

Ant Intelligent Robot: A Versatile and Low Cost Miniature Mobile Robot Platform for Swarm Robotics Research and Education

Dan M. Novischi

University Politehnica of Bucharest
Splaiul Independentei nr. 313,
Sector 6, Bucuresti, ROMANIA
dan.novischi@gmail.com

Adina M. Florea

University Politehnica of Bucharest
Splaiul Independentei nr. 313,
Sector 6, Bucuresti, ROMANIA
adina.florea@cs.pub.ro

ABSTRACT

In this paper we present the Ant Intelligent Robot (AIR), a miniature mobile platform designed for swarm robotic research and education. The proposed system has a modular and distributed architecture that provides the necessary versatility, robustness and user accessibility to enable the study of a broad range of applications, while achieving a low cost. We also present a comparison between AIR and other platforms that were recently used in collective experiments.

Categories and Subject Descriptors

B.8.1 [PERFORMANCE AND RELIABILITY]: Reliability, Testing, and Fault-Tolerance; I.2.9 [ARTIFICIAL INTELLIGENCE]: Robotics – *autonomous vehicles, commercial robots and applications, sensors.*

General Terms

Design, Performance, Reliability.

Keywords

Mobile robot platform, modular design, distributed hardware architecture, small size robot, multi-robot systems.

1. INTRODUCTION

Multi-robot systems, as a subfield of multi-agent systems, have attracted the attention of many researchers in the past decade. The attention received by these systems is twofold: hardware advancements – a typical smart phone of today's world is equipped with a CPU running at several hundreds of MHz – and the expected benefits of their improved efficiency – applications in this respect include autonomous exploration, unmanned search and rescue missions, dynamic target tracking, hazardous environment monitoring, waste cleaning and others. In these applications teams of robots not only have to rely on local information and overcome the inherent problems with imperfect sensors and communication medium, but they must intelligently coordinate in order successfully accomplish the tasks at hand. Moreover, the dynamic of various application environments, time

constraints, robustness, flexibility, cost, size, ease of use and the ever increasing demands of user requirements add to the complexity of designing such robot platforms.

In this paper we present the Ant Intelligent Robot (AIR), a miniature mobile platform that provides the versatility, robustness and flexibility necessary for the study of single- and multi-robot applications, while still achieving a low cost. The modular and distributed architecture of the platform meets key design goals facilitating user accessibility – required in education – and multi-agent system engineering – required in swarm robotic research. A comparison between AIR and similar mobile robot platforms that were recently used in collective experiments is also provided in this paper.

2. RELATED WORK

To date research on swarm robotics has mainly focused on the underlying multi-agent system design. As such, there is a large body of work addressing key application areas: bio-inspired systems [6], coordination [15], task-allocation [27], communication [11], localization and mapping [30] and control approach [9], just to name a few. Reviews on multi-agent research areas for swarm robotics can be found in [19][37] and a more recent review on swarm robotics multi-agent engineering can be found in [28]. In order to support the requirements of these multi-robot systems the individual robot designs are task centric [5][7][13][16][25] or make tradeoffs between: cost, modularity, size, processing power, energy consumption, versatility of sensory systems and communication capabilities, robustness, ease of use and monitoring [1-4][8][12-14][16][19-21][28][30].

Early work on mobile robot designs used in swarm robotics research and education was done by Mondada et al. [1]. This work featured Khepera, a small robot with only a 55 mm diameter, but very limited processing power, flexibility and ease of monitoring. This design was the basis for the subsequent commercial versions: Khepera II, Khepera III and Khepera IV developed by K-Team Co. [20][33]. The Khepera IV robot is targeted toward research applications and costs \$2000+ in parts for the basic configuration, which may inhibit its use in swarm robotic applications. A more versatile and also small size robot design is Alice [3]. To the best of our knowledge, Alice, is the smallest mobile robot to date, having a rectangular shape with a side of only 20 mm. It also has a modular hardware design and multiple robots can be collectively monitored through a radio link. Several other mobile robot designs are relevant to our own: S-bot [2], JL-I [7], ZeeRO [4], Kobot [12], Centibots [5], Bebot [20], E-Puck [10][16][22]. These platforms have processing power ranging from 20MHz – 400 MHz, provide range and local

communication capabilities using IR sensors, have wireless or Bluetooth for long range communication and multi-robot monitoring, are equipped with a vision sensor and have sizes ranging from 120 mm to approximately 350mm in diameter. Some of the designs also have modular hardware architecture which may provide an increased robustness in the face of various hardware failures.

More recent robot designs leverage key desirable features – modularity, processing power, sensor variety communication mechanisms and energy consumption – against cost and size. MarXbot [21] is a design that features some improvements over its predecessor the s-bot [2]. It has more processing power, more energy – with its higher capacity battery and “hotswap” system – and an increased bandwidth for inter-module communication. These improvements are leveraged over the size of the robot – being with 50 mm larger than the s-bot. The platform was used in “Swarmanoid” project [25] and other related projects [23]. Flockbots [30] is a low cost design (approximately 500\$) that uses off-the-shelf components and is built around two of the most popular embedded boards: Raspberry PI [31] and Arduino [38]. The sensory system is composed of a pan-tilt camera system and 5 IR sensors. In the extreme of low cost is Kilobot [24] with a manufacturing price of only \$14 in parts. The robot size is only 33 mm and it uses a slip-stick locomotion system based on vibration motors. Communication and sensing is limited and is accomplished through a single IR sensor. Wolfbot [28] is another low cost robot design (\$550 in parts) with an approximate diameter of 178 mm that is built around the popular Beaglebone embedded board [32]. The design incorporates an accelerometer, a vision sensor with integrated microphone, 5 IR sensors and wireless communication. Navigation with the Wolfbot is achieved through an omni-directional drive system.

Finally, in designing the AIR platform we also considered self-reconfigurable robot designs such as those presented in [17][19]. Design challenges for this type of systems are presented in [14] and a review of architectures and existing platforms to date was done by Yim, Mark et al. in [8].

3. DESIGN GOALS

In order to provide the robustness, flexibility and scalability required by multi-robot research applications, as well as ease of use, versatility and maintainability for use in education at a low cost, we considered design aspects along three directions, namely: multi-agent system engineering, educational requirements and the manufacturing process.

3.1 Multi-agent System Engineering

In swarm robotic research applications, multi-agent system engineering plays a crucial role, as it shapes key aspects of the robotic swarm, such as: what are the underlying physical mechanisms used in self-coordination and task allocation, what schemes of interaction and/or communication are required, what are the characteristics of the environment in which robots will act and many others. To support the research endeavors for these systems in designing our platform we made the following considerations:

1. The physical environment in which the robotic swarm will act is: partially observable, stochastic, dynamic and continuous. This assumption is based on the several key facts, namely: real sensors and actuators present inherent imperfections, the state of the environment may change both with and without the swarm acting

upon it and evolves over time according to laws of physics with some degree of randomness.

2. Robots are multipurpose, have a variety of skills, are able to interact using some form explicit and/or implicit (direct or indirect [36][37]) communication and are not intentionally adversarial. In order to support various tasks, such as: collaborative manipulation, multi-robot localization and mapping, search and rescue, robots within the swarm must possess a variety of perceiving mechanisms and must be able to communicate to achieve coordination.

3. Individual robot failures can be detected and communicated to the swarm with a probability greater than zero. This is a very desirable characteristic, as it directly influences the robustness of the swarm. Thus, a state of the art robot design should include mechanisms for detection and communication of sub-system failures and execution of actions.

4. Robots within the swarm must be able to act in a real-time manner according to the application specification in order to successfully meet the application goals according to some optimization criteria (e.g. lowest possible energy consumption, achieve a certain goal in a given time, etc.). Thus, the platform should be equipped with a considerable amount of processing power in order to support processing intensive algorithms employed by agents that run on the robots.

5. The individual robots must have a relatively small size in order to be able to accommodate a large number of them in a relatively limited space.

6. The design must support tools for multi-robot programming and monitoring in order to facilitate the development of multi-agent applications.

3.2 Educational Requirements

For educational purposes our robot design assumptions were based on maximizing the students experience while providing a broad range of engineering applications, namely:

1. The software tools and middleware for the robot platform should provide a comprehensive and easily accessible interface for the students. This feature is aimed at providing a self contained programming environment and easy access to robots hardware while imposing minimal limitations for use of other software tools.

2. The platform should be versatile enough to provide a wide range of educational applications, such as: signal and image processing, embedded programming, automatic control, distributed systems, sensor fusion, planning, machine learning, etc...

3. The energy management of the design should maximize the running time versus the off-time in order to increase the user experience and the robot reliability.

3.3 Manufacturing Process

From the manufacturing stand point our design goals are based on cost, ease of assembly and maintainability:

1. The design should be modular in order to facilitate easy assembly and/or modification.

- The composing modules must have a simple enough design in order to allow easy repair and maintainability.
- The design should be as low cost as possible in order to be affordable even for single unit manufacturing. This also guarantees low costs for repair or unit replacement.

4. AIR PLATFORM

The Ant Intelligent Robot (AIR) in Fig. 1 is a miniature platform of rectangular shape with a side of 100 mm. The platform is composed of several modules, namely: main processing unit, range and local communication, vision, audio and navigation. Each module represents a single independent unit that has its own processing power, communication interface and power regulation. This distributed architecture (see Fig. 2) is essentially different from the classical single processing unit approach [1][3][5][16][24][28][49] and has several advantages:

- It provides the ability to physically change the skill set of the robots without the need to for hardware redesign by simply adding or removing certain modules.
- Periodic and preemptive tasks necessary for signal processing, motion control and many others can be distributed among the composing modules reducing the computational load for the main processing unit.
- It provides the ability to sense and communicate individual module malfunctions and action execution faults improving the robustness of the robotic swarm (see Software subsection).
- It has a higher degree of flexibility for assembly and maintenance related tasks.

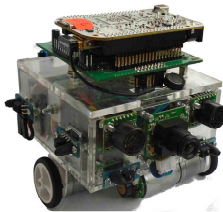


Figure 1. Ant Intelligent Robot Platform.

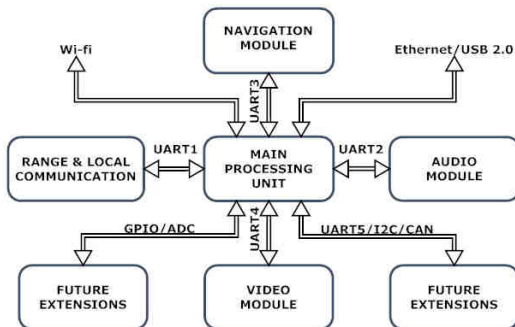


Figure 2. Platform hardware architecture.

In the order to achieve a low cost in the implementation of our platform we used both off the shelf and custom made components. Inter-module communication is provided through a serial UART

connection (see fig 2). Although higher data rates can be achieved through CAN-bus, which our platform supports, we used serial communication in order to maintain flexibility in choosing off the shelf components. Future extensions of our platform such as adding a gripper is supported through GPIO, ADC, I2C, UART and CAN busses.

The base of our robot (Fig. 3) is a low cost solution of only \$4. It is composed of six acrylic plastic parts corresponding to the sides of the robot, interconnected through metallic angled brackets. This base houses all components required by our design and facilitates easy access to hardware components for repairing and maintainability purposes.

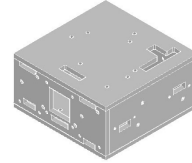


Figure 3. Base isometric CAD drawing.

4.1 Main Board

The main board in Fig. 4 is a custom made printed circuit board (PCB) that holds the main processing unit, the audio module and the communication interfaces for the other modules and extensions. It also provides the necessary circuitry for battery management, signal conditioning and status LED's.

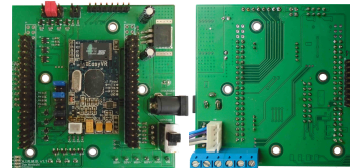


Figure 4. Base isometric CAD drawing.

Table 1. Beaglebone vs. Raspberry Pi comparison

| Feature | Beaglebone | Raspberry Pi |
|-------------|---------------|--------------|
| Proc. Type | Arm Cortex-A8 | ARM11 |
| Proc. Speed | 1GHz | 700 MHz |
| RAM | 512 MB DDR3L | 512 MB SDRAM |
| GPIO | 69 | 12 |
| ADC | 7 | None |
| UART | 5 | 1 |
| CAN | 1 | None |
| I2C | 2 | 2 |
| USB | 2 | 2 |
| Ethernet | 1 | 1 |
| Min. Power | 1.05W (210mA) | 3.5W (700mA) |
| Cost | \$45 | \$35 |

For the main unit we used the Beaglebone [32] single board computer (SBC) which comes in two versions. Both of these versions are supported by our mother board. The choice of Beaglebone over similar SBC's such as the Raspberry Pi [31] was based on its more powerful CPU (ARM Coretex-A8), lower power consumption and available interfaces. Table I shows a comparison between the two boards.

4.2 Perception

Perception capabilities of the Ant Intelligent Robot are distributed among four modules, namely: range and local communication, vision, audio and navigation. Each module maintains up-to-date sensed information that is provided to the main processing unit on a request basis.

For vision we used an off the shelf LinkSprite camera [40] equipped with a 90 degree lens that can capture 640x480 JPEG encoded images at 30 fps. The audio module is an EasyVR [41] voice recognition board. This board allows both speaker dependent and independent commands to be trained. Thus, it provides support for easy development of human-robot interaction applications.

The range and local communication module is composed of a custom standalone acquisition board (see Fig. 5), eight custom IR sensors and two Maxbotix EZ0 [39] sonar range finders. The acquisition board integrates the interface for managing both sensor types and all signal filtering and post-processing functionality. The IR sensors are mounted along the circumference of the platform (two on each side), while the sonar sensors are mounted in front. Our custom IR (see Fig. 5) sensors have emitters and receivers that are modulated at 38 kHz to avoid ambient light interference and feature two modes of operation: active and passive. In active operation the sensors perceive proximity of objects within 100 mm, while in passive operation the sensors are used for local communication that is detailed later in this section.

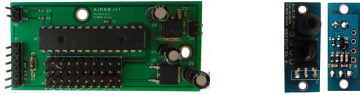


Figure 5. Range and local communication acquisition board – left; IR sensor front and back side – right.

From the perception stand point, the navigation module provides odometry sensing capabilities. These capabilities are discussed next.

4.3 Navigation

AIR's navigational module (see Fig 6) is composed of a differential drive motion control board (MCB), a two stage H-bridge and geared DC motors with magnetic quadrature encoders. The module provides a command interface for controlling the speed and position of the robot and includes capabilities such as: traveling on a straight line with a specific speed, traveling on an arc of specific length and radius at a certain speed, maintaining a certain ratio between the two motor speed, rotating in place to a specific angle and many others. The module also acts as a position sensor for the main unit. The position is maintained in odometric form (x, y, θ) and is updated at a rate of 1 ms by the control loop.

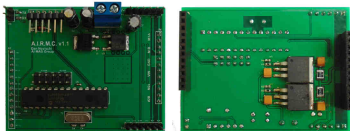


Figure 6. Motor control board: and back side.

In order to support applications where more torque is needed (e.g. collaborative box pushing) versus application where more speed is needed (e.g. collaborative target tracking) the geared motors we selected have a maximum torque of 0.91 Nm and maximum speed of 130 rpm at the gear output shaft. The motors are equipped with

low resolution magnetic encoders (only 48 counts per revolution) in order to keep the costs low for our navigation solution. Due to this choice of low resolution encoders, to accurately control the differential drive, we designed a real-time closed-loop control algorithm that runs at rate of 1 KHz. Figure 7 shows the navigation block diagram for the control and odometry sensing scheme.

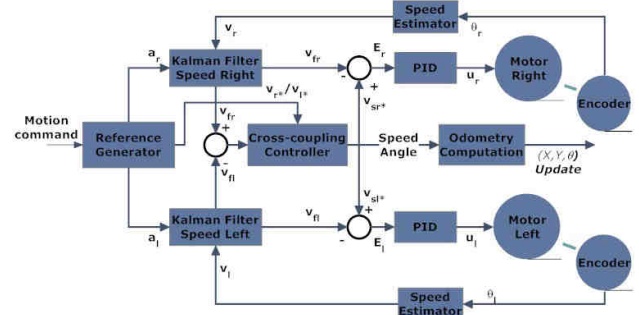


Figure 7. Navigation block diagram.

The algorithm greatly improves the resolution with which the speed is sensed through a sensor fusion scheme allowing us to smoothly control the platform. It also minimizes the effects of robot skid and slip through a cross-coupling method. Further details about our control algorithm are presented in [42].

4.4 Communication

Communication – local and global – is of special interest to swarm robotics because it is employed by a vast majority of collective robot algorithms. It is also the underling mechanism through which the behavior of the multi-robot system is monitored and analyzed. Therefore, it is critical that the AIR platform supports both local and global communication mechanisms.

Multi-robot programming and monitoring are provided through three interfaces, namely: USB, Ethernet and Wi-fi. The first two are included by our Beaglebone main unit, while the last is achieved by using an Edimax Wi-fi [43] dongle. The dongle is mounted on the primary USB port of the Beaglebone. Wi-fi communication is also used as a global communication mechanism and is made available through our middleware described in the Software subsection.

Local communication is achieved by using the IR sensors in passive mode. When in this mode, the IR's can sense the signals emitted by other sensors. This mechanism equips AIR with implicit (via sensing) communication abilities [44] and increases the robustness of our platform in face of global communication failures. Explicit communication through the IR sensors is also possible, but it is reserved for future work at the time of this writing.

4.5 Power Management

AIR's energy consumption is distributed among its modules. At full load (all modules are running, all sensors are enabled and the motors run continuously at full speed) AIR draws a maximum current of 0.697 A. In order to minimize the down time of the platform we selected a three cell lithium polymer battery with a capacity of 1.5 Ah and a 3 A high charge current rate. This enables AIR to have over 2 hours of running time under continuous full load and a charge time of only 30 minutes.

Charging a single unit is accomplished by using an off the shelf tree cell balanced charger, while charging multiple units is accomplished by using the same single unit charger and an off the shelf parallel charge adaptor. This setup enables us to easily charge up to six robots in the same amount of time that a single unit would be charged.

4.6 Software

The goal of providing comprehensive software tools in order to maximize AIR's accessibility and facilitate easy application development has been realized through our middleware. The middleware is divided in three parts: operating system, development environment and AIR's standard development kit (SDK). For the operating system, we use a custom real-time Linux 3.8 kernel with an Ubuntu 14.04 root file system which is deployed on the Beaglebone. This setup provides a wealth of software packages and libraries, such as OpenCV [46] or CMU Sphinx [47], with minimal impact on the main unit processing power. The development environment is based on Eclipse [45] with additional plugins for monitoring (remote access, image visualization) and a suite of cross-platform development tools (compilers, linkers and debuggers). This environment can be deployed on a wide range on Linux enabled machines and offers an integrated development solution for users, as well as the ability to easily deploy software on multiple targets and real-time debugging capabilities over a Wi-fi connection. Both development host and robot installation setups (either single or multiple) are managed by custom automated shell scripts. As a result, increasing the number of robots or changing the development host is completely transparent. In order to facilitate easy access to AIR's hardware a C/C++ SDK was written. The SDK is structured in five components: video, audio, range and local communication, navigation and Wi-fi. The five components are managed internally in a distributed manner by five threads of execution, where each thread acts as a client for the five components. The threads are responsible for providing up-to-date information to the main unit and checking module fault condition status. This scheme enables an agent running on AIR to immediately sense both command execution and module failures. It also enabled us to implement all of the user interface functions for each module in a non-blocking manner. As a result, processing intensive agent loops are not stalled due to hardware delays and are able to have immediate access to the most recent precepts. To illustrate, the following piece of code:

```
#include <iostream>
#include <Robot.h>
int main(int argc, char *argv[]){
    Robot::init();
    std::cout<< "x = "<<Robot::motors->getX()
    <<" , "<< "y = "<< Robot::motors->getY()
    << std::endl;
    Robot::close();
    return 0;
}
```

displays the robot relative coordinates at the console:

```
$ x = 0, y = 0
```

and represents an example of how a user would normally use the platform SDK. In this example, the navigation module is accessed through the motor object and the measured (x, y) relative coordinates are requested. The reader is correct to assume that the

getX() and getY() methods return immediately, and that any module on the AIR platform can be accessed in the same manner as the navigation module (e.g. range and local communication accessed through the sensors object). Thus, the requirements for a user to start using the AIR platform are knowledge the of C/C++ programming language and a basic description of the sensory and actuation system.

4.7 Cost

To allow manufacturing of large AIR collectives, one of our design goals was to achieve a low price for single unit manufacturing. When manufactured in single units AIR costs \$390. This cost is distributed among the parts of the composing modules. Table II shows a summary of the overall costs for these parts which include logistic costs. Dissemination of the data files (hardware and software), building instructions and the list of materials is accomplished through our group website [50].

Table 2. Parts costs

| Part | Cost |
|-------------------------------|--------------|
| Beaglebone | \$45 |
| Camera | \$45 |
| Audio | \$50 |
| Proximity and Range Sensors | \$60 |
| Geared Motors with Encoders | \$60 |
| Motor Driver | \$50 |
| Custom Printed Circuit Boards | \$40 |
| Miscellaneous Electronics | \$15 |
| Battery | \$10 |
| Mechanics | \$15 |
| Total | \$390 |

5. PLATFORM COMPARISON

A wide range of mobile robots are available both commercially and open-source. In this section we consider a subset of them that we find relevant to our own design. This subset is presented in Table III. Most of the platforms have a modular architecture, are equipped vision, range and position perception capabilities, use a differential drive for navigation and provide an interface for developing software for the robots.

Commercially available robots, such as Khepera IV, Amigo or E-Puk, feature a compact design and can be easily equipped with additional extensions. However, the prices of these robots are in range of thousands of us dollars in parts, which drastically limits there use in swarm robotics applications. In comparison, AIR, has a significantly lower price in parts, is equipped with more processing power, features a fault-tolerant distributed design and supports a integrated multi-robot development environment.

Similar – perception, processing, communication and ease of use – capabilities are provided by the WolfBot platform. While this platform has an increased autonomy provided by its larger battery and a modular design, its size is with 78% larger than AIR and all processing is done by the Beaglebone main unit. WolfBot also relies on a less accurate navigation solution than AIR that is based on an omni-directional drive and inertial measurements. Diverging from the classical single processing unit approach is ZeeRO with a clear modular and distributed design. Nonetheless, AIR has significantly more processing power, smaller size and a

Table 3. Platform comparison

| Robot | Arch. | Processing Power | Perception | Navigation | Comm. | Autonomy | Accessibility | Size | Cost in parts | Ref |
|------------|---------------------------------------|--------------------------|---|------------------------|---|------------------------|--|------------------|---------------|-------------|
| Khepera IV | Modular | ARM Cortex-A8 800 MHz | Vision, Audio, Range, Proximity, Inertial unit, Position | Differential Drive | Explicit-global via Bluetooth & Wi-fi | 5h | - Single-robot IDE with USB Bluetooth & Wi-fi support - C/C++ SDK | 14 cm Diameter | \$2000+ | [33] |
| Amigo | Non-Modular | SH2-7144 44 MHz | Range, Position | Differential Drive | Explicit-global via Wi-fi | > 2h | - Single-robot programming via Wi-fi - C/C++ SDK | 33 x 28 x 15 cm | \$2000+ | [5] [49] |
| E-Puk | Modular | 30F6014A 64 MHz | Vision, Audio, Proximity, Inertial unit, Position | Differential Drive | Explicit-global via Bluetooth | > 2h | - Single-robot programming via Bluetooth - C SDK | 7.5 cm | \$1000+ | [16] |
| JL-I | Semi-distributed | PXA255 400MHz | Vision, Position, Inertial unit | Differential Drive | Explicit-global via Wi-fi | 4h | N/A | 35 x 25 x 15 cm | N/A | [7] |
| ZeeRO | Modular & Distributed | PXA255 400MHz | Vision, Range, Proximity, Position | Differential Drive | Implicit-local via IR sensing Explicit-global via Bluetooth | N/A | - Multi-robot Server-Client Interface | 25 cm diameter | \$600+ | [4] |
| Kobot | Modular & Distributed | 16F877A 20 MHz | Opt. Vision, Proximity, Position | Differential Drive | - Implicit-local via IR sensing - Explicit-local via IR | 7.5h | - Multi-robot programming via Wi-fi | 12 cm diameter | \$650+ | [12] |
| BeBot | Modular & Distributed | ARM Cortex-A8 600 MHz | Vision, Range, Position, Inertial unit | Differential Drive | Explicit-global via Wi-fi or Bluetooth | N/A | - Single-robot IDE with RS232, Bluetooth & Wi-fi - C/C++ SDK | 9 cm sides | N/A | [19] |
| MarXbot | Modular & Distributed | ARM11 533 MHz | Omni-vision, Range scanner, Proximity, Position, Audio, Inertial unit | Differential Drive | - Implicit-local via IR sensing - Explicit-global via Wi-fi | 4h with hotswap system | - Multi-Robot IDE - Custom language SDK (ASEBA) | 17 cm diameter | N/A | [21] |
| Flockbot | Non-Modular | ARM11 700 MHz | Video, Range, Position | Differential Drive | Explicit-global via Wi-fi | 2h | - Single-robot programming via Wi-fi - C/C++ SDK | 18 cm diameter | \$500 | [30] |
| WolfBot | Modular | ARM Cortex-A8 1 GHz | Video, Audio, Range, Light, Inertial unit | Omni-directional drive | - Implicit-local via IR sensing - Explicit-global via ZigBee | 6h | - Multi-robot programming via Wi-fi - Python SDK | 17.8 cm diameter | \$550 | [28] |
| AIR | Modular, Distributed & Fault-tolerant | ARM Cortex-A8 1 GHz | Video, Audio, Range, Proximity, Position | Differential Drive | - Implicit-local via IR sensing - Explicit-global via Wi-Fi | > 2h | - Multi-robot IDE with USB, Ethernet & Wi-fi support - C/C++ SDK | 10 cm sides | \$390 | |

lower cost. On the other hand, MarXbot, the predecessor of s-bot, has more perception capabilities than AIR. It also has provides self-assembly capabilities through a custom designed gripper and increased autonomy through a hotswap system. Both s-bot and MarXbot are larger in size and have with more than 50% less processing power. A more compact design is BeBot, with 10% smaller in size than AIR. However, AIR has both explicit and implicit communication capabilities and more processing power at its disposal.

Other platforms such as the JL-I, Kobot or Flockbot, in comparison with AIR, achieve higher costs with less processing power at an increased size. AIR, also has a more accessible user interface, offloads the main unit by distributing processing tasks to composing modules and provides fault-tolerant capabilities through fault condition status checking for each of its components.

6. CONCLUSION

In this paper we presented the Ant Intelligent Robot (AIR), a miniature mobile robot platform designed for swarm robotic research and education. AIR's modular and distributed architecture is essentially different from the single processing unit approach and provides key features required by both research and educational applications: versatility of perception and communication mechanisms, accurate motion estimation and navigation, increased fault-tolerance and processing power capabilities, ease of use and multi-robot software development tools support. Furthermore, it facilitates easy assembly and maintenance while achieving a low cost.

A comparison with similar robot platforms was also presented. The comparison includes several design characteristics, namely: modularity and distributivity of the robot platform architectures, processing power, perception and communication mechanisms, autonomy, size, cost and user accessibility.

While AIR's low cost and small size enable larger collectives of robots to be built, its versatility and user accessibility allow a board range of applications to be studied.

7. REFERENCES

- [1] F. Mondada, E. Franzi, and A. Guignard. "The development of khepera." In Experiments with the Mini-Robot Khepera, Proceedings of the First International Khepera Workshop. No. LSRO-CONF-2006-060, 1999.
- [2] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J. L. Deneubourg, S. Nolfi, L. M. Gambardella and M. Dorigo, "SWARM-BOT: A new distributed robotic concept." *Autonomous Robots*, 17(2-3), 193-221, 2004.
- [3] G. Caprari and R. Siegwart. "Mobile micro-robots ready to use: Alice." *Intelligent Robots and Systems, 2005. IROS 2005 IEEE/RSJ International Conference on*. IEEE. 2005.
- [4] R. B. Rusu, R. Robotin, G. Lazea, and G. Marcu, "Towards open architectures for mobile robots: ZeeRO." In *Automation, Quality and Testing, Robotics, 2006 IEEE International Conference on*, Vol. 2, pp. 260-265. IEEE., May 2006
- [5] K. Konolige, D. Fox, C. Ortiz, A. Agno, M. Eriksen, B. Limketkai, J. Ko, B. Morisset, D. Schulz, B. Stewart, and R. Vincent, "Centibots: Very large scale distributed robotic teams," in *Experimental Robotics IX* (M. H. Ang and O. Khatib, eds.), vol. 21 of Springer Tracts in Advanced Robotics, ch. 13, pp. 131–140, Springer, 2006.
- [6] A. J. C. Sharkey, "Robots, Insects and Swarm Intelligence", *Artificial Intelligence Review*, vol. 26, pp. 255-268, 2006
- [7] H. Zhang, W. Wang, Z. Deng, G. Zong, and J. Zhang, A novel reconfigurable robot for urban search and rescue. *International Journal of Advanced Robotic Systems*,3(4), 359-366, 2006.
- [8] Yim, Mark, et al. "Modular self-reconfigurable robot systems [grand challenges of robotics]." *Robotics & Automation Magazine*, IEEE, 14.1, 43-52, 2007.
- [9] S. Fitzgerald and T. Geb, "Directed Stigmergy-Based Control for Multi-Robot Systems", *HRI'07*, March 8-11, Arlington, Virginia, USA, pp. 223-230, 2007
- [10] C. M. Cianci, X. Raemy, J. Pugh and A. Martinoli, "Communication in a swarm of miniature robots: The e-puck as an educational tool for swarm robotics." In *Swarm Robotics* (pp. 103-115). Springer Berlin Heidelberg, 2007.
- [11] A. H. Purnamadajaja, J. Iskandar and R. A. Russell, "Pheromone Communication Simulation for Mobile Robots Using Java 3D", *ICIS*, 2007.
- [12] A. E. Turgut, F. Gokce, H. Celikkanat, L. Bayindir and E. Sahin, "Kobot: A mobile robot designed specifically for swarm robotics research." Department of Computer Engineering, Middle East Technical University, Ankara, Turkey, Tech. Rep., 2007
- [13] J. McLurkin and J. Smith, "Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots." In *Distributed Autonomous Robotic Systems 6* (pp. 399-408). Springer Japan, 2007.
- [14] S. Kernbach, E. Meister, F. Schlachter, K. Jebens, M. Szymanski, J. Liedke, D. Laneri, L. Winkler, T. Schmickl, R. Thenius, P. Corradi and L. Ricotti, "Symbiotic robot organisms: REPLICATOR and SYMBRION projects," pp. 62–69, 2008.
- [15] G. G. Rigatos, "Distributed Gradient and Particle Swarm Optimization for Multi-Robot Motion Planning", *Robotica* vol.26, pp. 357-370, 2008.
- [16] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J. C. Zufferey, D. Floreano, A. Martinoli, "The e-puck, a robot designed for education in engineering." Proceedings of the 9th conference on autonomous robot systems and competitions. Vol. 1. No. 1. IPCB: Instituto Politécnico de Castelo Branco, 2009.
- [17] S. Kernbach, E. Meister, O. Scholz, R. Humza, J. Liedke, L. Ricotti, J. Jemai, J. Havlik and W. Liu, "Evolutionary robotics: The next-generation-platform for on-line and on-board artificial evolution." In *Evolutionary Computation, CEC'09. IEEE Congress on* (pp. 1079-1086). IEEE. 2009.
- [18] M. Yogeswaran and S.G. Ponnambalam, "Swarm Robotics: An Extensive Research Review", *Nature & Biologically Inspired Computing*, pp.140-145, 2009.
- [19] S. Herbrechtsmeier, U. Witkowski and R. Ulrich. "Bebot: A modular mobile miniature robot platform supporting hardware reconfiguration and multi-standard

- communication.” Progress in Robotics. Springer Berlin Heidelberg, 346-356., 2009.
- [20] A. Prorok, A. Arfire, A. Bahr, J. Farserotu, and A. Martinoli, “Indoor Navigation Research with the Khepera III Mobile Robot: An Experimental Baseline with a Case-study on Ultra-wideband Positioning.” In Proceedings of the IEEE International Conference on Indoor Positioning and Indoor Navigation, pages 1-9, 2010.
- [21] M. Bonani, V. Longchamp, S. Magnenat, P. Réturnaz, D. Burnier, G. Roulet, F. Vaussard, , H. Bleuler and F. Mondada, “The MarXbot, a miniature mobile robot opening new perspectives for the collective-robotic research.” In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, pp. 4187-4193. IEEE. 2010.
- [22] L. Wenguo and A. Winfield. “Open-hardware e-puck Linux extension board for experimental swarm robotics research.” Microprocessors and Microsystems, 35.1, 60-67, 2011.
- [23] S. Magnenat, P. Réturnaz, M. Bonani, V. Longchamp and F. Mondada, “ASEBA: a modular architecture for event-based control of complex robots. Mechatronics.” IEEE/ASME Transactions on, 16(2), 321-329. 2011
- [24] M. Rubenstein, C. Ahler, and R. Nagpal. “Kilobot: A low cost scalable robot system for collective behaviors.” Robotics and Automation (ICRA), 2012 IEEE International Conference on. IEEE, 2012.
- [25] F. Ducatelle, G.A. Di Caro, C. Pinciroli, L.M. Gambardella, “Swarmanoid: a novel concept for the study of heterogeneous robotic Swarms”, IEEE Robotics & Automation Magazine, Pages In press, 2012
- [26] K. Dantu, S. Berman, B. Kate, and R. Nagpal, “A comparison of deterministic and stochastic approaches for allocating spatially dependent tasks in micro-aerial vehicle collectives.” In IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012.
- [27] M. Brambilla, E. Ferrante, M. Birattari and M. Dorigo “Swarmrobotics: a review from the swarm engineering perspective”, Swarm Intelligence Volume 7, Issue 1 , pp 1-41, 2013
- [28] J. Betthausen, D. Benavides, J. Schornick, N. O'Hara, J. Patel, J. Cole and E. Lobaton, “WolfBot: A distributed mobile sensing platform for research and education.” In American Society for Engineering Education (ASEE Zone 1), 2014 Zone 1 Conference of the. pp. 1-8. IEEE. 2014.
- [29] A. Dirafzoon, J. Betthausen, J. Schornick, D. Benavides and E Lobaton, “Mapping of unknown environments using minimal sensing from a stochastic swarm.” In Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on (pp. 3842-3849). IEEE. 2014
- [30] ECLab Evolutionary Computation Laboratory, George Mason University, Flockbots: <http://cs.gmu.edu/~eclab/projects/robots/flockbots/pmwiki.php?n=Main.Platforms>, Jan 2015.
- [31] Raspberry pi: <http://www.raspberrypi.org>, Jan 2015.
- [32] Beaglebone: <http://beagleboard.org/>, Jan 2015.
- [33] K-TEAM Co.: <http://www.k-team.com/mobile-robotics-products>, Jan 2015.
- [34] CMU Sphinx: <http://cmusphinx.sourceforge.net/>, Jan 2015
- [35] OpenCV: <http://opencv.org/>, Jan 2015
- [36] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng. “Cooperative mobile robotics: Antecedents and directions.” Autonomous Robots, 4(1):7-27, 1997.
- [37] Y. Zhi, N. Jouandeau, and A. Cherif. "A survey and analysis of multi-robot coordination." International Journal of Advanced Robotic Systems 10, 2013.
- [38] [38] Arduino: <http://arduino.cc/>, Jan 2015
- [39] [39] Maxbotix: <http://www.maxbotix.com/>, Jan 2015
- [40] [40] Linksprite: <http://www.linksprite.com/>, Jan 2015
- [41] [41] EasyVR: <http://www.veear.eu/>, Jan 2015
- [42] [42] D. M. Novischi and A. M. Florea, “A computational approach for accurate control of differential drive mobile robots with low resolution sensors.” To be published.
- [43] Edimax Wi-fi: <http://www.edimax.com/>, Jan 2015
- [44] Y. U. Cao, A. S. Fukunaga, and A. Kahng. “Cooperative mobile robotics: Antecedents and directions.” Autonomous robots, 4(1), 7-27, 1997.
- [45] Eclipse: <https://eclipse.org/>, Jan 2015
- [46] OpenCV: <http://opencv.org/>, Jan 2015
- [47] CMU Sphinx: <http://cmusphinx.sourceforge.net/>, Jan 2015
- [48] M. S. Couceiro, C. M. Figueiredo, J. M. A. Luz, N. M. Ferreira and R. P. Rocha. “A low-cost educational platform for swarm robotics.” International Journal of Robots, Education and Art. 2011
- [49] Adept Mobilerobots: <http://www.activrobots.com/>, Jan 2015
- [50] AI-MAS Group: <http://aimas.cs.pub.ro/>, Jan 2015