Millibot-Miniature Mobile Robot Platform for Scalable Swarm Robot Research

Phan Gia Luan and Nguyen Truong Thinh *

College of Technology and Design, University of Economics Ho Chi Minh City–UEH University, Vietnam;
Email: luanpg@ueh.edu.vn (P.G.L.)
*Correspondence: thinhnt@ueh.edu.vn (N.T.T.)

***Abstract*—Applying control algorithms in a decentralized manner to large-scale robot systems is a growing area of research in swarm robotics. However, due to the manufacturing and installation costs, implementation time, and user operating limitations on large-scale swarm robots, the experiment's applications in these studies are currently relatively constrained. Most proofs of the algorithm's viability in these studies end with simulations or implementations on a small number of robots. Following the majority of platforms for swarm robots shown in previous studies, a design of our low-cost small mobile robot platform, the Millibot, which is consistent with those algorithms for mobile robot swarms is introduced in this work. Moreover, in an effort to allow operation easier for users, even when the robot swarm consists of hundreds of individuals, fundamental swarm scalable functions consist of scalable user-assigned power mode, automatic charging and scalable upgrade firmware operation will also be introduced and implemented to swarm of Millibot. The results show that using proposed platform enables users to build and control swarm robots of up to hundreds of individuals.**

***Keywords*—swarm robotics, robot platform, Millibot, scalable swarm robots**

I. Introduction

Swarm robotics is a multidisciplinary study field exploring bio-inspired cooperative control strategies that distributed govern large numbers of homogeneous and simple robots. Scalability, flexibility, and robustness are the main benefits of swarm robotics since it only uses decentralized control based on local communication and information [1]. Numerous distributed controllers that respond to certain collective behaviors of swarms such as self-organized aggregation [2–4], collective transport [5, 6], collective foraging [7, 8], etc. have been the subject of much investigation. Simulation is used for the majority of algorithm assessments in previous studies due to the cost, time, and operational complexity of actual swarm robots. Unfortunately, for the great majority of multi-robot simulators, scalability in terms of swarm size is not the primary priority [9]. Scalability is the capacity of a self-organized system to sustain greater or smaller populations without significantly affecting its performance. Although the swarm can function within a certain range at acceptable performance levels, this range should be as broad as possible [10]. Scalability validation typically necessitates testing the control algorithms on more than a hundred robots. Meanwhile, it's easy to see that operating a robot swarm, which has a size of up to thousands of individuals, with humans is substantially more difficult than directing a single robot. This leads to a lot of difficulties in implementing control algorithms on actual swarm robots.

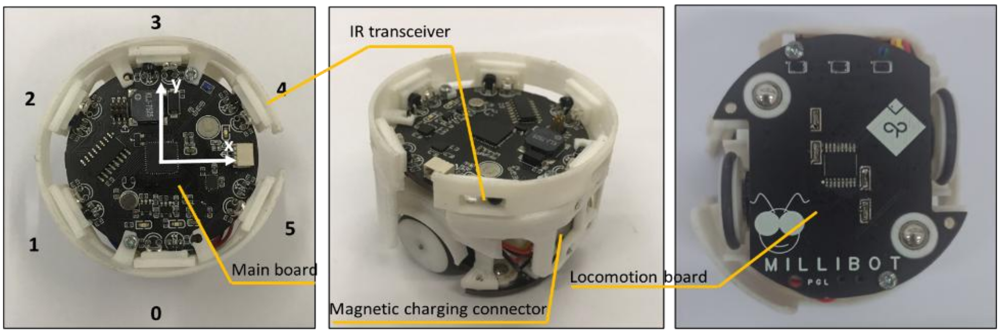
The majority of swarm robot investigations require a high number of individuals needed, resulting in real swarm robot experiments becoming too expensive. To address this issue, some studies limit the complexity of the functionality that can be examined and the actual potential of swarm robots by testing with swarms of just a few robots [5, 11–14]. Other researches focus on bringing down platform costs in order to make the scalability test more practical. However, the control algorithms that may be examined are likewise constrained due to the limited capabilities of the robots themselves. Even while the cost of the robot makes it feasible to assemble swarms of up to thousands of robots, running the entire swarm simultaneously remains a challenging issue. Even though it just takes a few seconds for each robot to execute, manipulating the whole swarm can take a lot of time. The complexity of operating whole swarm robots simultaneously, including charging, controlling, switching on/off, and flashing programs, also has an impact on the scalability of swarm.

To enable these functions in a scalability manner, robot used in swarm should have suitable components. Several robot platforms have been developed to enhance user control easier when the swarm size increases. The e-puck is a small-scale fully-integrated extendable swarm robot platform that is commercially available [15]. However, at roughly $1,000, the robot is pretty pricey for swarm robot applications. Even though other robot platforms such as the Colias series [11, 12], Pi swarm series [16], eSwarBot [17], Kobot [18], AMiR [13], Alice [19] and Jasmine [20] are more affordable, the complexity of controlling, programming, charging, and power switching of the robots goes still unnoticed. Among recent swarm robot platforms, Kilobot [21] has received a great deal of attention. Due to the price of individual, Kilobot is suitable for creating swarm of thousands of individuals.

Manuscript received May 8, 2023; revised July 22, 2023; accepted September 5, 2023.

doi: 10.18178/ijmerr.12.6.417-424

Studies using large numbers of robots are also applied to Kilobot since it supports scalable basic functions. However, the configuration of Kilobot platform is incompatible with many control algorithms that demand a high degree of mobility or require a range & bearing system.



(a) (b) (c)

Figure 1. Millibot, a miniature mobile robot platform (a) The top view of robot platform (b) The side view of robot platform (c) the bottom view of robot platform.

Our objective is to create a platform providing adequate functions which can be customized by users to conduct a variety of multi-robot research experiments. In this study, we introduce Millibot, a miniature mobile robot platform illustrated in Fig. 1, which costs only \$23 for each individual. Millibot has high similarity with existing mobile robot platforms in previous studies whose design is clarified in the Section II. Moreover, some basic scalable functions which consists of changing power mode, automatic recharging, and mesh-based firmware upgrading will also be considered in Section III. Based on our validation, in Section IV, the results and conclusion will be stated.

II. Hardware Design

The three main goals that guided our design of the Millibot can be summarized as follows: (1) To enable the scalability of swarm's basic operations: power switching mode, automatic charging and collective firmware upgrading; (2) To have more options in choosing communication system which is a key to compatibility with many control algorithms; (3) To help swarm robots to be able to perform long-term tasks. Resulting of the goals, the Millibot robot design, a low-cost and small-size miniature mobile robot platform, which delivers the fundamental components for swarm robotic applications such as locomotion, power-management, inertial measurement unit, infrared (IR)-based and sound-based inter-robot communication system, in addition to providing Wi-Fi-based global communication systems is proposed.

Millibot platform shown in Fig. 1 is a cylindrical mobile robot that has a diameter of 55 mm and a height of about 40 cm. The overview of electronic components of Millibot is illustrated in Fig. 2. Table I shows the list of Millibot's components and their price. Millibot's body is composed of two circular printed circuit boards (PCBs): Main board and locomotion board–which have specific indispensable functions. Main board accommodates its modules including main controller integrated Wi-Fi and Bluetooth functionalities, sensors and hybrid infrared-acoustic inter-robot communication system. Meanwhile, locomotion board is full of ingredients that help in robot's movement. The majority of the enclosure's components are 3D printed using common PLA-F material, making it simple to duplicate and modify the core. The In the following subsections, some important components of Millibot design will be clearly introduced which include: locomotion, main controller, power management and inter-robot communications.

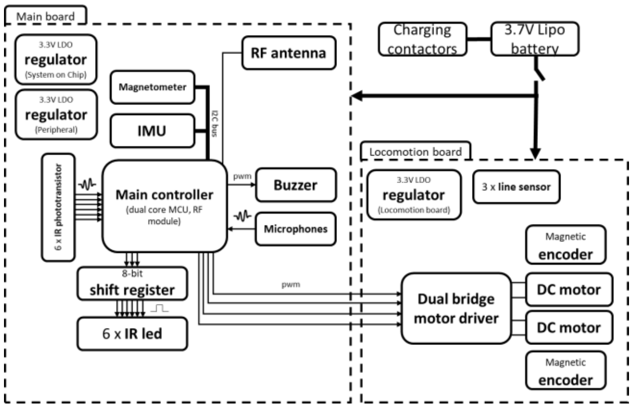


Figure 2. The two boards of Millibot's electronics: The main board and the locomotion board.

TABLE I. A Summary of Millibot Components Cost.

Category	Cost
Main controller	\$3.41
Locomotion	\$2.9
Power	\$4.01
IR-based communication system	\$0.24
Audio-based communication system	\$1.62
Inertial measurement unit	\$2.68
Mechanical part	\$1.25
Connectors	\$1.80
Miscellaneous	\$3.9
Total	\$21.81

A. Main Controller

Millibot is powered by Espressif's System on Chip (SoC) ESP32 series which has dual-core 32-bit microprocessors at 240 MHz and full features of Wi-Fi and Bluetooth functionalities. In order to reduce the size of main board, ESP32-PICO-D4 is selected since it integrated all peripheral components seamlessly, including a crystal oscillator, 4MB flash, filter capacitors and RF matching links in one single package. This microcontroller has many advantages in swarm robotics:

- A single ESP32 has 34 general-pupose input/output (GPIO) pins which can be assigned various functions by programming the appropriate registers, allowing it connects to all of the peripheral of Millibot.
- ESP32 provides 12-bit Successive-approximation-register (SAR) analog-to-digital converters (ADCs) and supports measurements on 18 channels. Furthermore, these ADC channels can directly send data to the memory without a CPU via the Direct Memory Access (DMA) transfer function. It is suitable for the simultaneous handling of multiple infrared transceivers in communication system of Millibot.
- With the use of advanced power-management technologies, ESP32 can switch between different power modes. It is required for designing the power optimization controller which is the most important component to help robots operate in the long term with little user intervention. This controller will be considered in more detailed in Section III.A.
- ESP32 supports wireless upgrade firmware via over-the-air (OTA) technique and mesh networking which can extremely reduce the user's physical interaction with individual robots.

B. Locomotion

The Millibot platform employs two micro speed-reduced DC motors (9161C) placed non-colinearly to minimize the total size of robot. Two ball casters are added to help balance robot which also placed non-colinearly produced skid-steered model. Due to its mechanical stability and accessibility, the skid-steered model is one of the most popular types of platforms used in field robots [22]. The complexity of the controllable wheel-terrain interaction models needs to represent the surface friction as the robot moves in a curved manner is a significant problem with these platforms. These models are complicated because a skid-steered platform's wheels must slide and/or skid when conducting a curvilinear motion. However, due to the computational limits of individual in swarm robots, curvilinear motion is rarely used in applications. For translation and in-place rotation, even though the motors do not rotate around the same axis, the robot still has the same net force and moment as would a robot with colinear motors. Hence, the two-wheel differential drive model is still applicable in our case. Let V and ω represent the angular and linear velocity of robot respectively, the right and left wheel velocities, $\dot{\theta}_r$ and $\dot{\theta}_l$, can be determined via Eq. (1).

$$\begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} = \begin{bmatrix} \frac{1}{r_w} & -\frac{r_R}{r_w} \\ \frac{1}{r_w} & \frac{r_R}{2r_w} \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \tag{1}$$

where r_R and r_w be the distance between two wheels and radius of each wheel respectively.

To drive the robot, a dual H-bridge driver (DRV8833) which is integrated circuitry to limit the winding current for each H-bridge is employed. Driver power supply voltage is regulated at 3.3 V and the maximum current provides to each motor is limited to about 133 mA. With the wheel diameter of 18 mm, the swarm has an average maximum speed of approximately 25 cm/s. In addition, the driver's inputs can be utilized to control the motor speed by using 8-bit pulse-width modulation (PWM) integrated in main controller and speed feedback is provided by a single channel magnetic encoder on each motor.

In order to help robot to move in the desired trajectory, two independent-velocity closed-loop controllers (PID controllers) for each wheel is embedded. The major issue in our design is the feedback signals provided by the single-channel magnetic encoder whereas almost design employing at least dual-channel encoder. Hence, conventional controllers obviously can only determine speed but not velocity, resulting in robot cannot reverse the direction of wheels.

To be able to control the motor to rotate in the desired direction, the virtual directional sign is introduced. The main idea of this method is to create a temporary setpoint at 0 for the PID controller when the motor reverses its rotation. Let $s \in \{1, -1\}$ be the virtual directional sign of a wheel, the measured wheel velocity can be defined by Eq. (2).

$$\dot{\theta} = s\frac{2\pi r_w}{T} m_g \tag{2}$$

where T be the time for one revolution of the motor shaft and m_g be the gear ratio of motor. Virtual directional sign will remain constant until the motor stops. At that moment, the desired setpoint will be set and s will be change. By that way, the response time of PID controller will be increased which depends on response time of zero-speed detection method. However, due to hardware limitation, the zero-speed is not detected, only the exceed of T above a given threshold T_0 can be detected. T_0 depends on minimum controllable speed of motor. In our case, with m_g=1:171 and average minimum linear speed of robot is 10 cm/s, T_0 is set to 40ms.

C. Power Management

High-compact 3.7 V 2-series Li-Po cell battery pack with 1000 mAh capacity is used for powering Millibot. The electronic components in the robot have different operating voltage ranges but can all work normally at 3.3 V so low-cost Low Drop-Out (LDO) regulators are used. RT9013-33G is the most suitable LDO regulator with wide operating voltage ranges from 2.2 V to 5.0 V and output fixed at 3.3 V. Specifically, drop-out voltage is 250 mV at 500 mA.

The highest amount of battery consumption is used by RF system in main controller and DC motors which can peak at 240 mA and 270 mA, respectively. If the locomotion board, communication systems and all sensors continuous works together, it can draw up to 620 mA. Meanwhile RT9013, just can provide at most 500 mA which is not sufficient to operate Millibot in this state. Hence, three 3.3 V voltage LDO regulators in parallel are used in which one is always enable to powering main controller, the two remaining are controllable (one for other components in mainboard, another for entire locomotion board). This approach isolates load in main controller and peripheral, increases stability of robot. Moreover, these regulators can be switched on and off by the microcontroller, enabling shutdown of the locomotion board and the communication system to conserve power consumption.

Millibot's power management give autonomy of more than two hours for the robot in saturation state (in which all sensors, locomotion board, communication system and microcontroller has the state of high energy consumption). Furthermore, this battery series features good capacity retention, a low self-discharge rate of less than 5% per month and the quiescent current of RT9013 peaks at 24 uA which is suitable for later designed robot power modes. The robot power mode will be introduced in Section III.A.

D. Inter-robot Communication System

1) IR-based short-range communication system

Compared to other wireless communication approaches, including radio frequency, audio and vision, infrared is an appropriate option as an inter-robot communication channel for robotic swarm applications due to its scalability. There are three advantages of the IR-based communication system compared to others:

- It can be both directional communication and broadcast communication depending on the application.
- It is situated communication, where the signal that transmits information also contains information, specifically the relative direction and distance of the transmitter with regard to the receiver. Position estimation, neighboring robot recognition, direct communication, and obstacle avoidance are all benefits of adopting IR in swarm applications.
- Due to the requirement of line-of-sight, IR-based communication system locally transmits robot's data to another. Hence, the load of communication channel does not depend on swarm size, i.e., its scalability.

Millibot is provided with 6 pairs of IR transceivers uniformly distributed around robot's main board. Each transceiver consists of one IR transmitter and one IR phototransistor having a wide-open angle of about 60^0. All IR transmitters are controlled via 8-bit serial-in, parallel-out shift registers which can provide up to 1 kHz output transition. 6 IR phototransistors directly connect to 6 separated 12-bit analog channels of main controller. Thanks to the direct memory access transfer function, data from these analog channels can queue directly to memory without interference from the CPU, resulting in maximum sampling frequency in all channels being up to 20 kHz. At 1 kHz transmit rate, in our testing arena, for a distance of up to 45 cm, data transfer is possible in all IR transceivers meanwhile the reflection signals can be recognized at distance of 15 cm.

The interference of IR signals between robots is a problem with both sensing and communication systems that use shared hardware. Without good interference detection, robot may not achieve a task due to failure in sensing tasks. Hence, continuous turned-on emitters for sensing are not allowed. Most previous studies used pulse modulation techniques that can provide meaningful messages between robots and reduce interference between robot IR signals. According to the application, the pulse modulation technique can be modified for being compatible with their proposed communication protocol. In this study, we do not put attention to communication protocol but the way to obtain situated information from IR-based communication system.

We begin with the obstacle distance estimation task of Millibot which is based on basic concepts of electromagnetic radiation and its reflections. The sensor output $p(d,\alpha)$ as a function of the distance d and the angle of incidence α with the target surface can be modeled using a straightforward Eq. (3) [23].

$$p(d,\alpha) = \frac{a_c}{d^2}\cos(\alpha) + b_c \tag{3}$$

where a_c and b_c are the model parameters. The a_c depends on the transmission medium and efficiency of IR transceiver which can be estimated empirically. Meanwhile, b_c represent for sensor output without present of IR emission. Hence, it can be taken when IR emitter is turned off.

In the presence of another robot's IR signal, the bearing can be estimated first. For the estimated bearing β that follows Eq. (4), a vector sum is used:

$$\beta = \text{atan2}\left(\sum_{l=0}^{5} p_i \sin\left(\frac{i\pi}{3}\right), \sum_{l=0}^{5} p_i \cos\left(\frac{i\pi}{3}\right)\right) \tag{4}$$

where β and $\{p_0, p_1,\ldots, p_5\}$ be the estimated bearing angle and the peak of translated intensity signal with respect to IR transceiver order in Fig. 1 respectively. According to [24], the estimated distance from the receiver to the emitter $\tilde{d}$ based on the estimated bearing can be formulated as Eq. (5).

$$\tilde{d} = \sqrt[4]{\left(\frac{p_l}{\sqrt{\cos(\beta - \gamma_l)}}\right)^4 + \left(\frac{p_r}{\sqrt{\cos(\beta - \gamma_r)}}\right)^4} \tag{5}$$

where p_l and p_r are translated intensity signals recorded by the left and right IR transceivers with respect to estimated bearing angle. The γ_l and γ_r are the heading of the left and right IR transceivers with respect to the estimated bearing angle.

2) Audio-based communication system

IR communication is capable of sensing and helping robots communicate with each other, but with a small number of sensors they often have a dead zone in the field of view (Fig. 3). In the case of 2 robots approaching each other, they will likely not be able to communicate with each other because they fall into the dead-zone. In some applications, constant communication is essential. To address the above important requirements of simplicity and effectiveness in building robot swarm applications, we propose an acoustic-based near-field communication system with lightweight computational demand.

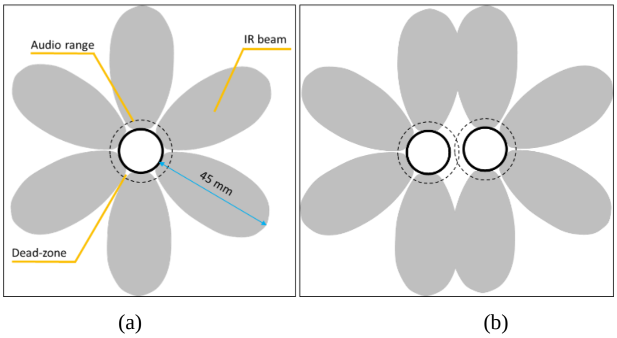


Figure 3. IR-based and audio-based communications field of views, (a): field of views of Millibot with the presence of dead-zone in IR-based communication system (b): Robots cannot recognize based-on IR transceivers each other due to dead-zone.

The acoustic-based communication used in most previous research limits the kinds of environments the robots can be deployed in. Echoes within an enclosed environment and bandwidth limitations for communication frequency and public disturbance due to the sound emitted by the robots can all contribute to this limitation. Hence, our system is designed to have very limited range of communication with maximum distance of 10 mm depending on source amplitude. Hence, the reflection signal is almost impossible to be recognized.

The microphone is directly provided sound magnitude to main controller via 12-bit ADC. The simple Fast Fourier Transform (FFT) is used to extract frequency from microphone. According to application, sampling rate and size of data frame are set up, its default value are 10 KHz and 512 samples respectively. Hence, the default highest measurable frequency is 5 KHz since FFT results are conjugate symmetric.

If sound communication is used, the robots could periodically emit a tone with a certain frequency, in-range robot can recognize this even if the IR communication systems are in dead-zone.

III. Scalable Operations

A. User-assigned Power Mode

As was already stated, operating a large number of swarm robots can be quite time-consuming when utilizing the conventional power on/off switch, which uses a physical switch. Additionally, the robot can be set up in an enclosed area that is big enough for it to operate, preventing interaction between humans and whole swarm. This only allows user's physical interference on the entire robot at the beginning of the experiment. The second drawback of the conventional on/off switch is that it is easy for the robot to run out of battery during the robot experiment without a sufficient power control scheme. Furthermore, in algorithm testing, users often take time for code development, during this time, swarm robots have no operation without turning off or standby mode then will continuously burn out battery. Hence, the design power control scheme is a prerequire for both developing and usable swarm robots. In this platform, we introduced 4 operating modes that can be assigned by the user during the operation process which is clearly illustrated in Fig.4:

- Normal mode: All peripherals are fully powered and operate according to the main program. In this mode, Esp32 is in active mode (All functions work normally according to main program).
- Upgrade mode: All peripherals and locomotion boards of the robot are deactivated. Meanwhile, Esp32 is still in active mode to establish a Wi-Fi-based mesh network which is used to serve the firmware upgrade process.
- Standby mode: Instead of shutting down the robot, this mode is utilized when the user wants the swarm to rest while they perform other tasks like writing code or correcting bugs. The wake-up source was set before the main controller Esp32 went to sleep, and it is in deep sleep mode with only the ULP coprocessor and RTC module running.
- Sleep mode: A minimal number of components are kept active when the robot is in this mode in order to conserve energy while maintaining the system's ability to be awakened by external stimuli. The main controller ESP32 is in deep-sleep mode. In this mode, robot just consume 0.5mAh, so robot can sleep and do not need recharging for more than 2 months. The wake-up source is the different between standby mode and sleep mode.

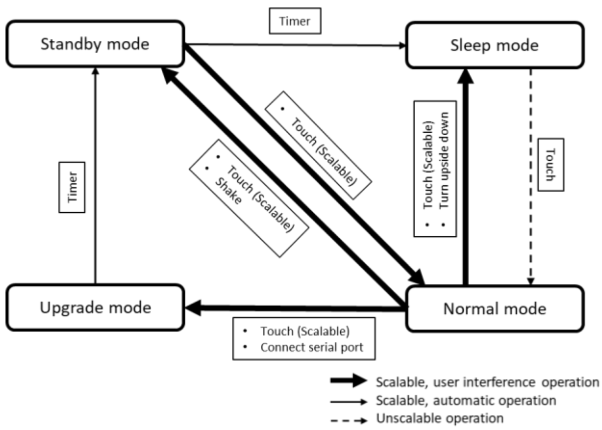


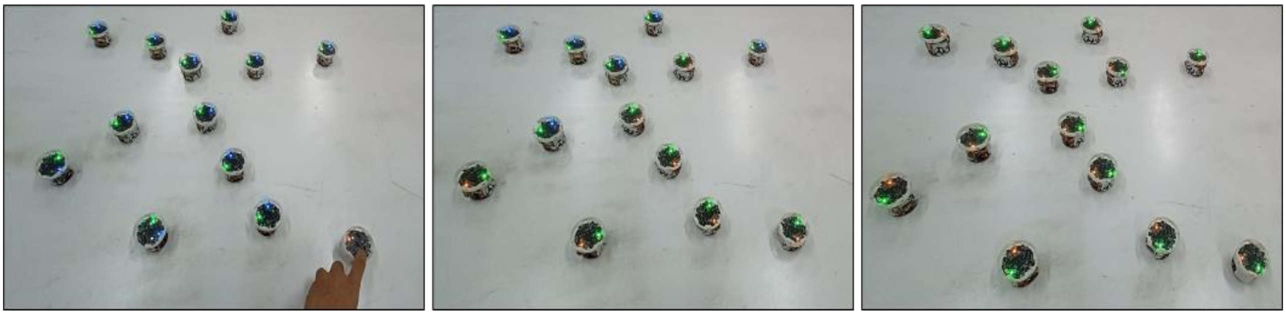
Figure 4. Millibot's diagram of the operating mode switching system

When in normal mode, the robot can change to any other mode depending on how the user interacts with it. A single robot or the entire swarm may be affected by this interaction. In particular, when a user wants to influence the entire swarm, they can use touch sensors to interact with the robot. The robot can identify the destination mode based on the user's touch. The robot will then turn in spot and broadcast an encoded message to all of its nearby neighbors for a while. When more robots acquire the message, they will keep acting in the same way, and eventually, the message will reach the entire swarm. With this approach, the swarm operation can be finished more quickly and with less user involvement. Robots can enter standby mode with a single rapid touch, update mode with a double fast touch, and sleep mode with a long touch of at least three seconds. An example of this scalable operation is illustrated in Fig. 5, in which the user only needs to interact with one robot but the effect is spread all over the swarm.

Additionally, users can switch to standby mode or sleep mode specifically for a certain robot by shaking the robot or flipping it upside down, respectively. The serial port can be used to load the firmware for the robot when a user wants to update the firmware for a single robot.

The main controller registers touch sensor and IR receivers as two wake-up sources before the robot enters standby mode. If the user touches robot or any IR source is detected, standby robots will be wake up. The user simply needs to touch one robot to wake up the entire swarm. As soon as the touched robot wakes up, it turns on all IR emitters, turns in spot for a while and causes all of the nearby robots to awaken as a result of the second wake-up source. In addition, robots go into sleep mode if they are not woken up after three hours to save battery power.

When a robot is in sleep mode, it can only be awakened by touching it; therefore, to perform this operation, the user must touch every robot that is asleep. However, the swarm robots can stay in this state for more than 2 months without recharging. Therefore, using the robot's power switch to turn it on and off is unnecessary when using sleep mode.



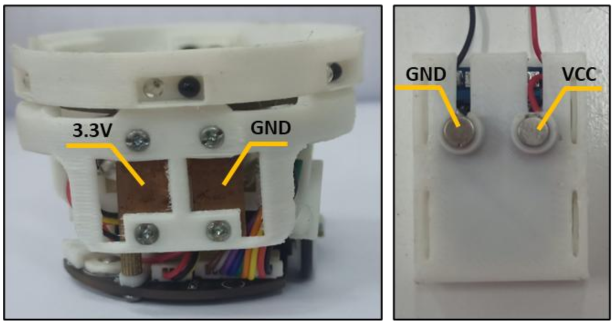
(a) (b) (c)

Figure 5. The snapshots of test switching from normal mode to standby mode. The blue led indicate robot is in normal mode while orange led indicates standby mode. (a) user touches on a robot in swarm to enable standby mode. (b) The "standby" message is spread all over the swarm. (c) The entire swarm is now in standby mode.

B. Automatic Charging System

Even if rechargeable batteries used in swarm robots can last for a few hours, their battery is burnt out by some long-term tasks before swarm robots complete their task. Furthermore, mobile robots typically need human assistance when their batteries need to be recharged. Hence, an automatic recharging task is an essential solution especially when the robot swarm can have hundreds of individuals.

To enable self-charging ability, the docking system is designed and implemented. Metal strips are placed on the charger robot as shown in Fig. 6. The charging station consists of two magnetic contactors directly connect to the charging circuit (TP4056). The charging stations are installed on the wall or any horizontal surface so that robot can be reachable. To assist in navigating the robot to the correct location and direction of the charging station, a black line is glued to the workspace floor for each charging station. Whenever an individual in swarm requires a recharge, it will maneuver itself to the charging station and try to make its contacts get close to the charger station.



(a) (b)

Figure 6. Docking system, (a): Two copper-plated tin strips placed in front of robot act as the female connector, (b): the charging station with two magnet contacts.

The basic method can be described as follow: (1) The robot will interrupt its present task and enter the state of looking for a charging location when the battery falls below a predetermined minimum charging threshold. In this state, the robot will explore the workspace for the pre-installed black line heading to the charging station by reading the signal from the line sensor. (2) After finding the line, the robot will follow the line to the charging station. (3) When it finally arrives at the charging station, the robot will check the change of battery level to know if it's charged or not. As soon as it realizes that it is being charged, the robot will go into sleep mode. (4) Based on available battery power and the given maximum charging threshold, the robot will decide to continue charging or leave the station and return to the current task.

To monitor the battery, one 12-bit ADC channel of the main controller Esp32 is used to measure the battery level during Millibot operation. However, the ADC input voltage limit is 3.3 V so the voltage divider is applied. To reduce noise of ADC input signal, a simple low-pass filter is employed. The minimum safe charge of a 3.7 V Li-po battery is 3.0 V. However, due to cut down voltage and drop-down voltage of regulator, the minimum charging threshold is set to 3.4 V. Meanwhile, the maximum charging threshold is set to 4.1 V.

The robot cannot monitor the battery during sleep mode, hence switching between normal and sleep modes is required while the battery is being charged. After sleeping for one minute, the robot will self-wake-up, check the battery level, and decide whether to go back to the previous task or keep sleeping.

It is essential to notify to prevent a robot from accessing the docking station that is already being used by another robot because the charging station is only intended for 1 robot that can be charged. This notification will be done by the robot. The robot will turn on the IR transmitter in the back before entering sleep mode and keep it on while in sleep mode. Any robot entering the line towards the dock being used will receive this signal and will treat it as an obstacle and quickly leave the line.

C. Mesh Upgrade Firmware

Using the over-the-air (OTA) update mechanism installed in primary controller Esp32, Millibot is able to upgrade the present program wirelessly. The mechanism demands that the robot establish a connection to a server and has at least three predefined partitions in flash memory of Esp32: two OTA app slots and one OTA data partition. Using this functionality, a present program running in one app slot can wirelessly replace the program code in the other app slot. Following the initial OTA update, the OTA data partition is modified to specify which OTA app slot partition needs to be booted next. The main controller finally reset itself in order to replace the robot's program with the new version and go back into standby.

However, this mode is not scalable if every robot in swarm directly connects to a single server to get a new program. To enable OTA update mechanism in the whole swarm, individuals in swarm should have the ability to share program with their neighbors instead of relying on server. ESP-MESH is the most suitable protocol since it is self-organizing and self-healing meaning the network can be built and maintained autonomously. ESP-MESH is a networking protocol built on top of Wi-Fi that permits the connecting of numerous robots (also known as nodes) spread out over a huge physical area under a single Wireless Local-Area Network. The main components of ESP-Mesh shown in Fig. 7 include server, router, root node and other nodes. The root node can directly communicate with server via router meanwhile other nodes branch off from root node. The given node, in turn, is the child's parent, and any sub-node of it is known as a child node. The root node doesn't have parent node. By this topology, the mesh network is scalable and suitable for swarm robot applications such as swarm upgrading firmware in our case.

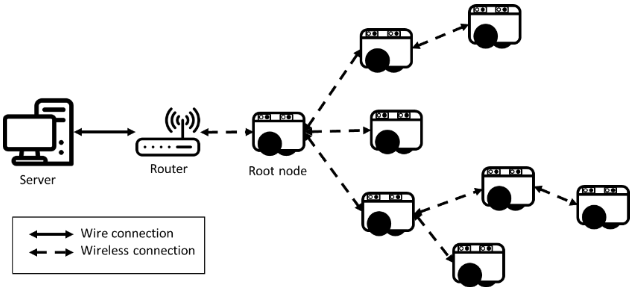


Figure 7. Diagram showing a configuration of mesh upgrade firmware

Many applications rely on mesh networks and each application has its method to share data across the network. In this study, we introduce the mesh upgrade firmware where the flooding technique [25] is used to share firmware across the whole swarm and OTA update mechanism acts as a backbone of method.

Once switched to upgrade mode, the swarm will immediately form a mesh network. Then, root node will request firmware from server and any child node will acquire firmware from its parent. Server sends firmware sector by sector to root and root node streaming down these data to its child. The address of the first sector and length of each sector are stored in all nodes. All children compute a sha256 sum of their downloaded firmware on the fly and send it to their parent, as well as how much of the parent node's firmware they have received. If the parent node receives a message with a sha that doesn't match its own, it starts directly transmitting pieces of its firmware to the slaves. Due to lacking delivery guarantee in proposed mesh network, the parent node keeps track of which pieces of the firmware the children report that they have still not received, and resends them if necessary. Included in the firmware chunks is the parent nodes sha. The child nodes keep track of received pieces and write them to the next OTA app slot if the parent node sha is different from its own. When a child node determines it has written all the pieces, it changes its boots partition and restarts.

The size (total number of nodes) in an ESP-MESH network is dependent on the maximum number of layers permitted in the network, and the maximum number of downstream connections each node can have. Both of these variables can be configured to limit the size of the network. However, the Mesh upgrade firmware is still a scalable operation in the reasonable range of swarm size (up to 1000 nodes).

IV. CONCLUSION

In this work, a miniature mobile robot – Millibot, is introduced with sufficient components for swarm robot research: inter-robot communication system, power management and locomotion. This platform is compatible with the majority of collective behavior researched in previous swarm mobile robot platforms. Specifically, the audio-based near-field communication system is yet to be fully integrated, as an optional sensing system that can help our proposed platform avoid dead-zone of IR communication introduced in most of the platforms mentioned above.

The platform design is low-cost, relatively plain and has size of 50 mm which is convenient for use in a small-scale workspace to develop swarm controllers. Some basic operations on platform are introduced: automatic charging, power mode assigning, and flashing program. These operations are scalable that allow users to perform on a large-scale swarm of Millibot with minimum physical intervention or not even needed.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Phan Gia Luan is the first author who wrote methodology, wring original draft, visualization, validation, configured; Nguyen Truong Thinh is correspondent who reviewed and edited project administration, methodology. All authors had approved the final version.

Funding

This research is funded by University of Economics Ho Chi Minh City–UEH University, Vietnam.

References

[1] H. Heiko, *Swarm Robotics: A Formal Approach*, vol. 221. Berlin: Springer, 2018.

[2] S. Onur and E. Sahin, “Probabilistic aggregation strategies in swarm robotic systems,” in *Proc. 2005 IEEE Swarm Intelligence Symposium*, 2005. SIS 2005.

[3] C. Nikolaus and A. Martinoli, “Modeling and designing self-organized aggregation in a swarm of miniature robots,” *The International Journal of Robotics Research*, vol. 30, no. 5, pp. 615-626, 2011.

[4] Y. Xinan, A. Liang, and H. B. Guan, “An algorithm for self-organized aggregation of swarm robotics using timer,” in *Proc. 2011 IEEE Symposium on Swarm Intelligence*, 2011.

[5] A. M. H. Mohammed, A. Narayan, and E. Tuci, “Cooperative object transport with a swarm of e-puck robots: Robustness and scalability of evolved collective strategies,” *Swarm Intelligence*, vol. 11, no. 3, pp. 185-209, 2017.

[6] W. Sean, et al., “Design of ant-inspired stochastic control policies for collective transport by robotic swarms,” *Swarm Intelligence*, vol. 8, no. 4, pp. 303-327, 2014.

[7] L. Qi, J. P. Hecker, and M. E. Moses, “Multiple-place swarm foraging with dynamic depots,” *Autonomous Robots*, vol. 42, no. 4, pp. 909-926, 2018.

[8] M. S. Talamali, et al., “Sophisticated collective foraging with minimalist agents: a swarm robotics test,” *Swarm Intelligence*, vol. 14, no. 1, pp. 25-56, 2020.

[9] B. Manuele, et al., “Swarm robotics: A review from the swarm engineering perspective,” *Swarm Intelligence*, vol. 7, no. 1, pp. 1-41, 2013.

[10] B. Levent and E. Şahin, “A review of studies in swarm robotics,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 15, no. 2, pp. 115-147, 2007.

[11] A. Farshad, et al., “Colias: An autonomous micro robot for swarm robotic applications,” *International Journal of Advanced Robotic Systems*, vol. 11, no. 7, p. 113, 2014.

[12] H. Cheng, Q. B. Fu, and S. G. Yue, “Colias IV: The affordable micro robot platform with bio-inspired vision,” in *Proc. Annual Conference Towards Autonomous Robotic Systems*, Springer, Cham, 2018.

[13] A. Farshad, K. Samsudin, and R. Ramli, “Development of IR-based short-range communication techniques for swarm robot applications,” *Advances in Electrical and Computer Engineering*, vol. 10, no. 4, pp. 61-68, 2010.

[14] A. E. Turgut, et al., “Self-organized flocking in mobile robot swarms,” *Swarm Intelligence*, vol. 2, no. 2, pp. 97-120, 2008.

[15] M. Francesco, et al., “The e-puck, a robot designed for education in engineering,” in *Proc. the 9th Conference on Autonomous Robot Systems and Competitions*, vol. 1, no. CONF. IPCB: Instituto Politécnico de Castelo Branco, 2009.

[16] H. James, et al., “The pi swarm: A low-cost platform for swarm robotics research and education,” in *Proc. Conference Towards Autonomous Robotic Systems*, Springer, Cham, 2014.

[17] M. S. Couceiro, et al., “A low-cost educational platform for swarm robotics,” *International Journal of Robots, Education & Art*, vol. 2, no. 1, 2012.

[18] A. E. Turgut, et al., “Kobot: A mobile robot designed specifically for swarm robotics research,” Middle East Technical University, Ankara, Turkey, METU-CENG-TR Tech. Rep 5.2007 (2007).

[19] C. Gilles and R. Siegwart, “Mobile micro-robots ready to use: Alice,” in *Proc. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2005.

[20] K. Sergey, O. Kornienko, and P. Levi, “Minimalistic approach towards communication and perception in microrobotic swarms,” in *Proc. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005.

[21] R. Michael, C. Ahler, and R. Nagpal, “Kilobot: A low cost scalable robot system for collective behaviors,” in *Proc. 2012 IEEE International Conference on Robotics and Automation*, 2012.

[22] O. Camilo, et al., “Learning of skid-steered kinematic and dynamic models for motion planning,” *Robotics and Autonomous Systems*, vol. 95, pp. 207-221, 2017.

[23] B. Gines, et al., “Using infrared sensors for distance measurement in mobile robots,” *Robotics and Autonomous Systems*, vol. 40, no. 4, pp. 255-266, 2002.

[24] G. Álvaro, et al., “Open e-puck range & bearing miniaturized board for local communication in swarm robotics,” in *Proc. 2009 IEEE International Conference on Robotics and Automation*, 2009.

[25] A. S. Tanenbaum, D. J. Wetherall, (Computer Networks (5th ed.). Pearson Education. pp. 368–370, March 23, 2010.