DSF), Digit Span Backward (DSB), and Digit Span Sequencing (DSS). The sets of digits are presented verbally and subjects recite the digits in forward, backward, and increasing order. Subjects have two trials for each length of digit sequence and the test discontinues when two trials both score zeros. The entire assessment was approximately 60 minutes in length.

4.7.2 Scoring Methods

Table 4.4 summarizes scoring methods used for the three TAG-Games. Scoring of TAG-GameA was developed by benchmarking the algorithm employed in the WAIS-IV Block Design subtest, addressing accuracy, item complexity, and speed (i.e., time taken to complete the item). Since TAG-Games are not standardized tests, the participants’ performance were analyzed to determine the score distribution in a similar fashion to WAIS-IV. On the items 1-10, 50% of the correct answers were made at or within 10 seconds resulting in 3 points. The other 50% of the correct answers taking more than 10 seconds yielded 2 points. For the items 11-20, one third of the correct answers were made at or within 20 seconds resulting in 7 points, another third between 20 seconds and 24 seconds resulted in 5 points, and the rest taking more
Table 4.4: Scoring methods for TAG-Game\textsuperscript{A}, TAG-Game\textsuperscript{S}, and TAG-Game\textsuperscript{M} where \( t \) measured in seconds is the completion time.

\begin{center}
\begin{tabular}{l|ccc}
\hline
 & A1-10 & t \leq 10s & t > 10s \\
\hline
TAG-Game\textsuperscript{A} & & 3 & 2 \\
A11-20 & & 7 & 5 \\
\hline
 & S1-5 & t \leq 4s & t > 4s \\
\hline
TAG-Game\textsuperscript{S} & & 3 & 2 \\
S6-10 & & 7 & 5 \\
\hline
 & M1 & M2 & M3 & M4 & M5 & M6 & M7 & M8 \\
\hline
TAG-Game\textsuperscript{M} & 4 & 6 & 8 & 10 & 4 & 6 & 8 & 10 \\
\end{tabular}
\end{center}

than 24 seconds yielded 3 points. For every item, incorrect assembly resulted in zero points. The total available score for TAG-Game\textsuperscript{A} is 100.

Similar to TAG-Game\textsuperscript{A}, scoring for TAG-Game\textsuperscript{S} was designed to take into consideration both accuracy and speed. For the items 1-5, 50\% of correct answers were recorded at or within 4 seconds (SD=0.89) resulting in 3 points. Correct answers made later than 4 seconds resulted in 2 points. For the items 6-10, one third of correct answers were made at or within 8 seconds resulting in 7 points, another third between 8 seconds and 12 seconds resulted in 5 points, and the rest taking more than 24 seconds yielded 3 points. For every item, incorrect shape matching resulted in zero points. The total available score for TAG-Game\textsuperscript{S} is 50.

Scoring for TAG-Game\textsuperscript{M} was based solely on correctness of memory and completion time was not factored into the score. The total possible score of 56 is derived from 2 sets of blocks of 4 possible lengths (i.e., 4, 6, 8, and 10). More specifically, the color and geometric pattern sequences yield scores of 28 each, for a total of 56 points.
Split-half and test-retest reliability:

In an effort to provide preliminary psychometric data on TAG-Games, split-half and test-retest reliability was assessed using Spearman correlations for each of the games. Given that items varied on play complexity ($C$), it was necessary to equate the sets on this variable before conducting the split half reliability. Our first split was based on an odd and even strategy; unfortunately the sets differed significantly in the play complexity ($C$) (i.e., 29.94 vs. 31.83 in TAG-Game$^A$; 22.50 vs. 42.54 in TAG-Game$^S$; and 114.14 vs. 261.74 in TAG-Game$^M$). Given knowledge of play complexity of each set, the items were moved strategically across sets to equate them on complexity. As a result, the TAG-Game$^A$ items were divided into \{1, 3, 5, 8, 9, 12, 14, 16, 17, 19\} and \{2, 4, 6, 7, 10, 11, 13, 15, 18, 20\}, the TAG-Game$^S$ items were divided into \{1, 2, 6, 7, 10\} and \{3, 4, 5, 8, 9\}, the TAG-Game$^M$ items were divided into \{1, 5, 6, 8\} and \{2, 3, 4, 7\}. The split-half reliability coefficients using Spearman's correlation coefficient were $r = 0.73 \ (p < 0.05)$ for TAG-Game$^A$, $r = 0.98 \ (p < 0.1)$ for TAG-Game$^S$, and $r = 1.0 \ (p < 0.1)$ for TAG-Game$^M$. Among 86 participants, 10 of them were administered TAG-Games at two different times approximately two weeks apart. The test-retest reliability was $r = 0.72\,(p < 0.05)$ for TAG-Game$^A$, $r = 0.62 \,(p < 0.05)$ for TAG-Game$^S$, and $r = 0.75 \,(p < 0.05)$ for TAG-Game$^M$.

Validity of complexity measures:

To evaluate the computational complexity measures, the correlation between the participants' performance is investigated with the complexity value for every item. The mean time required completing TAG-Game$^A$ is considered to be an index of performance and as such it was used to examine the impact that $C^A$ had on performance. For TAG-Game$^S$, incorrectness was computed by the normalized number of total incorrect answers for each item. Incorrectness in TAG-Game$^M$ was calculated by the normalized number of total images that the player could not or incorrectly memorized.
in each sequence item. Time to completion in TAG-GameA was highly correlated with 
C^A where r = 0.91 indicating that the more complex the assembly task, the longer the
participants needed to complete the design. Correctness on TAG-GameS was also
highly correlated with C^S where r = 0.95. Regarding TAG-GameM, 8 sequence items
were also examined, with high correlation obtained between the correctness and play
complexity (r = 0.86). For all three TAG-Games, the selected performance measures
were well fitted to a linear function with the corresponding R^2 close to 1.

Correlations between TAG-Games and WAIS-IV subtests:
Among 86, 25 completed both TAG-Games and WAIS-IV subtests. Table 4.5 sum-
marizes the scores from the three TAG-Games and the three WAIS-IV subtests,
with the mean, median, and standard deviation values. For each test, the mean
and median values were close to each other and therefore no transformation was
required. To establish the preliminary validity to the TAG-Games, correlations be-
tween the three TAG-Games and three subtests of WAIS-IV were performed (Table
4.6). As expected, TAG-GameA and the Block Design subtest scores were correlated
(r = 0.48). In addition, TAG-GameS and the Matrix Reasoning subtest were corre-
lated by r = 0.45. Weak correlations were also found between TAG-GameS and Block
Design (r = 0.39), TAG-GameM and Digit Span (r = 0.33), and TAG-GameM and
Block Design (r = 0.28).

Preliminary behavioral analysis
In addition to reliability and validity evaluation based on the participant’s perfor-
mance, preliminary behavior analysis was conducted by estimating the number of
rotational motions applied to the blocks and speed while playing TAG-GameA. The
algorithm for estimating the rotational motions was based on the threshold value and
the local maxima and minima in the acceleration data. The accelerations caused
Table 4.5: Summary of TAG-Game scores: The total number of participants ($N$), total available scores ($P$), mean scores ($M$), and standard deviation ($\sigma$) for each TAG-Game$^A$, TAG-Game$^S$, TAG-Game$^M$, and the three subtests of WAIS-IV (i.e., Block Design, Matrix Reasoning, and Digit Span).

<table>
<thead>
<tr>
<th></th>
<th>TAG-Game$^A$</th>
<th>TAG-Game$^S$</th>
<th>TAG-Game$^M$</th>
<th>Block Design</th>
<th>Matrix Reasoning</th>
<th>Digit Span</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>92</td>
<td>92</td>
<td>86</td>
<td>54</td>
<td>54</td>
<td>31</td>
</tr>
<tr>
<td>$P$</td>
<td>100</td>
<td>100</td>
<td>56</td>
<td>66</td>
<td>26</td>
<td>48</td>
</tr>
<tr>
<td>$M$</td>
<td>73.27</td>
<td>65.58</td>
<td>32.88</td>
<td>52.19</td>
<td>21.02</td>
<td>29.90</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>(14.59)</td>
<td>(12.50)</td>
<td>(5.38)</td>
<td>(10.52)</td>
<td>(2.66)</td>
<td>(4.98)</td>
</tr>
</tbody>
</table>

Table 4.6: Correlations among the three TAG-Games and WAIS-IV subtests (Block Design, Matrix Reasoning, and Digit Span). * Correlation is significant at the 0.05 level (2-tailed); ** Correlation is significant at the 0.01 level (2-tailed).

<table>
<thead>
<tr>
<th></th>
<th>TAG-Game$^A$</th>
<th>TAG-Game$^S$</th>
<th>TAG-Game$^M$</th>
<th>Block Design</th>
<th>Matrix Reasoning</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAG-Game$^S$</td>
<td></td>
<td>0.25**</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TAG-Game$^M$</td>
<td>0.22*</td>
<td>0.05</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Block Design</td>
<td></td>
<td></td>
<td>0.48 **</td>
<td>0.39**</td>
<td>0.28</td>
</tr>
<tr>
<td>Matrix Reasoning</td>
<td>0.03</td>
<td>0.45**</td>
<td>0.09</td>
<td></td>
<td>0.38**</td>
</tr>
<tr>
<td>Digit Span</td>
<td>-0.19</td>
<td>0.07</td>
<td>0.33</td>
<td>0.04</td>
<td>0.03</td>
</tr>
</tbody>
</table>
by the user manipulation were different than what resulted from the impact from block assembly. Therefore, determination of the threshold value by calculating the relationship between the acceleration and orientation was not easy. Instead, sample acceleration data (i.e. training data) was observed thoroughly with the threshold tuning process and applied to the other participant’s data set. For predicting the causal relationship between TAG-Game^A^ score and the number of rotations, a regression method was applied.

Five individuals in each high-score, average-score, and low-score group in TAG-Game^A^ were selected. For each, the speed of manipulation and the total number of rotations applied to the blocks were automatically detected. As shown in Fig. 4.30, while individual differences within the group were observed, a positive correlation was found between the score and the manipulation speed ($r = 0.45, p < 0.05$). On the other hand, as shown in Fig. 4.31, no significant group difference was observed on the total number of rotations applied to the blocks during play.

**Survey Questionnaires and Responses**

To further evaluate the participants reaction to and perception of the TAG-Games, they were surveyed regarding 1) how fun it was to play TAG-Games, 2) if they were willing to play the games again, 3) their favorite game among the three, 4) the participants contact information for those who answered yes in Question 2), and 5) if any, strategies they employed on TAG-Game^M^ (Question 5). Data for the first two questions were evaluated on a 10-point scale (i.e., 10 absolutely, 5 neutral, and 0 not at all). Question 3 was scored categorically, such that the number of participants selecting each game as their favorite was recorded. Data for Question 5 were categorized in one of four ways: no strategy, verbal strategy, visual strategy, or combined visual and verbal strategies. Descriptive data will be provided regarding the sample's strategy profile, with gender and major differences examined. In addition, individual
Figure 4.30: Speed of rotations for three groups. Fast decision-making and movements recorded higher scores in TAG-Game\textsuperscript{A}.

Differences in each TAG-Game performance will be examined as a function of strategy group.

The survey questions were analyzed in two ways: analyses of variance and chi square. In terms of ratings of enjoyableness (Question 1) and willingness to participate again (Question 2), results supported the use of the game. On a scale of 1 (dislike) to 10 (thoroughly enjoyed), a mean rating of 7.93 was endorsed for Question 1. Also a mean rating of 7.94 was given for Question 2. Games differed in their pleasurability (Question 3), with more students enjoying TAG-Games\textsuperscript{A} (49\%) than either TAG-Games\textsuperscript{S} (21.8\%) or TAG-Games\textsuperscript{M} (29.2\%). Participants’ answers for Question 4 were divided into four mnemonic groups depending on whether visual, verbal, both strategies, or no strategies were employed. According to the chi-square analyses, no significant gender and department group differences were observed. In an effort
to explore the relationship between memory strategies and performance, analyses of variance were conducted using the mnemonic groups and the performance scores on the TAG-Games. Although no significant differences were found among these groups, participants who used both visual and verbal strategies tended to perform slightly better in TAG-Games\(^M\) than those who did not.

### 4.7.3 Preliminary Evaluation of Memorix

A small-scale human subject study was conducted involving 12 participants to evaluate Memorix. This preliminary evaluation focuses on the followings: 1) technical feasibility of the developed technology (e.g., reliability of the entire system and accuracy of sensor-collected data); 2) score distribution; 3) validity of the proposed play complexity measure; and 4) the effect of the three pattern parameters (i.e., shapes, colors, and length of the geometric pattern) on the player’s performance.
Design and Method

University students aged between 22 and 34 years (M=27.8; SD=3.4) were recruited and participated in our preliminary evaluation. Among the total 12 participants, 2 were female and 10 were male students. Due to a small sample size, gender or age differences/correlations are not evaluated. Each participant is given a brief introduction of Memorix and how to play the game. Two sample items are provided to the participants to demonstrate the game. The Memorix game is scored based on whether the player memorized each item completely or not within the 30 second time limit. Partial completion is not counted. The total available raw score for 12 items is 98 where scores for individual items are (2, 3, 4, 5, 4, 6, 8, 10, 8, 12, 16, 20). It is then divided by 9.8, such that the maximum available score is 10.

Reliability and accuracy

There were some technical problems raised during the evaluation study. The pulling force between the rare-earth magnets within the blocks and the platform are stronger than desired. This occasionally results in a large impact force on the blocks when they are placed on the platform. This force causes a jump in noise within the blocks, which results in the LEDs on the blocks to blink and the patterns on the block faces to get disrupted. A way to fix this problem once it has occurred is to reset the LED patterns on the block by having the computer resend the patterns to it.

To evaluate accuracy of the automatic assessment, the sensor-collected wirelessly transmitted data were compared with manually recorded data. Reference data from manual recording involves correctness at each block placement and completion time for each pattern. Accuracy of detecting correctness of assembled blocks was 98.2%. Accuracy in time measurement was also 98.2% within an 1.5 second tolerance. These errors were also caused by the high pulling force described above, and therefore, replacing the current magnets with smaller ones would fix most of the technical problems.
identified during this preliminary evaluation.

Figure 4.32: Scores by participants

Figure 4.33: Number of participants who successfully memorized each pattern
Performance scores

The scores from 12 participants are shown in Fig. 4.32 ($M = 6.29$, $SD = 1.61$, max = 7.96, min = 2.65). Fig. 4.33 shows the number of participants who successfully memorized each pattern. All participants successfully memorized patterns with two geometric shapes, P1, P5 and P9. However, none of them was successful in P12.

Play complexity vs. performance

The Spearman’s rank correlation test was conducted on play complexity of Memorix, $C_i$, and performance by the failure rate. A strong correlation was found between the two, such that $\rho = 0.8376$ where $p < 0.01$. As shown in Fig. 4.34, an item with a higher value of $C_i$ tends to have a higher failure rate in memorizing the pattern. Despite the small sample size, this result is an interesting and important finding that leads us to further evaluate the relationship between geometric complexity and individual performance in geometric reasoning, processing, problem solving, and executions.

![Figure 4.34: Failure rate and pattern complexity variations for patterns (P1-P12)](image_url)
Table 4.7: Parameter sensitivity index

<table>
<thead>
<tr>
<th>$a_i$</th>
<th>Parameter</th>
<th>Sensitivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>Length</td>
<td>0.1232</td>
</tr>
<tr>
<td>$a_2$</td>
<td>Shape</td>
<td>0.3164</td>
</tr>
<tr>
<td>$a_3$</td>
<td>Color</td>
<td>0.8333</td>
</tr>
</tbody>
</table>

**Parameter sensitivity analysis**

Parameter sensitivity analysis was conducted to evaluate the effect of three parameters (shapes, colors, and length of the geometric pattern) on the player’s performance. High-level patterns in each parameter are selected and compared to each other. For example, P1 and P4 are compared to evaluate the length effect because both use identical geometric shapes with a single color. P4 and P8 are selected for differentiating the shape effect. Also, P8 and P12 are compared for the color effect. The following formula is used for incremental sensitivity of each parameter [62, 63]:

\[
S_i = \frac{\Delta F}{\Delta a_i}
\]  

(4.11)

where $S_i$ is sensitivity coefficient, $\Delta F$ is finite difference of output and $\Delta a_i$ is scaled small perturbation of input values. $a_1$, $a_2$, and $a_3$ refer to three parameters of length, shape, and color, respectively. Table 4.7 shows that the failure rate is more sensitive to colors than length or geometric shape used in the pattern.

**4.8 Summary of TAG-Games**

Three tangible geometric games (TAG-Games) with sensor integrated geometric blocks (SIG-Blocks) are developed for automated assessment of cognitive and developmental disabilities. The hardware technology realizes wireless and real-time assessment of cognitive skills by employing a set of SIG-Blocks with embedded motion sensors.
To provide a metric for difficulties of the game items, entropy-based play complexity is defined for each TAG-Games. To validate the proposed play complexity as well as technical feasibility of self-synchronization and assembly detection algorithms, technical evaluation and human subject evaluation were gathered from 86 young adults recruited from Case Western Reserve University. Reliability of the TAG-Game items was examined by split-half and test-retest reliability tests for evaluating internal consistency and stability over time. The results showed high correlations in the split-half reliability test as well as the test-retest evaluation. The defined measures of play complexity were found to be highly correlated with the participants’ performance in each game. Also presented was a tangible game technology consisting of a set of iSIG-Blocks with the tabletop platform and TAPware for tangible programming serve as a platform technology that can be used for various research, entertainment, and educational applications. TAPware also features a special embedded function that computes play complexity based on geometric properties of the pattern allowing the user to create a game or test with proper difficulty for target players. Preliminary evaluation of Memorix focused on technical feasibility, validity of computational complexity measure, and sensitivity of pattern parameters. The technical aspects of the TAG-Games include efficient wireless communication and reliable sensing in multi-blocks. To address the problems such as scalable wireless sensing network without data traffic and with robust synchronization, self-synchronization and hybrid wireless networking algorithms are developed and validated in DWSN. Also, clinical aspects of the TAG-Games should not have errors during administration and risks from the system, therefore, the system has to be fault-tolerant and safe to be used in a clinical test. As DWSN can communicate with multiple neighboring agents not just sending data solely to the host, bidirectional communication and sensing with redundant information collection ensured a higher accuracy and fault-tolerance.
Chapter 5

Extended Application I: Swarm Robots

5.1 Overview

Social insects, such as ants, bees, and termites, exhibit collective behavior by sensing local information, communicating with each other, and sharing information [64]. For instance, ants carry food in an optimal path using a pheromone to create a chemical trail followed by other ants [65]. Bees and termites construct complex hives or caves in a decentralized manner without a global leader [64, 66]. Inspired by such biological swarm intelligence, robotics researchers have been developing distributed, cooperative methods for a group of robots with limited functional capabilities to perform tasks that are typically difficult to be achieved by an individual robot. Several benefits are expected by using swarm robots over a single integrated system, including flexibility and accessibility due to the relatively small physical size, robustness achieved by the overall sustainability, and cost-effective control strategies by employing distributed control. For example, a couple of broken robots in a robotic swarm would not affect the overall task performance significantly. Despite these potential advantages, swarm
robotics is also deemed to present several technical challenges in a) forming wireless network using a specific type(s) of sensors; b) establishing reliable and fast communication among the robots; and c) controlling the flock of robots to perform a global task without centralized control. In an attempt to address the above challenges, this study presents the distributed sensing and communication algorithms in homogeneous networked robots with limited sensing, processing, and communication capabilities. Algorithms for global shape formation (i.e., dispersion and forming a line) was also presented to demonstrate global task performance via wireless communication.

5.2 Related Works on Swarm Robots

One of the primary goals of swarm robots application is to achieve collective task execution by communicating and collaborating with each other. To do so, the system must form a communication network among the swarm entities for detecting which robots are nearby and the distance between them, sending and receiving data among those within the communication range, and performing a task collectively. The specific task to be performed may vary depending on the application, but the reliable and efficient communication and localization capability serves as the fundamental functionality of the swarm system as a whole.

A large group of decentralized or distributed robots, called swarm robots, can work together to complete a complex task which is beyond the capabilities of an individual robot. Within this area of robotics, there is a wide range of active research topics that explore algorithms to control these robotic platforms to form a certain pattern. The swarm robotic platforms are designed and implemented with many different functionalities. Swarm-bots use a standard color web cam with a resolution of $640 \times 480$ pixels to detect and recognize each other. It uses a gripper based docking mechanism between the robots and communicates through the WiFi module [67]. In
Figure 5.1: (a) Swarm-bots [67], (b) Kilobots [68], (c) e-Puck [69], (d) Catom [70], (e) Jasmine [71], (f) I-Swarm [72]

an effort to make a tiny and low-cost robotic platform, Kilobot is developed with vibration based locomotion and a simple range only sensor using infrared [68]. Small size robotic platform can be developed for educational purposes, the e-Puck equipped with IR-proximity sensor, accelerometer, microphone, and camera provides sensory input of the robot and communicates with the host computer through Bluetooth technology [69]. Due to various sensors which are accessible in the platform, the e-Puck is used by a robotic research and educational platform [73]. Without using moving parts or mechanisms, an electromagnetic based actuating system was devised by researchers [70], using the electromagnetic coil for locomotion, adhesion, power transfer, communication, and topology sensing. Jasmine is a microrobot with two small DC motors, six IR channels for proximity sensing and communication, and ZigBee communication [71]. Facing the challenges in building micro scale robotic platform, I-Swarm is $3 \times 3 \times 3 \ mm^3$, equipped with a solar cell for energy scavenging, optical communication, vibration contact sensor, and piezoelectric legs for locomotion [72].

Ultrasound-, IR-, or RF-based technologies are most commonly used for wireless communication and localization in swarm robotics. Firstly, ultrasonic-based technol-
ogy requires a pair of ultrasonic transmitter and receiver. For example, the Millibot system used a sonar sensor with an acoustic reflector for reflecting incoming acoustic signals toward the ultrasound transducer [74]. An ultrasonic relative positioning system with high accuracy of about ±4mm and a 3m range was developed for localization in multi-robot systems [75, 76]. The experiments on this positioning system were conducted using only two robots and suffered from echo effects causing poor performance when more than two robots were employed in the test. Instead, communication and localization techniques using IR has several advantages over ultrasonic-based approaches. Although the ultrasonic-based technology is relatively cheap, the accuracy is lower in comparison with IR-based technology. Also, the whole infrastructure of the IR method is simpler than one for ultrasonic [77]. In addition, interference in ultrasonic-based communication is often greater than in IR-based communication [78]. Alice and E-Puck are the robotic hardware platforms that used IR-based technologies for robot-to-robot communication [79, 69]. The Alice microrobot is driven by two wheels for locomotion and is equipped with four IR modules for short-range communication with neighboring robots [79]. E-puck is a hockey puck-like circular robot driven by two stepper motors for locomotion. E-puck also has eight IR modules to achieve short-distance communication [69]. Although IR is still the most commonly used technology for wireless communication in swarm robotics, it has a significant drawback in that IR signals can be easily blocked or interfered with by an object because of the high directivity.

The methods using radio signals attempt to overcome the limitations of the IR-based techniques. Radio signals have wider coverage than IR, while the hardware implementation is still easy and the data-communication link is robust [80, 81, 82]. Localization using the RF data is typically achieved by processing the Received Signal Strength Indicator (RSSI) values, converting this data into physical distances, and performing triangulation or trilateration for localization [83, 84].
performance on finding the radio source, some studies employed additional hardware structures, such as antennae, to increase directivity [85, 15]. These studies showed improved direction sensing achieved by adding a relatively simple hardware structure(s), but neither study was fully validated for a large number of robots. Unfortunately, radio signals can be easily interfered with by obstacles or redirected by another antenna in close proximity. Moreover, fast and precise RSSI acquisition is difficult due to settling time for receiving the data packet. In addition, a small amount of data requires a large amount of data packets to be transmitted to the other receiver reliably. In theory, RSSI packets can be sent in less than a millisecond [86]. However, due to the existing problems such as latency, distance from the access point, and interference, actual transmission time is more than a millisecond. Particularly, collisions occur during bi-directional communication increases transmission time in high-speed modules as there is no mutual time coordination. A relevant study revealed that a transmitter can send only 170 to 180 packets in 30 seconds [87].

NFC (Near Field Communication) and Bluetooth are considered not suitable for swarm robotics due to limitations in network size. The Wi-Fi-based approach may be adopted for a small group of robots with high performance, but it is not desirable for swarm robotics due to the high system complexity and high cost. IrDA and ZigBee are widely accepted in swarm robotics because of the low complexity in hardware, relatively easy system implementation, and low power consumption. Despite the small bandwidths in these protocols, the swarm robots are expected to share only a limited amount of information through the wireless communication channel and therefore these protocols can accommodate the needs. Although the network size of IrDA is only 1, the module size is much smaller while the communication range is much larger than NFC. Therefore it is suitable for one-to-one communication among swarm robots.

In swarm robotics, like many other multi-robotic systems, synchronization in com-
communication is one of the most challenging technical problems as it requires mutual
time coordination among the multiple mobile nodes [88, 89]. It becomes particularly
important for small and low-cost robots as they carry a limited amount of energy [89].
One of the commonly used approaches for synchronization involves a specific reference
node generating a logical clock that can be shared among the multiple robots
via wireless network such as radio frequency (RF) [89]. The reference node may be
predesignated or it can be elected by a minimum number of connection nodes covering
all the connected swarms [88]. As another approach, self-synchronization, or peer-to-
peer synchronization without a master, can be realized using a variable-length source
code [11]. However, a common problem with this approach is that channel errors may
cause synchronization slippage.

A decentralized system is considered consisting of a group of swarm robots where
each robot has limited sensing capabilities and communication range. An IR-based
method was adopted for synchronization via short-distance communication (≤1 inch)
and an RF-based approach was used for localization and long-distance communication
(≤60 inches for localization and 300 feet for communication). Detailed description on
the developed communication and localization algorithms is provided in this section.

5.3 Hardware Development

5.3.1 Mobile Platform with a Corner Reflector

Fig. 5.2 shows the mobile platform where the developed RSS-based localization tech-
niques are implemented. The robot is about 3 inches (in) along each dimension. This
mobile robot is equipped with a differential drive system, an onboard microproces-
sor (ATmega328), a magnetometer, batteries (7.4V lithium polymer), and an XBee
wireless module with ZigBee network technology. The quadrature encoder provides
a resolution of 48 counts per revolution which corresponds to a linear resolution of
slightly under 3mm. The magnetometer (HMC5883L, Honeywell) measures the direction of the magnetic field, providing digital values through the I2C interface, so that the robot can recognize its absolute orientation. In indoor environments, the magnetometer does not provide an accurate orientation as the magnetic field is often distorted. However, it can sufficiently provide low-resolution angles, for example 45° of rotation.

The robot contains two DC gear motors installed for driving the system and enabling the tracked wheels 360° of rotation without translational motion. While a larger reflector seems to achieve a better directivity and detection range, the size of the physical robot shown in Fig. 5.2 is about 3 × 3 × 3 inches which constrains the distance between the two outer edges of the plates to 3 inches in order not to exceed the size of the robot. Therefore, $W = H = 2$ is selected and the dipole is located near the corner where $d = 0.1$ inches, which showed sufficiently high values of RSS and $\rho$ as shown in Fig. 5.3.

![Figure 5.2: CAD drawing (left) and physical prototype of the mobile robot (right).](image)

**Signal Reflector Design**

Among various reflector candidates, a corner reflector is considered due to its simplicity in design and large azimuth response width compared to a flat or parabolic reflector [90]. Assuming that the robots move horizontally on a flat surface propagating the signal in the azimuth angle, a corner reflector is sufficient to receive the
Figure 5.3: A corner reflector with a dipole located in-between two plates

signal along the azimuth direction.

A general design of the corner reflector contains a dipole antenna located between the two infinite-sized flat plates which limits the radiation patterns. The angle between the two plates may vary, but $90^\circ$ is selected to ensure maximum directivity while maintaining the communication range. Although a smaller angle would provide better directivity, it limits the communication range significantly [91]. When the angle between the two plates are fixed at $90^\circ$, the maximum directivity ($\rho_{\text{max}}$) is found at half of the given RF-module’s wavelength [91].

Figure 5.4: Experimental results of RSS and directivity $\rho$ versus $d$, measured at $D = 36$ inches (distance from the target)

It is physically impossible to build an infinite-sized antenna. For multi-robot
applications, it can be further constrained by the physical size of the robot in order
not to interfere the robots’ trajectories. To observe the changes in signal directivity
due to the size of the corner reflector and the location of the dipole, six different sizes
were tested experimentally for varying $d$ (distance from the corner to the dipole):
$(1 \times 1), (2 \times 2), \cdots, (6 \times 6) \text{ [in}^2\text{]}$. Fig. 5.4 shows RSS values and $\rho$ (directivity) versus $d$
for six selected finite-sized reflectors using a 2.4 GHz RF-module with 5 inches
of wavelength. Higher RSS and $\rho$ values were observed for larger reflectors. More
specifically, the maximum radiation intensity was found around $d = 2.5$ inches for the
reflectors larger than $(2 \times 2)$ complying with the infinite-size antenna model described
in [91]. On the other hand, the smaller reflectors showed that $\rho_{\text{max}}$ is near the vertex.
This may be a result of diffracted signals captured by the dipole near the corner.
Moreover, the signals are hardly blocked on the other side of the reflector causing a
lower directivity and RSS for the smaller reflectors.

5.3.2 InchBots

InchBot is a small mobile robot with limited sensing and communication capabilities.
By collaborating with other InchBots as a team or by stacking additional modules
on top of the processing board, it can improve its processing, sensing, power, and
communication capacities. This section describes the detailed architecture of InchBot.

Disadvantages of small-sized robots include limited mobility range, limited energy
availability, and possibly reduced sensing, communication and computation ability.
Our approach to overcoming these disadvantages is an increased modularity through
the stackable board design, novel wheel design, and efficient collaboration among the
robots. The primary components of InchBot include actuators, microcontroller, mo-
tor driver, batteries, sensors, wireless module, and associated circuitry. Four micro
DC motors (6mm in the diameter, typically consumes 120 mA) and two bi-directional
motor drivers (LB1836) as the main actuators. The robot is controlled by an AT-
Figure 5.5: The InchBot is an 1 inch cube, stackable, agile, and omnidirectional micromobile robotic platform. The omnidirectional locomotion is enabled with flexible spoke wheels.

mega328p microcontroller from ATMEL, which is also characterized by a low power consumption (0.2mA at 1MHz). Small coin batteries were initially considered to power the system, but their low maximal discharging current due to high internal resistance did not provide enough current for propelling the robot. Therefore, a rechargeable lithium polymer battery (3.7V, 20C, 50mA) was selected.

The system and its functionality can be extended by simply adding additional modules. 10 × 1 pin headers are adopted for reliable connections between layers to reduce mechanical failure due to repetitive assembly. Two headers share the common pins so that sixteen digital/analog IO are provided including UART and power. Every module shares the same connector for augmented functionality when stacked together. At the bottom of the motor driver, a different wheel chassis can be attached such as a two wheel driving system or a four wheel driving system as shown in Fig. 5.5.
Kinematic Model

A kinematic model of the following omnidirectional robot at \((x, y, \theta)\) shown in Fig. 5.7 is given by

\[
\begin{align*}
    v_y(t) &= dy(t)/dt; \\
    v_x(t) &= dx(t)/dt; \\
    \omega(t) &= d\theta/dt
\end{align*}
\]  

(5.1)

The wheel velocity can be obtained by multiplying the deflection of a flexible spoke wheel, \(\delta_n\), such as \(v_n = \delta_n \cdot \omega_n\) where \(n = 1, \cdots, 4\), assuming that the deformation on the wheels at ground contact due to the load is trivial compared to the centrifugal force of the rotating wheels. The relationship between the wheel velocities and robot

95
velocities is:

\[
\begin{bmatrix}
  v_1(t) \\
  v_2(t) \\
  v_3(t) \\
  v_4(t)
\end{bmatrix} =
\begin{bmatrix}
  -1 & 0 & d \\
  0 & -1 & d \\
  1 & 0 & d \\
  0 & 1 & d
\end{bmatrix}
\cdot
\begin{bmatrix}
  v_x(t) \\
  v_y(t) \\
  \omega(t)
\end{bmatrix}
\]  

(5.2)

From the above equation, the robot speed in relation to the wheel speeds is determined by

\[
v_x(t) = \frac{1}{2}(v_3(t) - v_1(t))
\]

\[
v_y(t) = \frac{1}{2}(v_4(t) - v_2(t))
\]

\[
\omega(t) = \frac{1}{4d}(v_1(t) + v_2(t) + v_3(t) + v_4(t))
\]

(5.3)

**Flexible Spoke Wheel Analysis**

When there is no inertial load applied, the flexible spoke wheel is a cylindrical rubber tube as shown in Fig. 5.8(left). During the locomotion phase, the cylindrical rubber tube is rotated and deformed into a spoke wheel shape by centrifugal force (Fig. 5.8(right)). Due to the nonlinearity of the DC motor between the voltage input and the rotational speed and between the current and the torque in the real world, analyzing a flexible spoke wheel with different shapes/sizes and different number of spokes is challenging without using a well-defined simulation package. To investigate the relationship between the rotational speed and deformation of a flexible spoke wheel, between number of spokes and deformation of flexible spoke wheel, and between the section area where maximum stress is applied and deformation, and further the scaling issues between simulation and actual experimental study, the ANSYS software package with APDL language was used.
Figure 5.8: Flexible spoke wheels when there is no rotational inertial force applied (left) and when there exists a rotational inertial force causing the four spokes on the cylindrical tubes to be fully extended (right).

**Centrifugal force and deformation**

Centrifugal force applied to the flexible spoke wheel is computed by

\[ F = \rho \omega^2 \int_{r_0}^{R} r A(r) \, dx. \]  

(5.4)

where \( \rho \) is the density of the flexible spoke wheel, \( A(r) \) is the area of section where distance to rotor shaft is \( r \), \( \omega \) is the rotor’s rotational speed (rad/sec), \( r_0 \) is the shaft radius, and \( R \) is the radius of flexible spoke wheel. The maximum deflection at the tip, \( \delta \), and the slope at the free end, \( \theta \), can be computed as

\[ \delta = \frac{FR^2}{24EI}; \quad \theta = \frac{PR^2}{8EI} \]  

(5.5)

where \( E \) is modulus of elasticity and \( I \) is the moment of inertia of the section.

**Nonlinear FEA simulation of deformation**

To determine the stress distribution and deflection of the flexible spoke wheel when it is rotated by motors, a model of a cylindrical tube was constructed using FEA for a deformable material, a rigid surface, and inertial loads. The material used for the flexible spoke wheel is polyurethane rubber, an elastomer material with properties of
elasticity. The meshed FEA model of the flexible spoke tube is modeled with SOLID 186 which is a 3-D 20-node structural solid [96]. This material has hyper elasticity and the ability to withstand large strains. In addition, each node has three degrees of freedom. The properties applied to the material model are 0.02GPa for Young’s modulus, 910 kg/m$^3$ for the density, and 0.49 for the Poisson’s ratio [97]. Fig. 5.9 shows the FEA deformation model simulated for 0 rad/sec, 100 rad/sec, and 200 rad/sec.

5.4 Algorithms

5.4.1 Directional RSS-Based Localization

Camera-based localization for both indoor and outdoor applications is one of the most widely accepted methods for mobile robotic systems while its performance can be often sensitive to the camera characteristics (e.g., resolution and frame rate) and external factors (e.g., lighting and shadow) [92, 93]. In general, a high-speed processor is also required to handle on-board image processing. Ultrasound, infrared (IR) radio frequency identification (RFID) and radio-signal-based technologies (e.g.,
time difference of arrival (TDOA) and received signal strength (RSS)) have also been employed for indoor/outdoor localization of mobile robots. These technologies can provide a reliable communication range while requiring a much lower level of data processing than the camera-based approach. Ultrasound-based localization requires multiple pairs of ultrasonic emitters and receivers, as well as an additional RF (radio frequency), or equivalent, system in order to synchronize the receivers. Compared to the camera-based localization, this system can be easily implemented at a relatively low cost [94]. However, it also presents several limitations such as multi-path interferences, which may disturb the distance measurements between the emitter and receiver, and the exponentially increasing complexity when implemented in large scale [95].

IR-based systems are commonly found in commercial applications [108]. While IR cameras are less sensitive to external factors such as lighting or shadow when compared to regular cameras, they still require line-of-sight in order to collect information for localization. In addition, IR proximity sensors usually work within a range of a meter that only allows short-distance localization. RFID tags and readers communicate via magnetic coupling and read the range between the tag and the reader. Passive tags, however, limit the reading range to less than 4 inches in general. Ultra high-frequency (UHF) RFID reader can extend the reading range up to 100 ft, but the modules are typically very expensive (e.g., $500-2000 per module). Regarding the radio signal based localization, some studies reveal that TDOA shows better sensing range and resolution characteristics than RSS [103]. However, it usually uses a sound source for measuring the TDOA and therefore can be more susceptible to environmental barriers than a radio signal. TDOA may use a radio signal source instead of sound while requiring additional hardware implementation to enable high-speed data processing.

RSS-based localization techniques are well-suited for multiple mobile robot appli-
cations due to their simplicity, easy identification of multiple robots, and communication and distance sensing capabilities. There still exist several technical challenges: 1) noisy raw RSS data and inconsistent radiation patterns; 2) nonlinearity between RSS data and physical distance values; 3) low signal transmission power limiting the range of distance measurements; iv) difficulty in simultaneous mobile localization; and v) no information on orientation. In an attempt to address these challenges, several filtration techniques, such as a particle filter [109] or an extended Kalman filter (EKF) [110], have been applied to smoothen the noisy RSS data. Distance-only measurements often require an additional algorithm, such as triangulation, to estimate the position [111]. Range-only SLAM (Simultaneous localization and mapping) can perform position estimation without any such additional algorithm [112]. To obtain the orientation in addition to the distance, the system may utilize additional hardware such as magnetic landmarks for magnetic field based global localization [113]. Range-only SLAM may utilize an odometry sensor, such as a camera, or kinematics of a mobile robot to detect orientation.

In terms of inconsistent radiation patterns, reflectors are widely employed to modify the radiation patterns or increase RSS for low-power radio localization [114, 115, 116]. A recent study investigated the accuracy of RSS-based indoor localization by identifying the channel parameters by applying linear regression [117]. In addition, the Signal-Index-Pair method was proposed to preprocess the data to enhance the precision of the NN (Neural Network) locating model [118]. Graefenstein et al. [114] proposed a system that maps RSS to distance and direction measurements with improved localization performance compared to the direction-only techniques. The system achieved improved directivity of the radiation pattern by adding a simple aluminum metal plate as a reflector antenna. In terms of the antenna reflector design, a parabolic reflector with rotational actuator is employed and experimentally evaluated to estimate direction of arrival (DoA) of radio signal [116]. However, no extensive
research has been conducted in the area of the antenna reflector design for RSS-based localization of networked mobile robots.

This study presents a localization method for multiple mobile robots using filtered directional RSS as illustrated in Fig. 5.10. Antennae are designed to increase directivity of the radiated signal patterns, but have been minimally used for RSS-based localization. In our system, however, a single corner reflector is designed to better estimate the relative orientation of the target. For the collected raw RSS data, robust parameter estimation is conducted using a well-known path loss model [84] for distance measurements. The RANdom Sample Consensus (RANSAC) method is first applied to filter out outliers in the data. Selected inliers are then fitted into a least square model. Simultaneous localization involves two steps: 1) real-time data filtration to identify well-conditioned RSS data for pose (position and orientation) estimation; and 2) path planning between the tracker and the target. The Kalman or particle filter is frequently used for noise cancellation to solve this specific problem [119, 120]. However, these filters require processing time for predicting and correcting the data. To reduce the computational costs, a simple statistical filter is proposed in this study and its processing time is evaluated by comparing with the methods using the Kalman filter and the particle filter.

**Log-distance Path Loss Model**

The RSS measurement quantifies the received power of wireless packets sent via the IEEE 802.15.4 protocol. In the real (free space) case, this value varies inversely with the square of the distance and therefore has been suggested as a means to estimate distances between nodes in mobile sensor networks [103, 104]. In order to map the RSS values to the distance measures, the indoor propagation model is adopted based
Figure 5.10: Localization is realized by 1) improving the signal strength and directivity by adding a corner reflector; 2) estimating the parameters; 3) identifying well-conditioned RSS data; and then 4) controlling the robot to locate the stationary/mobile target.

on the log-distance path loss model given by [84]

\[ L = L_0 + 10\gamma \log_{10} \left( \frac{D}{D_0} \right) + X_g \]  \hspace{1cm} (5.6)

where \( L_0 \) is the pass loss at the distance \( D_0 \) measured in decibel (dB), \( \gamma \) is the path loss exponent, and \( X_g \) is a Gaussian random variable with zero mean and a standard deviation, \( \sigma \). Fig. 5.11 shows the mean RSS values of 10 samples measured at each distance between 2 to 130 inches where the bar length indicates the standard deviation. The standard deviation tends to increase as the distance becomes greater. Also, the RSS measurements decreases monotonically until about 84 inches and starts to slow down afterwards. The RSS data was linearly dependent to the \( \log_{10} \)-distance up to \( 10^{1.7} \sim 10^{1.8} \), corresponding to 50 ∼ 63 inches. Therefore, it is considered that the reliable range of robot-to-robot distance measurements is up to 60 inches, where it follows the \( \log_{10} \)-distance path loss model in Eq. (5.6). The estimated parameters for this range are computed by \( L_0 = -19.96 \) dB, \( \gamma = -2.14 \), and \( d_0 = 2 \) [15].
5.4.2 Control Scheme

The RSS-based cooperative localization technique can be applied for localization of multiple robots. Without using any additional sensor board, such as an infrared
sensor, an InchBot with the basic modules (i.e., processor board, XBee communication board, motor driver board, and power board) can locate and track the goal with the directional derivative based control method. The goal (G) and three robots (R1, R2, R3) communicate with each other and determine the position of the goal by received signal strength from the goal as shown in Fig. 5.12. The goal’s location information (L) is then shared with the tracking robot (R0). At the same time, RSS from R1, R2, and R3 coordinates the tracking robot in local coordinates.

Let $\vec{f}_0 = [x_{01}, x_{02}, x_{03}]^T$ be the estimated distance between R0 and three nearest robots based on RSS. To determine the orientation of G, R0 moves one step, $\Delta x$, forward until the RSS value changes. The change in the RSS value informs the robot which direction it should move in local coordinates to reach the goal. The updated distance estimate between R0 and (R1, R2, R3) becomes $(x'_{01}, x'_{02}, x'_{03})$ and the directional derivative of $\vec{f}_0$ with respect to $x$ is given by

$$D_x \vec{f}_0 = \frac{\Delta \vec{f}_0}{\Delta x} = \begin{bmatrix} \frac{x'_{01} - x_{01}}{\Delta x} \\ \frac{x'_{02} - x_{02}}{\Delta x} \\ \frac{x'_{03} - x_{03}}{\Delta x} \end{bmatrix}$$

(5.7)

If the robot is moving toward the goal, the dot product of $D_x \vec{f}_0$ and $\Delta \vec{d}_0$ will be positive where

$$\Delta \vec{d}_0 = \begin{bmatrix} L_{G1} - x'_{01} \\ L_{G2} - x'_{02} \\ L_{G3} - x'_{03} \end{bmatrix}$$

(5.8)

Otherwise, the robot changes its direction by $\alpha$ computed by

$$\alpha = \arg \max_{\theta \in [0, 2\pi]} (D_x \vec{f}_0 \cdot \Delta \vec{d}_0)$$

(5.9)
Fig. 5.13 illustrates the described control scheme for locating a single goal.

The IR self-synchronization algorithm was evaluated by the synchronization time between two robots. The duration between the times when an IR port of a robot receives an IR signal from another port and when synchronization is completed was recorded. Although outliers were detected when two robots were not well aligned, these outliers were disregarded in this preliminary evaluation as it focuses on testing the algorithm itself by keeping a well-structured condition. With the developed algorithm having reduced packet length, the frequency of RSSI was increased. In very close distance, frequency of updating RSSI recorded 70-80 Hz (Fig. 5.14). However, the frequency decreased with distance. Initially, it was expected that the frequency would be consistent over distance as the radio signal travels with the speed of light, but in reality, the physical distance affected the frequency of RSSI collection.

Figure 5.14: Histogram of the self-synchronization time for 50 trials (Mean = 340 ms; SD = 200 ms). Frequency of measured RSSI by distance.

One of the major advantages of using RSS for distance sensing is that it is already built in most radios and therefore requires no additional sensing hardware. However, recent work has showed that the inherent inaccuracies of using RSS in practical environments makes it almost useless for distance sensing without significant prepro-
cessing or computational resources [104, 105]. RSS data typically involves a large amount of outliers and therefore the data preprocessed through the linear regression can only fall into a bad fit. To address this problem, RANSAC [106] is first applied which is an iterative method to estimate model parameters from a set of observations containing outliers. A subset of measurements is randomly sampled, its average is computed, and finally all the other measurements are tested against this average value. If measurements fit well to the average, they are added to the subset and rejected otherwise. To further remove the outliers, the least square method is applied to the RANSAC filtered data. Fig. 5.15 shows the RSS data processed by the RANSAC and least square regression methods.

![Figure 5.15: Inliers extraction from raw RSS data versus distance (left) and log10-distance (right)](image)

For multiple mobile robot applications, online RSS data processing and path planning using the processed data are essential. While there exist several well-known algorithms, such as the Kalman and particle filters, that are proven to handle noisy measurement data, these filters still require the data training process which may not be desirable for real-time, embedded applications.

To reduce computational costs, a simple, yet effective, online filtering algorithm is developed based on accumulated statistical data. This filter determines well-
conditioned RSS measurements for estimating the target distance. The filtered RSS data is considered well conditioned if the sampled data forms a Gaussian distribution. A previous work showed that RSS data is non-Gaussian [107]. However, by adding the corner reflector, our experiments showed that directional RSS data exhibit a Gaussian distribution in a near field (less than 60 inches). A non-Gaussian distribution is observed when two RF-modules are apart from each other by more than 60 inches, or when there is no corner reflector. The second case is obvious since the radiation pattern is not symmetric.

The ‘GET RSS’ function shown in Algorithm 2 achieves well-conditioned RSS values by comparing them with the mean-median-mode statistic metric. Consecutive RSS values satisfy the normality if the mean, median and mode values are the same. To be comparable with the RSS values given by integers, the mean value is also rounded to a nearest integer. It also eliminates the outliers if the values are not in $\pm 2\sigma$ (95% confidence interval), where $\sigma$ is the standard deviation.
Algorithm 2 Online RSS filter

1: procedure Get RSS(rss)
2: \( \hat{X} = \text{sort}(X); X_{sd} = \text{std}(\hat{X}) \)
3: \( X_{\text{mean}} = \text{round}(\text{mean}(\hat{X})) \)
4: \( X_{\text{median}} = \text{round}(\text{median}(\hat{X})) \)
5: \( X_{\text{mode}} = \text{round}(\text{mode}(\hat{X})) \)
6: if \( X_{\text{mean}} = X_{\text{median}} = X_{\text{mode}} \) then
7: \( RSS = X_{\text{mean}} \)
8: else
9: if \( rss - 2X_{sd} \leq rss \leq rss + 2X_{sd} \) then
10: \( X \leftarrow rss \)
11: end if
12: end if
13: return RSS
14: end procedure

Algorithm 3 estimates the target distance by mapping the RSS values into the euclidean or \( \log_{10} \) distance. If the estimated distance is less than \( D_a \), the shortest distance threshold, the robot stops and waits for the next task. Otherwise, the robot first scans RSS around itself by rotating 360° for orienting the target. As the corner reflector provides a significant RSS change every 45° rotations, initial scanning is conducted at every 45°. Once the robot collects eight measurements, it returns to the orientation where \( RSS_{\text{max}} \) was obtained. The robot then moves forward with the step distance, \( D(RSS) \), defined by

\[
D(RSS) = 2 \cdot \alpha \cdot 10^{-(RSS + 19.96)/21.4} \tag{5.10}
\]

where \( \alpha \) is the ratio of the step distance to the estimated distance. The step size
is a function of RSS, i.e. a longer step distance when the target is far away and a shorter step distance when the target is close. Constant values were determined by replacing $L$ in (5.6) with $D$. A smaller value of $\alpha$ results in a smaller $D(RSS)$. That means, the robot will take a greater number of steps to reach the target. On the other hand, a large $\alpha$ indicates a large $D(RSS)$ implying that the robot will move a rather large distance in a single step, proportional to the exponential of RSS. In any case, the robot will converge to the target as $D(RSS)$ would be fairly small near the agent. To find the optimal value of $\alpha$ (0.2, 0.4, 0.6, and 0.8), 10 simulations were performed for different values of $\alpha$. The results showed that the minimum number of steps were required to reach the target when $\alpha = 0.4$. This simulation does not take account the step time, therefore it may not necessarily reflect the time efficiency. However, experimental results measuring the total time for reaching the target were highly correlated with the computed number of steps ($r=0.948$, $p<0.05$).

### Algorithm 3 Target Tracking

1: if $D(RSS) < D_a$ then Break
2: else
3: for $i = 1$ to 8 do
4: Rotate 45°
5: $RSS[i] \leftarrow$ Get RSS
6: end for
7: $RSS_{max} = \max(RSS)$
8: for $i = 1$ to 8 do
9: Rotate 45°
10: if Get RSS $\geq RSS_{max}$ then
11: Move Forward with $D(RSS)$
12: Break
13: end if
14: end for
15: end if
5.5 Experiments and Simulations

5.5.1 Experiment on Mobile Robots with a Corner Reflector

The developed algorithms were implemented in four mobile robots (Fig. 5.2) and tested for the following experimental scenarios: 1) three robots tracking a single mobile target, and 2) a single robot localizing three stationary targets for mapping. Performance of the proposed online statistical filter was evaluated by comparing the task completion time with the Kalman and particle filters as they are well-known leading filters in wireless sensor networks [121]. The effectiveness of the corner reflector was tested for localizing the target by evaluating moving directions towards the target at each step.

Evaluation of the online statistical filter

![Graph](image)

Figure 5.16: Experimental result on Tracking time [sec] vs. distance [in]

Experiments were conducted to validate the statistical online filtering algorithm by analyzing the tracking time for different distances and comparing the results with the Kalman and particle filters. 10 to 60 inches of distance between the robot to
the target were considered and, for every 10 inches, tracking times were repeatedly measured for 10 times and averaged. As shown in Fig. 5.16, the proposed online statistical filter shows the shortest time to locate the target among the three filters. The particle filter was somewhat better than the Kalman filter, but both of them took significantly more time than the new statistical filter. For the Kalman filter, process noise and observation noise models were tuned at \( w_k \sim N(0, Q_k) \) and \( v_k \sim N(0, R_k) \), where covariance \( Q_k = 0.01 \) and \( R_k = 0.08 \). In general, a particle filter requires a large set of particles, but when the number of particles increases, the system showed latency due to the low-speed processing of the microprocessor. In our test, 16 particles showed successful tracking with minimal particles.

Fig. 5.16 shows trajectories of mobile robot moved from the left to the right repeatedly for 20 times and its orientation histogram toward the agent. The mean orientation error was about \(-4.01^\circ\) while the interim trajectories were also limited by \(< |\pm 45^\circ|\) as anticipated by the properties of the corner reflector.

**Fixed and mobile target localization**

The presented algorithms were embedded in four identical (but with different roles assigned) mobile robots. First, three robots communicate with and track a single mobile target. Second, a single mobile robot locates three stationary targets by visiting one after another. Fig. 5.17 (a) and (b) show experimental snapshots. Among 10 trials in total, it showed a 80% success ratio for the first scenario. Two trials were stopped due to collision among the robots. The second scenario demonstrated sequentially visiting the three targets successfully for all 10 trials. During the experiments, signal obstruction was observed when the mobile robots are lined up toward the target. Although the robot eventually tracked the target, it took additional time to reach the target.
5.5.2 Mobility Experiment on InchBot

Our preliminary experiments focused on testing the utility of flexible spoke wheels for generating forward, diagonal, and turning motions. Detailed description on RSS-based communication techniques that can be implemented in InchBots can be found in [15].

Agility in proposed locomotion

The flexible spoke wheels are not perfectly uniform due to manufacturing errors. Therefore, it requires a calibration process by giving the robot weight factors to each wheel to move the same distance. The InchBot is slippery on the ground but highly agile (faster than 450mm/sec) as shown in Fig. 5.18.

To evaluate the robot’s speed for different duty cycles, 100% duty cycle (3.3V), 75% duty cycle (2.475V), and 50% duty cycle (1.65V) of PWM inputs were tested. Fig. 5.18 shows that the robot moves with increasing speed. Also, high speed motion has more slip when it starts moving whereas low speed motion has less initial slip. The robots were placed on an acrylic plate and recorded with an overhead camera so that distance and time could be retrieved with collected video files.
Forward, Diagonal, and Turning motions

The proposed locomotion allows the consideration of much simpler robot control. Without the limitations of the wheels, the InchBot is capable of omnidirectional locomotion. To demonstrate the locomotion, forward, diagonal, and turning motions were tested by changing the speed of rotation of each flexible spoke wheel. Fig. 5.19 shows a forward motion by rotating two motors. As shown in Fig. 5.18 and Fig. 5.19, robot accelerates due to the slip between the wheel and the ground during high speed motion. Some of the challenges, such as consistent speed control, remains as future work. Different material selection for the wheel could be a possible solution for this. Fig. 5.20 and 5.21 show turning motion by rotating a single motor and diagonal motion by using two motors.
5.5.3 Simulations on Dispersion and Line Formation

To demonstrate collective task execution by swarm robots with limited sensing and communication capabilities, a group of swarm robots are considered, each equipped with IR sensors for self-synchronization and a ZigBee module with about 60 inches of localization range. The swarm robotic system was modeled based on a virtual spring damper model. Algorithms for dispersion without a leader and line formation with an interim leader using only the distance estimation among the neighbors.
Dynamic Model of the Robotic Swarm

Our dynamic model of swarm systems involving multiple agents is based on a virtual spring damper model. To realize attractive and repulsive forces in a stable manner, a virtual damper is added to the spring system to suppress the oscillatory motion of the agents. Modeling dynamic multi-agent systems frequently requires global optimization by designing objective functions. As the number of agents increases, the computational cost for optimization increases exponentially. To address this problem, geometric conditions were used that guarantee global shape formation from geometric primitives based on the attractive and repulsive forces between the agents.

Figure 5.22: Attractive and repulsive forces between the nodes are modeled with a spring and a damper.

A dynamic model of collective agents is considered while maintaining connectivity using attractive and repulsive potential fields based on a mass, spring, and damper system as shown in Fig. 5.22. Each node and edge represent an agent and a communication/sensing/social connection respectively where the spring constant and damping ratio may be differently defined depending on each application. The mass of each node may indicate the weight or relative importance of that agent within the group.

Let $A_1, A_2, \cdots, A_n$ be the agents in two-dimensional space. For the $i^{th}$ agent, the two-dimensional Cartesian coordinate is denoted by $\vec{x}_i = [x_i, y_i]^T$. Likewise, the
vector from the $i^{th}$ agent to the $j^{th}$ agent is given by $\vec{x}_{ij} = [x_i - x_j, y_i - y_j]^T$ where $||\vec{x}_{ij}|| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. Assuming that the $i^{th}$ agent has a mass, $m_i$. Then it follows Newton’s second law: $\vec{f}_m^i = m_i \ddot{x}_i$ where $\vec{f}_m^i$ is the force at the $i^{th}$ node and $x_i$ is the position of the $i^{th}$ node. The force exerted by the spring and damper is described by $\vec{f}_k^i = -k_{ij} \vec{d}_{ij}$ and $\vec{f}_b^i = -b_{ij} \dot{\vec{x}}_{ij}$ where $k_{ij}$ is the spring constant, $b_{ij}$ is the damping coefficient between agent $i$ and neighboring agent $j$, and $d_{ij}$ is a displacement vector between the two. In this modeling, orientation from the agent to another agent is uncertain since each agent measures the distance to other agents only. However, $\vec{d}_{ij}$ is expressed in a vector form since the sum of multiple interactions to an agent will force the agent to a certain direction in the global coordinate system. Therefore, the net force acting on the $i^{th}$ agent in the global coordinate system can be derived by above relationships:

$$\vec{f}_m^i = \vec{f}_k^i + \vec{f}_b^i = -\sum_{j \in M_i} (k_{ij} \vec{d}_{ij} + b_{ij} \dot{\vec{x}}_{ij})$$

where $M_i$ is the number of neighboring agents of the $i^{th}$ agent. The acceleration of the $i^{th}$ agent is calculated as $\ddot{x}_i = -\sum_{j \in M_i} (k_{ij} \vec{d}_{ij} + b_{ij} \dot{\vec{x}}_{ij})/m_i$. In practical control with a microprocessor, the above equation can be rewritten in a discrete time representation given by

$$\hat{x}_i[t+1] - \hat{x}_i[t] = -\sum_{j \in M_i} (k_{ij} \vec{d}_{ij} + b_{ij} \dot{\vec{x}}_{ij}[t])/m_i.$$ 

In case of $m_i = 1$ for all $i = 1, \ldots, n$, the anticipated velocity at $t + 1$ and position at $t + 2$ are given by

$$\hat{x}_i[t+1] = \hat{x}_i[t] - \sum_{j \in M_i} (k_{ij} \vec{d}_{ij} + b_{ij} \dot{\vec{x}}_{ij}[t])$$

$$\bar{x}_i[t+2] = 2\bar{x}_i[t+1] - \bar{x}_i[t] - \sum_{j \in M_i} (k_{ij} \vec{d}_{ij} + b_{ij} \dot{\vec{x}}_{ij}[t]).$$
Dispersion and Paired Line-Formation Algorithms

Two algorithms are introduced here: 1) dispersion based on trigonal planar elements without a leader (Fig. 5.23 (a)) and 2) line formation using paired line elements using an interim leader (Fig. 5.23 (b)). It is assumed that only the distance data measured from the RSSI within the sensing range is available.

Figure 5.23: Elementary geometric shapes: (a) trigonal planar and (b) paired line.

Dispersion using trigonal planar elements:

Let $N$ be a set of dynamic agents. For every $O \in N$, label the closest three neighbor agents as $A, B,$ and $C$. By using the proposed attractive and repulsive forces, let $L_{OA} = L_{OB} = L_{OC} = L_d$ where $L_d$ is the desired distance between the agents.

Forming a line using paired line elements:

Let $N$ be a set of dynamic agents. For every $O \in N$, label the closest agent as $A$. Let $L_{OA} = \epsilon$ and form a pair, $(O, A)$. Find the nearest pair and label as $(B, C)$. Let $L_{OB} = L_{OC} = L_{AB} = L_{AC}$. Note that $L_{OB} = L_{OC}$ for $L_{OA} = L_{BC} = \epsilon$ as $\epsilon \to 0$. When all rectangles are connected to each other, two ends are designated as interim leaders of the connected chains in order to stretch the line to opposite directions.
Simulation Results

MATLAB simulations were performed with 50 agents randomly spread in two-dimensional space. The simple trigonal elements guarantee dispersion of the agents and paired line elements guarantee line shape while maintaining connectivity as shown in Fig. 5.24. If the communication range is less than 60 inches, it is possible to form several smaller discs or lines resulting in outliers. Fig. 5.25 shows two simulation results using two different communication ranges, 40 inches and 60 inches.

5.6 Summary of Swarm Robots

In a belief that limited capabilities of the small mobile robot can achieve a global goal by collecting and collaborating their limited intelligence, the swarm robotic study is started based on DWSN system. In an effort to localize the robots without using overhead camera or central coordinator, sensor-based localization algorithms were investigated in this study. Infrared or ultrasonic-based sensing without communica-
tion between the robots cause limitations such as identification; as robots would not recognize whether a detected object is another robot or other objects such as walls or obstacles. Therefore, distance estimation by communicating simultaneously is required through the use of infrared and radio frequency. Algorithms applied to the TAG-Games such as self-synchronization and hybrid wireless network are devised to be adaptable to this application. To track and follow the robot with limited capability in sensing and communication, a corner reflector was designed and installed on the mobile robot to estimate the distance and orientation to another robot by using signal strength received from the robot. Physical demonstration is done with multiple robotic platforms. While working on the simulation of forming radial dispersion and a line with limited information such as distance-only, the shape forming algorithm was developed based on geometric primitives and simulated it with Matlab. In the meantime, the micro mobile robotic platform called InchBot was developed for further study as the robot has to be small and low-cost to perform physical experiments by deploying hundreds or thousands of robots. Although the physical experiment remains a future study, the locomotion of the robot showed that proposed novel wheel design will allow easy control of the robot by using its omnidirectional feature.
Chapter 6

Extended Application II: Wearable Sensors

6.1 Overview

DWSN technologies are relevant to our daily life as exemplified by the large amount of people who use their smartphone and enjoy various social network service (SNS) such as Facebook, Twitter, and Instagram. Wearable sensors can help people by providing healthcare services such as medical monitoring, memory enhancement, control of home appliances, medical data access, and communication in emergency situations [122, 123]. Real-time and continuous monitoring with wearable sensors connected to the network will increase early or timely detection of emergency conditions and provide a wide range of healthcare services for people with cognitive or physical disabilities. In this context, wearable devices were studied that can perform not only biological and environmental monitoring but also social data monitoring by developing a Socio-biosensor. To validate the prototyped devices, a preliminary experiment was performed with three lab members by designing a path to move inside and outside of campus building and analyzing the recorded data accordingly. The results revealed
that the system can be potentially used for real-time monitoring of healthcare.

6.2 Related Works on Wearable Monitoring Sensors

Determining a person’s location, biological condition, environmental information, and activity are key functionalities in many wearable sensor applications. Most of the current researches in location sensing are based on absolute position measurement such as GPS [124]. However, the absolute location estimation using this reference-based system is limited for indoor use. Indoor navigation systems used in robotic applications often deploy a planar laser scanner, wheel encoders, and a IMU device with a prior generated map. However, a laser scanner-like system is hard to embed in wearable sensors for human use due to the safety issue and the volume and weight of the laser scanner. A foot-mounted sensor has been developed for navigation, the sensor system includes the IMU sensor and three FSRs (force sensitive resistors) and integrates these to provide the navigation solution. The position accuracy was good for low-speed gate but the error increases as the gate speed increases [125].

Recent development of wearable devices have made many choices commercially available. Compared to the commercial product, the research prototypes are relatively too big to intrigue customer’s interest. However, each platform provides various functionality specialized for research domains including stress monitoring and anxiety measurement. A minimally-obtrusive wearable sensor for monitoring mental stress is developed in ambulatory settings (Fig. 6.1 (a) [126]). The system consists of a network of wireless sensors (skin conductance, respiration, heart rate, muscle activity) and a MCU and additional sensors (GPS, acceleration). For multiparameter monitoring purpose, AMON has been developed which employs a miniaturized wristband-type device (Fig. 6.1 (b) [127]). The AMON is capable of monitoring
blood oxygen saturation (SPO2) and skin temperature [127]. Fig. 6.1 (b). Fig. 6.1 (c) is an iCalm which is a compact wearable sensor for long-term measurement of electrodermal activity, temperature, motor activity, and photoplethysmography [128]. Measuring blood pressure by placing two sensors along the artery is challenging due to instabilities in hydrostatic pressure caused by a change in hand position. A modified hydrostatic-based oscillometric method enables significant device miniaturization without the need of a high pressure, actuated cuff [129] as shown in Fig. 6.1 (d). Not only measuring heart rate, Polar also estimates its location by using embedded GPS module [130] as shown in Fig. 6.1 (e). The most well-known wearable wristband is Fitbit Fig. 6.1 (f) [131]. Fitbit logs average heart rate and number of steps during daily life or an exercise period.
6.3 Hardware Development

In an effort to develop real-time environmental, biological, and social monitoring, Socio-biosensors are designed and developed which is wristband-type wearable device. Although many wristband-type devices are commercially available such as Fitbit [131] or iWatch [132], none of them are specifically designed for social monitoring using relative distance measurement. GPS-embedded devices [133] can track their location outdoors but are not applicable for indoor navigation. Socio-biosensor uses received signal strength (RSS) to estimate the distance between users whether they are indoors or outdoors. Socio-biosensor can be used in a variety of domains such as epidemiology and health-care applications. For example, highly infectious diseases such as Ebola virus disease and MERS (Middle East Respiratory Syndrome) need rigorous tracking and monitoring of subjects. Not only the distance monitoring, socio-biosensor will also detect the skin temperature of the subject to alert the disease control center to prevent the spread of the disease.

6.3.1 Socio-biosensor V.1

The first prototype of socio-biosensor is designed and manufactured with 3D printer using two different filaments: an ABS filament and a flexible filament. The design of the device is that there are two small boxes that house the components of the wearable device (Fig. 6.2). These boxes are connected by ribbon cable which allows the boxes to move with the user’s wrist rather than constraining the user’s motion or feeling uncomfortable. Each box has a top and bottom half. The bottom half holds the circuit board with the components and is printed from ABS filament so that it is rigid and protects the circuit. The top half is designed to cover the components from the environment and is printed with Ninjaflex, a brand of thermoplastic polyurethane filament for 3D printing. Ninjaflex is a flexible, high tear resistant material that
was chosen because the flexibility of the material improves the comfortableness of
the device and also creates a tight seal with the bottom half that makes it easy to
assemble but difficult for the top to fall off during use.

Each box measures $1.85 \times 1.85 \times 0.875$ inches with a small slot in the side of the
box that houses the battery that allows for toggling of the power of the sensor with
a pen or a finger. Two boxes were used in this design since a single box with a long
circuit board would be very rigid and was found to be too uncomfortable to wear for
an extended period of time. By using two small boxes, the wearable sensor can twist
with the user’s arm and does not restrict motion. The device is secured using velcro
straps that thread through attachment slots that extend off the side of the bottom
half of the boxes. A small cover for the pulse sensor allows for the velcro strap to
thread through it so that the pulse sensor can be held in place as well.

A separate hub is designed for communication with a host computer and serves
as a node for social data monitoring. The hub measures $2.50 \times 3.55 \times 1.05$ inches
and contains the circuit for communication with the computer. There is a slot on the
top of the hub that allows for FTDI reprogramming and serial communication with
the microcontroller.

Figure 6.2: Socio-biosensor V.1 a) with cover on (left), b) without cover on (right).
6.3.2 Socio-biosensor V.2

The second prototype of socio-biosensor is modified from the first prototype with $3 \times 2 \times 1$ inches. Despite the user testing of V.1 is being tested by users revealing that it functions well, design modifications are made for better ergonomic comfort. The second prototype has been designed with a rubber pad on the bottom providing more comfort to the user while the first version was made with ABS material for the bottom. Fig. 6.3 shows the CAD of the V.2. As the detection of the heart rate is sensitive to the location of the pulse sensor on the artery, the sliding structure of the pulse sensor housing is designed so that the user can adjust the location easily. The rubber pad is manufactured by laser cutting and the outer shell, pulse sensor housing, and the top cover are built with 3D printer. Fig. 6.4 demonstrates when user straps on the sensor.

![Figure 6.3: CAD representation of Socio-biosensor V.2](image)

6.4 Embedded Sensors and Measurable Data

Prototype 1 and 2 have different dimensions but the embedded sensors are the same as following components (Table 6.1):
Processing Unit

For the microcontroller, an ATmega328 chip based Arduino Pro Mini is selected for control and processing of sensor data due to the compact size and low-cost feature as well as an adequate number of digital and analog I/O pins. Also, the open source libraries of each component are provided by the manufacturers or the Arduino user community. In order to program, the FTDI to USB converter has been used as the Arduino Pro does not support USB-based programming. 3.3V is required for the microcontroller and embedded regulator switch down the 3.7V from the Lipo battery.

Biological data acquisition

The biological data includes skin temperature, pulse of the user, and acceleration data to capture the user’s daily motion. Skin temperature is measured with the TMP006 infrared temperature sensor. For the V.1, thermistor-based skin temperature measurement has been used. However, there is a non-negligible latency when measuring the temperature using the thermistor. Therefore, the infrared-based fast measurement method is used for the V.2. Regarding the pulse sensor, an oxygen saturation
### Table 6.1: Components embedded in the socio-biosensors

<table>
<thead>
<tr>
<th>Components</th>
<th>Description</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATmega328</td>
<td>Main processor in Arduino Pro Mini</td>
<td>$2.59</td>
</tr>
<tr>
<td>ADXL345</td>
<td>Tri-axial accelerometer</td>
<td>$1.38</td>
</tr>
<tr>
<td>TMP006</td>
<td>Infrared temperature sensor for ambient temperature and skin temperature</td>
<td>$7.96</td>
</tr>
<tr>
<td>Si1143</td>
<td>Pulse sensor measuring saturated oxygen level</td>
<td>$20</td>
</tr>
<tr>
<td>Microphone</td>
<td>Recording sound and interactions between users</td>
<td>$4.5</td>
</tr>
<tr>
<td>XBee</td>
<td>Zigbee wireless module for communication and RSSI measurement</td>
<td>$20</td>
</tr>
<tr>
<td>RHT03</td>
<td>Capacitive humidity sensor</td>
<td>$7.5</td>
</tr>
<tr>
<td>NeoPixel RGB LED</td>
<td>Tri-color LED to signal that a heartrate has been detected or indicating whether the power is on</td>
<td>$0.5</td>
</tr>
<tr>
<td>Micro-SD reader</td>
<td>Data logger for collected information</td>
<td>$11.96</td>
</tr>
<tr>
<td>Lipo battery</td>
<td>Rechargeable 3.7V lithium polymer battery</td>
<td>$1.2</td>
</tr>
<tr>
<td>Switch</td>
<td>Power switch accessible by user</td>
<td>$0.4</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>$77.99</td>
</tr>
</tbody>
</table>

Sensing module from Modern Device is adopted as it showed better performance when compared to the small pulse sensor from Pulse Sensor Amped. Although the Pulse Sensor Amped is more attractive due to its smaller size, the test result showed more artifact is generated by touch compared to the sensor module from Modern Device. Although the pulse sensor and the skin temperature sensor both uses I2C communication protocol to collect the data from the microcontroller, different addresses allow both sensors to be used simultaneously. To provide pulse feedback in real-time to the wearer of the device, a tri-color NeoPixel LED is embedded in the cover of the box that is farther up the forearm. This LED is initially turned off on startup of the device.
but illuminates when a pulse is detected. When a pulse is detected to be between 60 and 75 bpm the LED glows green. However, if a pulse is detected but is not within this preferred pulse range the LED turns red. Acceleration data is gathered using a tri-axial, low power MEMS accelerometer. Communication with the ATmega328 is achieved over analog inputs. Raw acceleration data is gathered for the X, Y, and Z directions. Using a triaxial accelerometer, tilt angles along the $x$- and $y$-axis can be calculated based on the motion detection algorithm.

**Environmental data acquisition**

Environmental data is collected on the ambient temperature and humidity. Ambient temperature and humidity are measured with an RHT03 sensor. This sensor has small power and voltage requirements but has very high precision. Humidity of the environment is calculated by the property that the capacitance of the sensor changes by absorbing humidity in the air. Relative humidity measurement is accurate $\pm 2\%$ and the temperature measurement is accurate within $\pm 0.5^\circ$ Celsius. Although the RHT03 has an embedded temperature sensor which is a thermistor-like component, more precise ambient temperature is measured by the infrared-based skin temperature sensor as the sensor also can capture the environmental temperature by calculating the circuit-board temperature instantly.

**Social data acquisition**

Quantitative social data monitoring is achieved by determining the distance between users and the amount of time that users spend in close proximity with each other. A RF-based signal strength method is adopted to measure the proximity among users. Using the XBee modules embedded in each device a Received Signal Strength Indication (RSSI) technique is applied to obtain a relative measure of distance between users. Threshold values for close proximity in terms of RSSI values are experimentally
obtained by having three users wear a device and examining the values measured with respect to the physical distance between the users in a variety of settings including inside and outside as well as crowded environments. The preliminary experiment was performed to show whether the RSSI can be utilized for social distance estimation by making travel scenarios on campus.

6.5 Preliminary Experiment

To validate the socio-biosensor, a preliminary experiment has been performed with three subjects. Each subject with socio-biosensor is scheduled to move to designated locations around campus every two minutes. To demonstrate the RSSI (Received Signal Strength Indicator) as a distance/proximity measurement, five different scenarios of realistic social encounters are simulated at each time slot. The wireless communication modules transfer and receive both encounters ID and RSSI to estimate the distance between subjects. Initially, subject D was with subject S in White building while subject C was outside of the Olin building. This social distance is reflected in the signal strength. Likewise, the time slots between 2-4 minutes, 4-6 minutes, 6-8 minutes, and 8-10 minutes also reveal which users were in close proximity and which were not. Assuming there is a transient time between the time slot, no users met during 6-8 minutes as indicated in the weak or no signal strength. The RSS data over time can be seen in Fig. 6.5.

6.6 Summary of Wearable Sensors

In this study, two designs of wearable sensors were prototyped with the goal of measuring real-time person-to-person interactions and biological/environmental data. TAG-Games showed the theoretical benefits of the DWSN, such as robustness in a dynamic environment and scalability in adopting more blocks and sensors, while the wearable
sensors applied the advantages of using the DWSN. Also, the algorithms of TAG-Games and swarm robots such as motion detection algorithms and distance estimation based on the received signal strength are embedded in wearable sensors. The first prototype was developed with two small boxes manufactured from 3D printers and linked with ribbon cables to each other to provide comfort and flexibility. The preliminary experiment to ensure reliable data collection showed the relative slow temperature data collection from the thermistor necessitated the modification of the design. Therefore, the second prototype called Socio-biosensor V.2 is developed with an infrared-based temperature sensor in a single box structure. The main benefits of the socio-biosensors are measurement of social interaction in both an indoor and outdoor environment compared to the other existing wearable sensors. A preliminary experiment has been performed with three lab members. A subject wearing the socio-biosensor is scheduled to navigate a designated path on campus every two minutes.
And the results from the collected data and the scenario were compared to validate the functionality in social interaction monitoring. Currently, the wearable sensor is also used for interaction with a sociable robot called Philos. The user’s behavior such as waving their hands or shaking hands will be analyzed and the data will be sent to Philos for human-robot interaction.
Chapter 7

Extended Application III: Origami Robots

7.1 Overview

Origami has brought novel inspiration to other areas of science, engineering, and education. It has been receiving significant interest in mathematics and art as well as other specialized areas including investigation of DNA folding mechanism, design of medical stents, design of crumple zones in automobiles, and building design [134]. In addition, structural engineers deployed origami folding for both the flexibility and the rigidity the folding patterns provide [135]. An origami-inspired approach is also found in shelter system design [136] and novel crash box design that allows for the absorption of energy during vehicle collisions [137]. Many researchers and educators have also pointed out that origami can have further educational benefits and effects [138, 139, 140, 141, 167]. Origami folding requires dexterous hand manipulation that can improve cognitive skills such as hand-eye coordination [143], fine motor skills [144, 145], and working memory skills [146]. Also, origami is highly associated with attention skills, as it requires listening and watching an instructor or reading
instructions carefully [138, 139]. In this study, three origami robots are presented utilizing the semi-soft structures to achieve a crawling robot, mobile locomotion, and a safe manipulative gripper. Difficulties in hardware implementation such as low-cost and safe design will be discussed as well as network distribution for efficient data traffic management for real-time video streaming.

7.2 Related Works on Origami Robots

The word *Origami* describes a paper-folding technique that creates 3D from 2D paper [147, 148, 149]. Due to the space creation and possibility of miniaturization, origami has been receiving growing attention in the robotics community [150]. In particular, the unique coexisting properties of flexibility and rigidity that the folding patterns provide can bring a breakthrough in robot design. Traditional rigid-body robots have been suited for many situations, but have not fully met the needs for specific situations that may involve unknown or extreme environments where the robot is expected to perform unspecified or a variety of different tasks. Furthermore, paper or thin layers of any foldable material used for building the robot may significantly reduce the weight and the material cost while also being eco-friendly. In the field of robotics, the power of origami structures have been discovered for generating useful functionalities such as actuators [151, 152], springs [153], and printable robotics [150]. More recently, origami design was used for a deformable wheel robot [154] which facilitates quick movement with large wheels and the ability to move through small gaps by folding and reducing the wheel size. Many of the previous robotic designs inspired by origami utilized folding features actuated by Shape Memory Alloys (SMAs). The SMAs are used to extend and contract the structure when a specific amount of heat is applied by a current [155]. Although there are low temperature (70°C) activated SMAs, it can still harm materials that are not thermally resistant, such as paper. In order
to utilize SMAs, the origami bodies could be made with special polymers such as polyester, polyether (PEEK), or polytetrafluoroethylene (PTFE) with high thermal resistance [150].

A clever robotic wheel based on the magic ball pattern is applied to the mobile robot to allow shape deformation of the wheel radius to increase mobility. For example, a wheel with a large radius is better at climbing over things while a wheel with a smaller radius is better at squeezing under things [156]. Robogami [157] is a low profile origami robotic system and crawls by SMA actuation. Robogami is designed and fabricated as flat sheets using a novel manufacturing processes and material combinations with great precision and compactness. A worm-like peristaltic locomotion is demonstrated with origami-inspired robot by employing laser-machined origami patterns to build in a fast and low-cost way [158]. The robot crawls about 50 mm in 3min by folded passive flaps which allow single directional motion. More recently, a centimeter-sized origami robot has been developed which walks, swims, and dissolves. The folded structure contains a permanent magnet and is controlled by external magnetic field underneath the robot. The robot is very lightweight and can
perform a variety of tasks reliably despite its simplicity [159]. By using tile structure and SMA actuator, self-morphing robotic origami is developed [160]. This universal 32-tile sheet demonstrates building a paper airplane and a boat by programming the patterns. Researchers from MIT built the robot from five layers of materials cut by a laser cutter. The sandwiched structure with copper, papers, and a shape-memory polymer allowed folding when heated [161].

7.3 Hardware Development

7.3.1 OrigamiBot-I: Robot that Crawls and Manipulates

OrigamiBot-I is a thread-actuated origami robot to demonstrate a physical application of an origami design for robotic manipulation and locomotion. The selected design can generate twisting and bending motions by pulling, pushing, or torsional force applied to the origami structure. Thread-based actuation also enables various shapes and motions by using different numbers of threads and routing them through different paths. The kinematics for twisting and bending motions based on estimated parameters is derived. To evaluate the potential use of origami for real-world applications and to identify structural weaknesses, preliminary stiffness and durability testing was conducted. For physical demonstrations of robotic manipulation and locomotion, OrigamiBot-I was equipped with four independently-routed threads, where each thread is controlled by a geared DC motor. The robot successfully demonstrated its simple manipulation and locomotion capabilities. In this study, a thread actuated robot called OrigamiBot-I is presented based on an origami design called “twisted tower.” To evaluate its potential utilities and applications, stiffness and durability testing of the origami structure were conducted. In addition, two physical demonstrations of OrigamiBot-I were performed.
Robot Design & Control Scheme

Twisted Tower

![Diagram of Twisted Tower](image)

Figure 7.2: Twisted tower: (a) creases to fold an origami segment, (b) folded segment, (c) assembled tower when extended, (d) assembled tower when fully contracted, and (e) assembled tower when bent.

After analyzing many different origami structures, the twisted tower is selected, which was designed by Mihoko Tachibana [162], for physical demonstration of locomotion and manipulation. The twisted tower is made of identical origami segments that are connected in an octagonal pattern and stacked to form a tower. To begin creating the tower, a single piece of rectangular paper is selected (Fig. 7.2 (a)). The size of the rectangular piece determines the diameter and height of each octagon layer and therefore must be determined based on the size and weight of the actuators, sensors, and associated circuits to be embedded in the robot structure. Each octagon pattern is made up of sixteen individual origami segments that have been folded together as shown in Fig. 7.2 (b). Any number of layers can be added to form a tower with a desired length and stroke. A $4.25 \times 2.125$ inches piece of construction paper is selected where the diameter of the circumscribed circle of the assembled octagon pattern is 3.5 inches. Thickness of the construction paper is 0.012 inches and the thickness of the single origami segment is about 0.119 inches when it is fully compressed. Fig. 7.2 (c) - (e) shows three different configurations of the twisted tower.
Figure 7.3: Three different configurations of the twisted tower by applying (a) linear contraction by twisting two layers in the counter-clock-wise direction; (b) linear contraction by twisting the lower layer in the clock-wise direction and the upper layer in the counter-clock-wise direction; and (c) bending by collapsing one of eight sides of the octagon layer.

Figure 7.4: Examples of three different routings of the threads.

Thread-Actuated Hardware Design

The selected origami structure allows for the generation of screw and bending motions by selectively twisting and squeezing the structure. To achieve these motions, a thread-based actuation system is considered that achieves the desired motions by routing the threads via different paths. Fig. 7.4 shows some examples. Fig. 7.4 (a) shows linear motion achieved by routing two pieces of thread diagonally and the other two linearly, (b) shows torsional motion achieved by routing the thread diagonally, and (c) shows alternative torsional motion by routing the thread linearly. For experiments, four motors are used that control four linearly routed threads independently (Fig. 7.4 (c)). Thread-based actuation is selected for OrigamiBot-I over Shape Memory Alloys (SMAs) which are commonly used for such applications. In order to use the SMAs,
thermal resistant materials are needed for hardware to endure the heat generated by SMAs, which is typically more than 70°C Celsius. Paper is used because it's recyclable, cheap, easily obtainable, and easily handcrafted. Therefore, SMAs were excluded from our consideration due to flammability. Four DC motors generate strokes by winding and releasing the thread, which is better than the small percentage shrinking rate of the SMAs.

Control Circuit Design

Figure 7.5: Memoryless control circuit for OrigamiBot-I.

OrigamiBot-I can generate simple oscillatory motions by using a low-complexity, memoryless control module for forward locomotion without preprogrammed logic in embedded memory or a processor. The circuit only contains low complexity electronic components such as diodes, transistors, resistors, and a timer (Fig.7.5). For oscillatory
motion, 555 timers are used with stable mode generating continuous pulses. One timer controls the top and bottom motor which create forward locomotion, and the other timer sets the frequency of the oscillation of right and left side motors, which allows the robot to bend to the side. The pulse width is 50% for both timer and each time period is modulated manually by potentiometers. To detect a wall or an obstacle while navigating, two photo sensors are installed and control the right and/or left side motor. Two serially connected 3.7V (7.4V) lithium polymer batteries power the robot. To achieve a higher level of manipulability and programmability, an Arduino-based control circuit is implemented.

7.3.2 OrigamiBot-II: Amphibious Robot that Deforms Wheels

Mechanical and Electrical Design

OrigamiBot-II is an autonomous mobile robot with four reconfigurable origami wheels equipped with RF-based wireless communication modules for communication and localization. The robot can move on various types of ground and water surfaces by actively controlling the width and rotating speed of the origami wheels.

Origami Wheels

The origami design used for the wheels is called the twisted tower [162]. The same design has been previously used for demonstrating robotic manipulation and crawling motions [163]. Each wheel consists of two octagonal layers with 2 inches of linear stroke. As shown in Fig. 7.3, the two layers can be twisted in the same or opposite direction while resulting in linear displacement vertically. Among these two options, the second one is selected for OrigamiBot-II as the structure is expected to generate a greater propulsion force. As OrigamiBot-II is intended to be used both on ground and water surface, papers are no longer an option as a folding material for the origami
wheels. Instead, flexible plastic sheets were used. This material is easily foldable while being waterproof. Although two tower layers for the wheels are used, additional layers can be used if a longer stroke is desired. In addition, the diameter of the wheels can be changed by using larger or smaller sheets for folding the origami segments used in the twisted tower.

**Actuation Mechanism**

Figure 7.6: Design of the reconfigurable origami wheel (top) and the overall mechanical design of the robot (bottom)

Figure 7.7: Rotational displacement ($\theta$) of the servo motor result in linear displacement in the wheel changing the width ($d$).
The robot is actuated by four DC motors, one for each wheel, and two servo motors to control the width of the front and rear wheels. Fig. 7.6 shows the design of a single origami wheel with the main rotary actuator (DC motor) and three different views of the assembled robot. These motors are fairly small ($0.94 \times 0.39 \times 0.47$ inch$^3$) while generating 12 oz-in torque. Fig. 7.7 shows the mini-servo motor with a two-bar mechanism that simultaneously controls the width of the front and rear wheels along the same rotating axis. By rotating the links attached to the mini-servo motor, two wheels are fully stretched or contracted. The relationship between the rotational displacement in the servo motor ($\theta$) and the linear displacement on the wheel ($d$) is given by:

$$\theta = \cos^{-1} \left( \frac{L^2 - r^2 - (D_{\text{max}} - d)^2}{2r(D_{\text{max}} - d)} \right). \quad (7.1)$$

where $d = D_{\text{max}} - D$ and $D_{\text{max}} = 48\, \text{mm}$ resulting in $d$ varying between 0 and 26mm. Physical limitation of the servo rotation caused smaller contraction where $\theta$ is approximately 2.79 rad.

**Processing Architecture**

![Schematic of the overall processing blocks](image)

Figure 7.8: Schematic of the overall processing blocks

The overall architecture of the processing module is shown in Fig. 7.8. Two
mini-servo motors are controlled by PWM (Pulse Width Modulation) and four DC motors are controlled by two quadruple high-current half-H driver which provides bidirectional drive current up to 1A at the voltage level of 4.5V - 36V. An Atmega328-based Arduino compatible board is used which provides 14 digital I/O ports and 6 PWM channels among the digital ports. 10 digital ports were used for the DC motor control and 2 PWM channels for two mini-servo motors. When fully stretched, each servo motor consumes less than 0.1A while consuming about 0.2~0.3A when fully contracted. The micro geared DC motor rotates with 120 RPM and 360mA power consumption.

**Waterproof and Water-Resistant Mechanism**

![Image of waterproof mechanism](image)

Figure 7.9: Waterproof mechanism for sealing the DC motor used for wheel actuation.

One of the design challenges is in how to make the actuators, electronic circuits, and power source completely waterproof. The main chassis of the robot was 3D printed and tested for potential water leaking through the layered plastic materials. Crucial parts to be sealed include shafts and electronic circuits. Too large of a pressure on the seal will increase the friction between the seal and the shafts which would consume a large amount of energy from the motor whereas too little pressure would result in water leakage. In OrigamiBot-II, grease is applied around the shaft to be water resistant. To completely seal the electronic circuits including controller and batteries, the electronic components are embedded inside the waterproof housing. The housing was sealed with a rubber panel attached on the top cover, which was fastened by four rare-earth magnets with 2.25lbf pulling force per each magnet. The
holes, where the wires run through from the sealed housing to the motors, were sealed with glue to be waterproof. Fig. 7.9 shows the waterproof mechanism with grease and hot glue. Grease is filled into the housing before inserting the motor and hot glue is applied to resist water around the motor. Regarding the batteries, two lithium polymer batteries (3.7V x 2ea) are serially connected as they can be recharged and assembled in a customized shape while also being equipped with a PCM (Protection Circuit Module). However, the imbalance between the two batteries often caused over charging and inflation in the battery pack. Also, the lithium polymer batteries can be dangerous when leaking occurs. Therefore, a relatively safe 9V-alkaline battery is used for powering the robot.

7.3.3 OrigamiBot-III: Three-Finger Origami Robotic Manipulator

Figure 7.10: OrigamiBot-III: Overall structure with arm and gripper (left) and control board (right)

The robot consists of a manipulator arm with three fingers for locating and grasping objects within a workspace. The main arm and three fingers are folded manually and nylon wires are used to contract and extend the structure by pulling and releasing with low-cost servo motors. The arm length is 10 inch and each finger is 6
inch long when fully extended. The arm can contract up to 6 inches. The robot is
installed in a $16 \times 16 \times 18$ inch frame and controllers and circuits are located on top
of the frame. Maximum power consumption without payload in the gripper is about
10W (fully contracted arm and gripper), and less than 2W when fully extended by
gravitational force. Fig. 7.10 shows the overall hardware of the OrigamiBot-III. The
control board (Fig. 7.10 (right)) consists of two main processing unit: a Raspberry
Pi and an Arduino Uno R3. Captured video streaming is processed in the Raspberry
Pi which has an embedded graphical processing unit while Arduino controls the four
servo motors (arm-control) and three mini servo motors (finger-control).

Table 7.1: Component cost used in OrigamiBot-III

<table>
<thead>
<tr>
<th>Parts</th>
<th>Description</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Papers for Arm</td>
<td>0.012 inch thickness paper</td>
<td>$2</td>
</tr>
<tr>
<td>Plastic sheet for fingers</td>
<td>Polyethylene terephthalate</td>
<td>$8</td>
</tr>
<tr>
<td>Raspberry Pi B+</td>
<td>700MHz SoC with GPU, GPIO, display</td>
<td>$29.9</td>
</tr>
<tr>
<td></td>
<td>interface, and etc</td>
<td></td>
</tr>
<tr>
<td>Camera module</td>
<td>CSI interface 5 megapixel pi camera</td>
<td>$24.9</td>
</tr>
<tr>
<td>SG5010</td>
<td>Servo motors for arm (4ea)</td>
<td>$3.9 x 4</td>
</tr>
<tr>
<td>SG90</td>
<td>Servo motors for gripper (3ea)</td>
<td>$1.2 x 4</td>
</tr>
<tr>
<td>Arduino Uno R3</td>
<td>16MHz microcontroller</td>
<td>$24.95</td>
</tr>
<tr>
<td>Ethernet module</td>
<td>W5100 chipset-based module</td>
<td>$6.9</td>
</tr>
<tr>
<td>Frame for workspace</td>
<td>quater inch plywood and acylic sheets</td>
<td>$10</td>
</tr>
<tr>
<td>Cables</td>
<td>Ethernet cable, USB cable</td>
<td>$2</td>
</tr>
<tr>
<td>Power</td>
<td>2A adapter</td>
<td>$5.9</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>$134.95</td>
</tr>
</tbody>
</table>

**Actuation Mechanism**

For the arm link, four wires from each pulley are connected to the end of the arm
through the structure (Fig. 7.11 (left)). For fingers, on the other hand, one wire
is installed to the origami-structure externally not through the structure due to the property of the plastic film used for the fingers that made them more susceptible to tearing when creating the holes for routing the wires internally. Also, the scale of the fingers compared to the arm is smaller, so it is inefficient to route the holes through the structure which causes a small bending. The potential types of motions resulting from bending due to wire-based actuation are shown in Fig. 7.4. For instance, the twisting motion (Fig. 7.4 (b)) can be applied to the humanoid robot hand to open the door.

**Embedded Actuators, Sensors, and Electronics**

OrigamiBot-III is an internet connected manipulator which allows remote monitoring and controlling. Raspberry Pi B+ is used as the main processing module which adopts a 700 MHz single-core ARM CPU and 250 MHz Broadcom VideoCore IV GPU allowing video processing. A five megapixel high resolution camera is connected to capture each frame during manipulation and causes high data rates using CSI interface to Raspberry Pi. Two types of servo motors are used in the OrigamiBot-III to control the structural shape by pulling and releasing the wire installed over the structure. To control the arm, four SG-5010 motors are installed at the top of the manipulator and pull the nylon wire rolled on pulleys providing 11 kgf-cm. The radius of the pulley is
5cm, so the maximum force to pull the wire is 2.2kgf. For the grippers, three SG-90 motors are installed at each finger and pull the wires with 1.8kgf-cm. The wire is connected 3cm away from the servo axis so that each finger is bent by 0.6kg pulling force. In an attempt to simplify programming and debugging as well as simplicity of sensing and motor control, an Arduino Uno R3 has been used. Rasberry Pi requires an external Analog to Digital Converter (ADC) such as MCP3008 or ADS1115 for analog sensing and requires a logic converter to use 5V processing modules while Arduino Uno R3 collects the analog data and performs the motor control easily by using the preinstalled library and built-in functions. Three Force Sensor Resistors (FSRs) are installed at the end of the grippers to get force feedback and control the motors.

7.4 Kinematics

7.4.1 OrigamiBot-I: Structural Kinematics

The kinematics of the twisted tower provide interesting properties, such as the low bending stiffness and high axial stiffness due to the fact that the paper bends easily but barely stretches. Therefore, the structure generates “rigid-body-like” motions that allow us to use traditional rigid-body kinematics with the addition of physical constraints due to paper thickness and structural limitations. It is assumed that each of top and the bottom plates of an octagon layer are rigid where the possible motions are limited to 1) twisting about the vertical axis and 2) bending towards each of the eight outer edges of the octagon layer, where workspace is dependent upon the number of layers (Fig. 7.3). One of the unique properties of the twisted tower is that the horizontal twisting motion generates vertical displacement. While both clock-wise and counter-clock-wise rotations result in the same linear displacement vertically, the orientation of the top layer is determined by the rotation direction
of each layer. Bending occurs when one of the eight segment areas is compressed. Another interesting property of the twisted tower is that the diameter of the structure does not change while extended or contracted.

![Diagram of a single layer origami tower](image)

Figure 7.12: Single layer origami tower with the assigned body-fixed frame where $h$ is the link length, $\theta$ is the rotation about the z-axis, and $l$ is the displacement between two plates measured along the z-axis.

**Twisting (screw) motion**

Fig. 7.12 shows a single octagon layer. As shown in the figure, rotation about the z-axis generates translation along the same axis. Such motion is called a *screw motion* [164]. The screw motion is generated by twisting the structure by $\theta$ where $-\pi/4 + \epsilon_\theta \leq \theta \leq \pi/4 - \epsilon_\theta$ and $\epsilon_\theta$ is a small angle caused by the thickness of the folded paper when the top and bottom plates are fully collapsed. Full collapse of the top and bottom plate, i.e., $\theta = -\pi/4 + \epsilon_\theta$ or $\theta = \pi/4 - \epsilon_\theta$, results in a small distance between the plates, $\epsilon_h$, where the estimated relationship between $\epsilon_\theta$ and $\epsilon_h$ is $\epsilon_h \approx (4h/\pi)\epsilon_\theta$.

The rigid-body transformation by twisting the top plate by $\theta$ radian is given by

$$
T = \begin{bmatrix}
    e^{i\theta} & \left(h - \frac{4h}{\pi}|\theta|\right)\vec{w} \\
    \vec{0}^T & 1
\end{bmatrix} \in \mathbb{R}^{4 \times 4}.
$$

(7.2)
Note that $\vec{w} = [0, 0, 1]^T$ is the unit vector specifying the axis of rotation, and $\hat{w}$ is the corresponding skew symmetric matrix. Eq. (1) can also be written as

$$T = \begin{bmatrix}
c\theta & -s\theta & 0 & 0 \\
s\theta & c\theta & 0 & 0 \\
0 & 0 & 1 & h - \frac{4h}{\pi}|\theta| \\
0 & 0 & 0 & 1
\end{bmatrix}$$

(7.3)

where ‘$c\theta$’ and ‘$s\theta$’ denote ‘$\cos \theta$’ and ‘$\sin \theta$’ respectively. This shows that $l = h - \frac{4h}{\pi}|\theta|$.

For a tower consisting of $N$ layers, the composite transformation is given by

$$T_0^{N} = \left[ e^{\vec{w}\theta_1, \ldots, N} \left( Nh - \frac{4h}{\pi} \sum_{i=1}^{N} |\theta_i| \right) \vec{w} \right] \in \mathbb{R}^{4\times4}$$

(7.4)

where $\theta_1, \ldots, N = \sum_{i=1}^{N} \theta_i$.

For our physical prototype, $h = 1.35$ [inch] and $\epsilon_\theta = 0.113$ [radian].

Figure 7.13: Schematics of a single layer twisted tower when bending occurs.

Bending motion

A bending motion in a single octagon layer occurs when one of the eight edges on the top plate collapses towards the bottom plate.

As shown in Fig. 7.13, the transformation from the bottom plate to the top plate
in a single octagon layer, where the frames are assigned at the center of each plate where the z-axis pointing upward, is given by

$$T = \begin{bmatrix}
c\theta & -s\theta & 0 & 0 \\
c\phi s\theta & c\phi c\theta & -s\phi & r - (r + l)s\phi \\
-s\phi s\theta & -s\phi c\theta & c\phi & l c\phi + r s\phi \\
0 & 0 & 0 & 1
\end{bmatrix}$$  \hspace{1cm} (7.5)

where \( r \) is the radius of the circumscribed circle of the octagon layer, \( l \) is the displacement between two plates measured along the \( z_0 \)-axis, \( \phi \) is the bending angle, and \( \theta \) is the twisting angle caused by bending. Using that \( \theta = 0 \) when \( \phi = 0 \) and \( |\theta| \approx \theta_{max} \) when \( \phi = \phi_{max} \), the relationship between \( \theta \) and \( \phi \) can be approximated by

$$\theta \approx \frac{\theta_{max}}{\phi_{max}} \phi$$

where \( \phi_{max} \) is the maximum bending angle that a single layer of the twisted tower can generate and \( \theta_{max} \) is the resulting torsional angle. The relationship between \( l \) and \( \phi \) can be also estimated by

$$l \approx h - \frac{(h - \epsilon_h)}{\phi_{max}} \phi.$$

Our physical prototype measures \( r = 1.75 \), \( \theta_{max} = 0.672 \), and \( \phi_{max} = 0.323 \) where \( 0 \leq \phi \leq \phi_{max} \).

**Singularities**

It is assumed that each layer can generate either pure twisting or pure bending motion. For the layer generating twisting motion, the singularity occurs when \( \theta = 0 \) or \( \theta = \pm(\pi/4 - \epsilon_\theta) \) where the structure is fully stretched or fully compressed by twisting.
For bending motion, the singularity also occurs when $\phi = 0$ or $\phi = \phi_{\text{max}}$ where the structure is fully stretched or fully bent.

### 7.4.2 OrigamiBot-II: Experimental Kinematics

Figure 7.14: Kinematic model of OrigamiBot-II: The location of ICR in relation to the linear tread velocity (i.e. $v_l$ and $v_r$).

OrigamiBot-II uses a wheeled skid-steering drive mechanism and our kinematic description follows the method presented in [165]. The robot’s frame is attached at the center of the robot where the Cartesian coordinates of the robot with respect to the global reference frame are given by $\vec{q} = [x, y, \phi]^T$ where $(x, y)$ is the position and $\phi$ is the angular displacement. When the robot turns, the coordinates of the instantaneous center of rotation (ICR) is expressed as $[x_{ICR}, y_{ICR}]^T$. Assuming that the robot is an ideal symmetrical kinematic model, the kinematics of OrigamiBot-II on the plane can be derived as [165]:

$$
\begin{pmatrix}
\dot{y} \\
\dot{\phi}
\end{pmatrix} = \frac{\alpha}{2x_{ICR}} \begin{bmatrix}
x_{ICR} & -x_{ICR} \\
-1 & 1
\end{bmatrix} \begin{pmatrix}
v_l \\
v_r
\end{pmatrix} \tag{7.6}
$$

where $x_{ICR} = -\dot{y}/\dot{\phi}$ and $\alpha$ is the correction factor that take account for mechanical unknown factors on the wheel design. Note that $\dot{x} = 0$ under the ideal symmetrical
kinematic model. A simple experiment was conducted to determine $x_{ICR}$ and $\alpha$ following the protocol presented in [166]. The results found that $x_{ICR} = 0.54$ and $\alpha = 0.97$ for fully extended, $x_{ICR} = 0.60$ and $\alpha = 0.94$ for fully contracted. The results show that wider wheels give better straight line motion with better steering efficiency.

### 7.4.3 Origamibot-III: Workspace Analysis

In this section, the theoretical workspace and manipulability of the OrigamiBot-III is analyzed in order to assess the capabilities and limitations of the system. In a realistic view, pointing forces exerted to the end of the origami structure where the wire tensions are applied as shown in Fig. 7.15. To analyze the displacement and deflection of the structure by these forces, techniques from finite element analysis (FEA) are implemented with structural properties obtained from the stiffness-testing.

![Force model applied to the OrigamiBot-III by wires](image)

**Figure 7.15:** Force model applied to the OrigamiBot-III by wires

Assuming that the stiffness of the structure ($K$) and the second moment of area ($I$) are constant along the length ($L$), the equations of the displacement ($\delta$) and
Deflection ($\nu$) are calculated from the following equations.

$$\delta = \sum_{i=1}^{4} \frac{V_i L}{EA}$$

(7.7)

where $V_i$ is the axial force applied by the wire ($F_i$). The modulus of elasticity, $E$, is calculated by using the stiffness ($K$) obtained from the stiffness testing. $E = \frac{KL}{A}$ where the $K = 8.61lb/in$, $L = 9in$, and $A = 7.65in^2$. The deflection by the bending moment generates the horizontal motion, and the bending moment is obtained from the product of the eccentric force ($F_i$) and the distance from the center ($d = 1in$). Assuming the slope at $z = 0$ is zero in the boundary condition, the deflection by bending moment ($M_i$) is as follow.

$$\nu_i(z) = \frac{1}{EI}(M_i z^2)$$

(7.8)

Fig. 7.16 shows the workspace of the arm ($L = 9in$). The maximum contraction is 4.43 inches, which is 49% of the original length, and the maximum deflection is 3.67 inches. Throughout the workspace analysis, incremental forces are applied to the maximum possible forces by the servo motors in 5% increments.

To evaluate the workspace simulation, the maximum force is applied to see if the manipulator reaches the expected point that the simulation result revealed. Ten trials have been completed and the end points are estimated by scaling the images and the results showing 0.3 inches error vertically and 0.2 inches horizontally. The workspace by the calculation was slightly larger than the physical experiment. This result reveals that the force applied from the motor has dissipated due to friction between the structure and the wire. But the small error can be negligible assuming there are measurement errors in the experiment.
7.5 Communication and Control Strategies

Internet-based teleoperation scheme

This section describes overall architecture used in OrigamiBot-III based on internet connection. The hardware configuration of the overall system is described in Fig. 7.17 including an OrigamiBot-III, a robot server workstation, web server, actuators, sensors, and video camera. In general, computer to computer communication can be implemented by applying a Transmission Control Protocol/Internet Protocol (TCP/IP) socket or the User Datagram Protocol (UDP). The connection between OrigamiBot-III and a remote PC is shown in Fig. 7.17. In the server side, Raspberry Pi is used as the main processor to send collected H264 encoded video using real-time protocol (RTP) over UDP. HTTP-based image streaming has been attempted but
CPU usage was around 70% and the transferred image did not scale as well as H264. Video streaming using real-time streaming protocol (RTSP) has been used over TCP but the streaming video was very laggy. The 5 megapixel video camera captures each frame (15 fps) and sends to the Raspberry Pi using the CSI interface allowing for fast and reliable high-resolution data transfer (1080p). While streaming the video by UDP to the remote PC, Raspberry Pi also communicates with the Arduino using Universal asynchronous receiver/transmitter (UART) to allow motor control and force feedback. Direct control from client (remote PC) is enabled via 802.11 (i.e., Wi-Fi) over TCP. Remote PC accesses to IP address + command packet to control the servo motors and receives the force feedback. A CC3000 Wi-Fi module is attached to the Arduino interfaces between the Arduino and the remote PC. On the Arduino side, node.js based script has been programmed which is open-source, cross-platform runtime environment for server-side networking applications written in JavaScript.

Figure 7.17: Overall system architecture with internet-based control

Fig.7.18 shows graphical user interface written in Visual C# to provide 1) real-time streaming, 2) motion script generation, 3) force limit set-up and feedback, and 4) motion running based on script file.
Figure 7.18: Graphical user interface for internet-based teleoperation

Sensor-based autonomous control scheme

When the robot needs to manipulate autonomously, the first step should necessarily be the object detection. Among many possible options for the gripper to know where to grip such as radar, sonar, laser, or tactile sensing, OrigamiBot-III utilizes a vision camera and force sensor resistor based approach. To address the advantages of the origami-based gripper, an egg shell has been used for our study as it is challenging to grip. To grip safely, the robot has to know the object property so a model-based approach has been addressed. In order to recognize the object shape, geometric and color features are used.

Algorithm 4 describes how the OrigamiBot-III detects the egg shell and navigates the gripper to the egg shell. First, the robot fully contracts the arm to provide a maximum viewing area. When the object detection algorithm starts, it performs Canny’s EDGE detection algorithm embedded in the OpenCV library. Although Canny’s edge detection algorithm is computationally more expensive compared to Sobel, LoG (Laplacian of Gaussian), and Prewitt, it performs better than all other
algorithm 4  Egg shell detection and gripping algorithm using geometry and color feature

1: while true:
2: Motor control: Full contraction of the arm to ensure the maximum view area
3: Object detection algorithm starts
4: Perform Canny’s EDGE detection
5: Convert RGB color to HSV color
6: Find CONTOURS
7: for CONTOUR in CONTOURS
8: N = number of polygonal curves
9: if N > 8 then
10: CONTOUR is elliptical shape
11: if 0.65 < Minmax ratio < 0.8 and HSV in brown color range then
12: Brown egg shell is detected
13: else if 0.65 < Minmax ratio < 0.8 and HSV in white color range then
14: White egg shell is detected
15: end if
16: end if
17: if Egg shell is detected then
18: Find the centroid and maximum equivalent diameter
19: end if
20: Motor control: Reach to the centroid on the ground (z=0) and grip the egg shell
21: end while:

algorithms as they are more sensitive to noise. HSV (Hue, Saturation, Value) is known to have better results for image segmentation than the RGB color space which is capable of emphasizing human visual perception in hues [167]. The algorithm finds all possible contours and calculates the number of polygonal curves using $\epsilon = 0.5\%$ which specifies the approximation accuracy. When $\epsilon$ becomes bigger, distance between the original contour and polygonal curve becomes larger. With $\epsilon = 0.5\%$, elliptical shape is defined when N is more than 8. Within the elliptical shape, Minmax ratio is calculated with the following equation.

$$R_{\text{minmax}} = \frac{R_{\text{min}}}{R_{\text{max}}}$$

where $R_{\text{min}}$ is the minimum distance from the perimeter to a centroid and $R_{\text{max}}$
is the maximum distance from the perimeter to the centroid. The centroid \((C_x, C_y)\) is calculated from the image moment, \(M_{ij}\) where

\[
M_{ij} = \sum_x \sum_y x^i y^j I(x, y)
\]

for \(I(x, y)\) is pixel intensity of the image. Then the centroid \((C_x, C_y)\) is defined as follow.

\[
C_x = \frac{M_{10}}{M_{00}}; C_y = \frac{M_{01}}{M_{00}}
\]

Egg shell’s Minmax ratio varying depending on the brand and the color ranges from 0.65 to 0.8, where the color range in HSV is calculated by pointing to a pixel in the captured images. Once the centroid is calculated, the robot manipulator reaches to the ground to grip the egg.

### 7.6 Experiments

**Stiffness and Durability**

![Image](image.png)

Figure 7.19: Loading and unloading durability testing.

Preliminary experiments are conducted to evaluate the potential of OrigamiBot-I as a robotic manipulator or a mobile robot. Because the main body is made with
paper and the spring-like behavior generated by the origami design is an essential property for achieving the desired motions, stiffness and durability of the structure are first tested. Physical demonstrations of OrigamiBot-I acting as a manipulator arm and a worm-like crawling robot are also presented.

![Performance result of 1000 cycles of loading and unloading origami structure. The color bar on the right side shows the number of cycles of the loading and unloading up to 1000 cycles.](image)

Figure 7.20: Performance result of 1000 cycles of loading and unloading origami structure. The color bar on the right side shows the number of cycles of the loading and unloading up to 1000 cycles.

The origami tower was tested by an Inston servo-hydraulic testing machine (model 8501) in order to estimate the stiffness and force capacity, and dynamic characteristics, such as fatigue over time, (see Fig. 7.19). 1000 sinusoidal cycles were modeled with 1Hz frequency and 1.5 inch amplitude (3 inches stroke). A 250 lb capacity force transducer is used for the test. As shown in the Fig. 7.20, the performance results tell us the origami structure made of paper functions well as a spring until 2 inches stroke out of 6 inches long. The average stiffness for 1-inch contraction to 3-inch contraction is 8.61 lb/inch for loading and 5.81 lb/inch for the first cycle. This hysteresis indicates that the energy is dissipated by friction or heat during loading and unloading.

Another observation is the fatigue of the structure by repetitive motion. Fig.
Figure 7.21: Stiffness changes induced by fatigue.

7.21 shows that the stiffness changes gradually over the time but not significantly indicating there will be limited life-time using the paper based origami robot. Paper is used rather than other materials due to low cost, obtainability, and disposability. However, recyclable polymers such as PETE or high-density polyethylene (HDPE) can be considered for the next version of OrigamiBot.

Manipulator Arm

Figure 7.22: Video captured images of bending motion (top) and loading test (bottom).

An interesting application of OrigamiBot-I is to use it as a manipulator arm (Fig. 7.22). The Arduino-based controller was used to generate manipulating motions. Using two actuators, the manipulator arm can demonstrate linear extension and contraction and swing from the left to the right or vice versa. With additional
actuators, the manipulator arm is capable of reaching specific locations within the workspace. This type of movement can be beneficial in a situation where a flexible arm is needed to move objects from one place to another. By adding more segments to OrigamiBot-I, the manipulator arm could reach a larger area as well as allowing obstacle avoidance using the flexible structure. To test the amount of load that the robot can hold without damaging the origami structure, 1lb, 1.5lb, 2lb, 2.5lb, and 3.0lb of load were attached to the robot’s end-effector (Fig. 7.22 (bottom)). Table I shows the deflection angle ($\alpha$) caused by each weight. Although the origami structure was damaged with 3lb of load, the result shows that the structure is fairly durable for manipulating with weight attached.

Table 7.2: Deflection angle ($\alpha$) per loaded weight.

<table>
<thead>
<tr>
<th>Weight (lb)</th>
<th>1.0</th>
<th>1.5</th>
<th>2.0</th>
<th>2.5</th>
<th>3.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$ (degree)</td>
<td>3</td>
<td>6</td>
<td>8</td>
<td>12</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Worm-like Crawling Robot

Figure 7.23: Forward movement in top view (Oscillation frequency is 0.25Hz).
**Forward locomotion**

In order to make the robot move forward with oscillation, mass-distribution and friction were considered. OrigamiBot-I generates a symmetric flexion and extension motion by winding and releasing the thread via rotaries. Unlike the majority of the worm-like crawling robots which employ peristaltic wave motion for locomotion, a composite of two different friction material-made passive flaps (Fig. 7.24) is used. Conventional flap based locomotion uses an anchoring mechanism [150]. However, the anchoring mechanism of the current design does not guarantee locomotion on a slippery surface such as glass or fine plastic. OrigamiBot-I adopts flexible and directional passive flaps so that locomotion can be achieved by increasing directional friction. One side of the passive flaps has low-friction and the other side with high friction such as rubber. For simplicity, coated tape is attached on one side of the rubber to reduce the friction. Forward locomotion with directional passive flaps was tested as shown in the video-captured images in Fig. 7.23. OrigamiBot-I moves at speed of 0.25 inch/sec. No backing motion is observed since the directional passive flaps prohibit backward movement during locomotion. In this experiment, the robot carries a load that weights about 0.3 lb including batteries, circuits, sensors, and four DC geared motors.

![Figure 7.24: Directional passive flaps.](image-url)
Obstacle navigation by turning motion

For obstacle navigation, a dome-shape cover is added around the control module to easily turn around an arbitrarily shaped obstacle without advanced control (Fig. 7.25). To move forward, the bottom motor winds up and releases the wire to make contract and release the structure with cyclic motion. Turning to the left or right involves a holding period in the top motor while the left or right motor starts oscillating (Fig. 13). This sequential control scheme enables OrigamiBot-I to move to the left or right allowing it to navigate around obstacles in its path. On average, utilizing the tail flicking movement, OrigamiBot-I can turn at $0.5^\circ$ per second to either the right or the left. The total weight of the load that OrigamiBot-I carries in this experiment was about 0.76 lb. The asymmetric weight distribution due to circuits and dome weight was not a significant factor for the performance compared to the directional passive flaps.

Figure 7.25: Obstacle navigation with sequential control scheme.
Locomotion capabilities of OrigamiBot-II

Our experimental study focused on a demonstration of the amphibious locomotion capabilities achieved by OrigamiBot-II. The robot was controlled manually using an RF remote controller. Waterproof testing was also performed. Amphibious locomotion includes moving on the ground and on water. OrigamiBot-II has been tested on the road, grass, rugged terrain, and on the water. The OrigamiBot-II can move forward, backwards, and turn to the right and left. In the general model for friction between surfaces, there are certain inherent assumptions that friction force is independent of area of contact. However, better traction can be obtained with wide tires, or tires with lower pressure. Even though the friction force is independent to the area of contact, the shear stress is dependent on the area so a wider tire can spread a given shear force.
over twice the area so they can transmit a larger accelerating/decelerating/cornering forces.

On the grass and the rugged terrain (Fig. 7.26 (top) and Fig. 7.26 (middle)), the robot could not move when the wheels were fully contracted but it started moving when the wheels were expanded. The robot is also tested to determine if it can behave normally when flipped. Since OrigamiBot-II is designed to be symmetric between top and bottom, the robot could move even when it fell and flipped over (Fig. 7.26 (bottom)). The robot was also tested for water-surface locomotion. As shown in Fig. 7.27, OrigamiBot-II successfully demonstrated its locomotion capability on the surface of water.

Table 7.3: Printing or folding time and cost of each part

<table>
<thead>
<tr>
<th>Parts</th>
<th>Printing or Folding</th>
<th>Time to make</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body frame</td>
<td>Printing</td>
<td>7.5 Hours</td>
<td>$4.8</td>
</tr>
<tr>
<td>Servo mount</td>
<td>Printing</td>
<td>1.2 Hours</td>
<td>$0.6</td>
</tr>
<tr>
<td>Wheel frame</td>
<td>Printing</td>
<td>3.6 Hours</td>
<td>$2.4</td>
</tr>
<tr>
<td>Transmission bar</td>
<td>Printing</td>
<td>0.8 Hours</td>
<td>$0.4</td>
</tr>
<tr>
<td>Wheels</td>
<td>Folding</td>
<td>8 hours</td>
<td>$3</td>
</tr>
<tr>
<td>Geared motors</td>
<td>NA</td>
<td>NA</td>
<td>$12</td>
</tr>
<tr>
<td>Mini Servo</td>
<td>NA</td>
<td>NA</td>
<td>$3.6</td>
</tr>
<tr>
<td>Wireless module</td>
<td>NA</td>
<td>NA</td>
<td>$20</td>
</tr>
<tr>
<td>Controller</td>
<td>NA</td>
<td>NA</td>
<td>$7</td>
</tr>
<tr>
<td>Motor driver</td>
<td>NA</td>
<td>NA</td>
<td>$2</td>
</tr>
<tr>
<td>Batteries</td>
<td>NA</td>
<td>NA</td>
<td>$8</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>NA</td>
<td>NA</td>
<td>$1</td>
</tr>
</tbody>
</table>

Manipulation capabilities

Primary benefits of proposed manipulation using origami-structure is that any irregular shape can be gripped without control effort. The lifting force can be adjusted
by deploying high-torque servo motors. Five different shapes and weight objects are used for the manipulation test.

![Objects](image)

Figure 7.28: Five objects used for manipulation test: plastic bottle, shuttlecock, egg shell, paper cup, and a block

Each object’s weight is as follows: 42g (plastic bottle), 5g (shuttlecock), 6g (egg shell), 8g (paper cup), 82g (block).

Fig. 7.29 demonstrates the manipulation of the OrigamiBot-III for the five objects. Each step includes 1) ready to grip, 2) grip, 3) lift and bend, and 4) ungrip.

**Structural force distribution**

Compared to a rigid structure, the OrigamiBot-III can manipulate fragile objects safely. To evaluate the force distribution while gripping the object, force sensing resistors are used, as well as a power supply to measure the consumed current which is highly related to the torque of the actuators in this system. First, same-scale manipulator gripper is designed with the same actuators. The rigid structure has a free joint at each finger so rubber is used to adjust the backward tension. Also this rubber functions to compensate the structural difference between the rigid structure and origami-structure.

Although the precise force measurement element is a load-cell, force sensing resistors are embedded due to their light-weight on the tip of the finger. The force versus resistance is provided from the manufacturer, calibration is performed in our system. The FSR value versus applied force is shown in Fig. 7.31 and the force distribution
Figure 7.29: Manipulation experiment for five different objects: plastic bottle, shuttlecock, egg shell, paper cup, and a block (from the top)

experiment is inferred by this relationship.

When the motor pulls the wire with an increasing stroke (0 to 18 mm), the current
consumption is indicated by the power supply. A white egg has been adopted for this demonstration. The rigid gripper holds the egg by two points of contact while the OrigamiBot-III grips by bending resulting many points contact. Fig. 7.32 shows that more current is consumed as the stroke increases for the rigid gripper while the current consumption does not linearly increases as much as stroke increases.

The result shows the rigid gripper is susceptible to get more load as the structure is not bent when there are resultant forces by contact. However, the origami-structure is able to deform the structure to distribute the applied force from the load.
Three different origami robots are presented which demonstrate crawling motion, mobile locomotion, and manipulation using the semi-soft properties of origami structures. By adopting origami folding to the robotics, a low-cost, recyclable, and fast manufacturing technique is enabled easily by hand crafting. The proposed twisted tower allowed the robot to change the width of its wheels instantaneously so that...
the robot can successfully move on a road, grass, rugged terrain, and the surface of the water. The last prototype utilized the proposed origami structure to make a safe gripper for fragile object manipulation. Although the origami robots are not multi-agent systems, unlike the previously mentioned applications, the DWSN system is applied to this system to overcome technical problems by distributing hardware and network protocols. To keep the low-cost design criteria and accessible resources, every single component had to be easily obtainable. Therefore, low-cost, opensource hardware platforms are embedded in the system such as the Raspberry Pi and Arduino. Moreover, the highly dense network traffic due to real-time video streaming and remote controlling through the network is distributed by two internet protocols: TCP (Transmission Control Protocol) and UDP (User Datagram Protocol).
Chapter 8

Conclusions

Theoretical framework and algorithms of DWSN systems are addressed in this dissertation and applied primarily to the TAG-Games application, which is an automated cognitive assessment system utilizing sensor-embedded geometric blocks (SIG-Blocks) along with an interactive graphical user interface. A large variety of games can be developed by adopting multiple blocks with wireless sensing capabilities. However, the reliable sensing and efficient wireless networking are technically challenging to address when the system becomes complex and larger. Also, reliable and accurate data transfer without missing any data had to be guaranteed in this system as it was to be tested clinically on human subjects. To resolve these technical challenges, self-synchronization, hybrid wireless networking, motion detection, and assembly detection algorithms were developed. The technical evaluation of the algorithms and hardware implementation revealed that the TAG-Games can be used for assessment purposes as a durable, accurate, and efficient system. Also, the human subject study was performed focusing on evaluation of 1) reliability and consistency of data collected from the TAG-Games, 2) play-complexity measures, 3) validity of TAG-Games on measuring target cognitive skills by adopting a traditional cognitive assessment method, WAIS-III. The statistical results demonstrated that the TAG-Games can be
a fun and automated assessment tool with great potential in rehabilitation for people with mental disability or impaired motor skills.

Due to the long-term human subject study and multi-agent system of the TAG-Games, the theoretical frameworks and algorithms could be applied to three extended applications in parallel: swarm robots, wearable sensors, and origami robots. Through these extended applications, novel hardware platforms could be implemented in small scale and low-cost which are necessarily required for DWSN to deploy a large amount of agents. In addition, new and modified algorithms were applied to each application in order to overcome the technical problems and improve up on the traditional system so as to achieve a goal with multi-agents which have limited sensing, networking, and processing capabilities respectively.

In the swarm robots application, two hardware platforms were developed: mobile robots with a corner reflector and omnidirectional micromobile robots. Preliminary swarm control schemes were validated and the locomotion of the omnidirectional robots were demonstrated. Although the physical experiment with InchBot was not included in this study, geometry-based formation control algorithms were demonstrated to form global shapes with local interactions.

In the wearable sensors application, two socio-biosensors prototypes were created to measure social interactions as well as biological and environmental monitoring. Integration of the received signal strength from the proposed hybrid wireless network algorithm and motion detection algorithm are embedded in this system. To validate the technical evaluation, three lab members were involved in a simulated environment by moving to designated locations every 2-minutes with the wearable devices. The results showed that the socio-biosensors can be utilized to provide multi-modal monitoring in healthcare applications.

In the origami robots application, robots with a novel origami design were developed for achieving amphibious locomotion and articulated manipulation utilizing the
semi-soft properties of origami structures. In this application, the data traffic due to real-time video streaming is controlled by distributing the hardware and internet protocol efficiently. Physical demonstration as well as building the structure itself can be efficient education kits which teach programming as well as develop motor skills.

8.1 Evaluation of DWSN

Despite the great potential of the DWSN, previous works mostly focused on degree of nodes (number of connections at a node) to evaluate how many communication routes are established at each node (agent) based on the graph theory [168]. As there not many works or post-experimental evaluation, evaluation metric for a DWSN system is required. In this context, the evaluation criteria for a DWSN system are provided below: 1) how well the total communication load is distributed across the network system, 2) how efficient the system is in terms of processor/memory usage and power consumption, 3) how fast the system can transmit the data, 4) how large the sensing coverage is, and 5) how scalable the network size is. This metric is calculated based on how entropy is maximized when the probability distribution is uniform. Each probability accounts for the time, size, or rate of each agent’s performance in a whole system. The performance parameters will include the following:

For each applications, values of the addressed parameters are collected in the Table 8.2.

To evaluate how well a system is distributed with respect to the performance parameters, information-theoretic entropy has been adopted based on the probabilistic approach.

\[ P(\text{Param}_i) = \frac{\text{Param}_i}{\sum_{i=1}^{N} \text{Param}_i}; \quad (8.1) \]
Table 8.1: Performance parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$DT$</td>
<td>Length of data transmitted and received wirelessly (byte)</td>
</tr>
<tr>
<td>$CD$</td>
<td>Length of collected data through embedded sensors (byte)</td>
</tr>
<tr>
<td>$PC$</td>
<td>Total amount of power consumption (W)</td>
</tr>
<tr>
<td>$SC$</td>
<td>Equivalent length of sensing coverage (inch)</td>
</tr>
<tr>
<td>$CR$</td>
<td>Number of communication routes</td>
</tr>
<tr>
<td>$PU$</td>
<td>Processor usage (%)</td>
</tr>
<tr>
<td>$MU$</td>
<td>Memory usage (%)</td>
</tr>
</tbody>
</table>

Table 8.2: Parameter values of the applications

<table>
<thead>
<tr>
<th>Parameter</th>
<th>TAG-Games</th>
<th>Swarm robots</th>
<th>Wearable sensors</th>
<th>Origami robots</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_{agent}$</td>
<td>10</td>
<td>50</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>$DT$</td>
<td>4(9), 6(1)</td>
<td>4(50)</td>
<td>2(10)</td>
<td>66000(1), 4(1), 66004(1)</td>
</tr>
<tr>
<td>$CD$</td>
<td>13(9), 54(1)</td>
<td>6(50)</td>
<td>33(10)</td>
<td>66000(1), 30(1), 0(1)</td>
</tr>
<tr>
<td>$PC$</td>
<td>0.95(9), 60(1)</td>
<td>2.85(50)</td>
<td>0.396(10)</td>
<td>4(1), 0.5(1), 60(1)</td>
</tr>
<tr>
<td>$SC$</td>
<td>1(9), 0(1)</td>
<td>60(50)</td>
<td>130(10)</td>
<td>8(1), 0(1), 0(1)</td>
</tr>
<tr>
<td>$CR$</td>
<td>4(9), 9(1)</td>
<td>4(50)</td>
<td>10(10)</td>
<td>1(1), 1(1), 2(1)</td>
</tr>
<tr>
<td>$PU$</td>
<td>0.9(9), 0.03(1)</td>
<td>0.45(50)</td>
<td>0.95(10)</td>
<td>0.42(1), 0.3(1), 0.08(1)</td>
</tr>
<tr>
<td>$MU$</td>
<td>0.74(9), 0.0001(1)</td>
<td>0.68(50)</td>
<td>0.73(10)</td>
<td>0.8(1), 0.4(1), 0.02(1)</td>
</tr>
</tbody>
</table>
Table 8.3: Evaluation of the applications

<table>
<thead>
<tr>
<th>Parameter</th>
<th>TAG-Games</th>
<th>Swarm robots</th>
<th>Wearable sensors</th>
<th>Origami robots</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H(DT)$</td>
<td>2.2935</td>
<td>3.9120</td>
<td>2.3026</td>
<td>0.6935</td>
</tr>
<tr>
<td>$H(CD)$</td>
<td>2.1270</td>
<td>3.9120</td>
<td>2.3026</td>
<td>0.0040</td>
</tr>
<tr>
<td>$H(PC)$</td>
<td>0.6503</td>
<td>3.9120</td>
<td>2.3026</td>
<td>0.2774</td>
</tr>
<tr>
<td>$H(SC)$</td>
<td>2.1972</td>
<td>3.9120</td>
<td>2.3026</td>
<td>0.1785</td>
</tr>
<tr>
<td>$H(CR)$</td>
<td>2.2582</td>
<td>3.9120</td>
<td>2.3026</td>
<td>1.0397</td>
</tr>
<tr>
<td>$H(PU)$</td>
<td>2.2135</td>
<td>3.9120</td>
<td>2.3026</td>
<td>0.9364</td>
</tr>
<tr>
<td>$H(MU)$</td>
<td>2.1974</td>
<td>3.9120</td>
<td>2.3026</td>
<td>0.7097</td>
</tr>
</tbody>
</table>

where $\text{Param} = DT, CD, PC, SC, CR, PU, \text{and } MU$.

$$H(\text{Param}_i) = -\sum_{k=1}^{N} P(\text{Param}_i) \log P(\text{Param}_i)$$  \hspace{1cm} (8.2)

According to the evaluation result, the swarm robots and wearable sensors are equally distributed as each deployed agent is designed and programmed to be equal. The TAG-Games is well distributed except for the total amount of power consumption ($H(PC)$) as it requires a desktop or laptop for monitoring the system. The evaluation revealed that the origami robots are not highly distributed system as they adopted few agents.

8.2 Future Work

TAG-Games

The limitation of the human subject study included limited validity due to the small sample size and the demographic characteristics of the participants. All participants
Figure 8.1: Evaluation of four applications

were undergraduate or graduate students at the university where most of the male participants were from the School of Engineering and most of the female participants were from the School of Arts and Science. In addition, the study focused on young, healthy adults aged between 18 and 30 where no significant age-related cognitive differences are expected. Therefore, the gender/age differences were not analyzed in the outcome measures and rather focused on evaluating preliminary reliability and validity of the computational measures and assessment functions. Also, the current version of the SIG-Blocks system is not necessarily more engaging than other existing systems or instruments, however, by integrating with augmented sensory feedback mechanisms, such as programmable LED/LCD display on the surface, tactile feedback using a vibration motor, and a speaker/buzzer for sound feedback it can be more fun to play with. TAG-Games can also feature enhanced graphics and animations and incorporate rewarding mechanisms to make the games more interesting and engaging. Such features become more important if this technology is used for continuous rehabilitation or learning. Nevertheless, the SIG-Blocks technology is uniquely positioned among other existing sensor-embedded blocks in terms of its communication strate-
gies and real-time wireless measurement capabilities. It utilizes low-cost IR sensors for local communication and a ZigBee-based protocol for global communication. One of the challenges in measuring intangible personal attributes, such as cognitive, memory, and motor skills is the difficulty in addressing individual differences and health status. A large-scale human subject evaluation for fine-tuning the game items and further testing the reliability and validity will reveal the potential of the SIG-Blocks technology for cognitive assessment. Once fully evaluated, the SIG-Blocks technology has the potential to provide advanced instrumentation for various research, clinical, and educational applications such as clinical psychology, early-childhood education, and cognitive and motor skill assessments with automated, objective, cumulative and real-time data collection that enables remote and continuous assessments. In addition to the young adult human subject study, a study with 40 children from 4 to 8 years old is finished and the collected data will be analyzed. More recently, TAG-Games has been used for patients with TBI (Traumatic Brain Injury) at the VA hospital in Cleveland. Once the clinical effectiveness is validated, the TAG-Games will broaden their utility in rehabilitation purposes as well.

**Swarm robots**

Future research problems include the power management for many robots as the size of the robot limits the battery capacity which allows limited operation time. One method may be to provide the power from the experimental platform. For example, a conductive floor and ceiling with a separate power polarity will contact the robot, thus forming a complete charging circuit.

**Wearable sensors**

Design of the real-time monitoring and continuous data storing system connected to each other was challenging due to limited processing capability of embedded con-

176
controller. The controller has to collect data with different sensing protocols (I2C, SPI), and process the wirelessly received signals. Also, integration of sensors and networking modules from the market made the system bigger although the system will be used in research purposes. The future plan needs to include the novel design of the hardware and customized sensor or networking modules. For example, the pick and place service for SMD types of components will be required to build the main board to be more compact and ergonomic design.

**Origami robots**

Miniaturization of the OrigamiBot will allow a multi-robotic experiment with low-costs such as in the swarm-robotic study. To allow for miniaturization of the structure, new actuating mechanisms will be studied such as SMA (shape memory alloy)-based locomotion or magnetic force control. Regarding the vision processing, the current Raspberry Pi B+ will be replaced with a model 2 to make processing faster and embed more computationally intensive algorithms.
Bibliography


[35] H. H. Lund and C. B. Nielsen, “Modularity for Modulating Exercises and Levels - Observations from Cardiac, Stroke, and COLD Patients Therapy,” 8th Inter-
national Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 2011.


182


Vita

Biography

Donghwa Jeong has worked as a Ph.D. student at Distributed Intelligence and Robotics Laboratory (DIRL) with Dr. Kiju Lee since 2009. He received his Bachelor of Science degree in mechanical engineering from Hanyang University, Seoul, Korea in 2009. During his undergraduate program at Hanyang University, he received National Science & Technology Scholarship for all four years for academic excellence. During his time in Hanyang, he served as a president at robotics research club and vice-president at invention club at 2007 and 2008, respectively. His work focused on designing and programming humanoid robots, and he built his first bipedal robot in 2004. After he came to Case Western Reserve University, he served as a mentor for FIRST robotics competition which is an internationally recognized event in 2010. He was awarded the Research Publication of the Year from the department of mechanical and aerospace engineering in 2015. His research interests include DWSN system, internet of things, networked robots, telerobotics, origami robots, and development of clinical device as well as human subject study.

Publications

1. D. Jeong, E. Kerci and K. Lee, “Sensor-Integrated Geometric Blocks: Towards Interactive Play-Based Assessment of Young Children,” CHI Workshop on In-


