*International Conference on Advanced Mechanics and Renewable Energies ICAMRE2018*
*November 28 & 29, 2018*
*Boumerdes - Algeria*

# Optimization of Energy Models and Behaviors in the Development of Mobile Robotics Applications

Touzout Walid[1], Benmoussa Yahia[1], Benazzouz Djamel[1], Eric Senn[2], and Ouelmokhtar Hand[1]

[1] Solid Mechanics and Systems
Laboratory-LMSS UMBB,
Boumerdes, Algeria
[2] LabSTICS National School for
Engineers, Lorient, France
w.touzout@univ-boumerdes.dz

**Abstract.** Mobile robots are widely used in many applications, but their energy limitation is one of the most important challenges. The goal of our research work is to enrich an infrastructure of an embedded computer with tools of modeling, monitoring and energy management in mobile robotics applications. The proposed approach must guarantee the accomplishment of critical missions by maximizing the autonomy of devices, as well as their maintenance in operational state. In this contribution, we present the state of the art of some related works by showing the general concept of energy consumption and minimization of mobile robots and some recent related works concerning energy behaviors modeling and optimization.

**Keywords:** Mobile robot; Energy behaviors; Energy optimization; Energy modeling; Operating system.

## 1 Introduction

Energy consumption of mobile robots has two points of view, electrical and mechanical each of them is classified into two types: static and dynamic power consumption. The static power consumption is the amount of energy consumed by the robot in the idle state whereas the dynamic power consumption is the amount of energy required to perform given tasks. Both static and dynamic powers can be optimized in three levels: physical, architectural and system level. The power reduction at physical level depends upon the type of the used device in a system and its electronic configurations like different ICs types (CMOS or TTLs), CPUs, types of conductors, etc [1]. At the architectural or micro-architectural level; instruction set, data-path, data-length, register structure are very important in minimizing power consumption. The energy optimization at the system level approach is to use an operating system to manage tasks and resources with energy constraints. [2]

In our work, we focused on the energy optimization at the system level where an operating system is to be used to schedule tasks and to manage dynamic and static power through several techniques. So we include in this paper some related works concerning the energy consumption of mobile robot with electrical and mechanical points of view with some examples and applications

## 2 Related Work

Many related works have been found in the literature concerning energy optimization of mobile robots. Stephen Dawson et al.[3] have presented a method called race to sleep, the idea holds that the most energy-efficient way of scheduling a computation is to put all hardware into the highest performance state then race to completion as quickly as possible, however this approach is useful to minimize the static power only, not the dynamic one. Yongguo Mei et al.[4] have proposed methods to optimize the dynamic power by determining the energy-efficient motion plan for a wheeled mobile robot, but their approach was to select a given number of motion plans and to measure the consumed energy in each scenario. Datoua et al.[5] and Shuang Liu et al.[6] have taken the same model (wheeled mobile robot), but they used the A* (A-star) algorithm to determine the path for which the robot consumes as few energy as possible. But the main drawback for these two related works is that they focus only on motion planning to reduce the amount of energy. Yongguo et al.[7] mentioned that motion is not the only power consumer so, they analyzed all energy consumers of a given robot then they built power model for each consumer (motion, sensors, electronic circuits, motors etc.) then they introduced two techniques, the Dynamic Power Management (DPM) and the Real Time Scheduling (RTS). Another work has been done by Mudass Wahab et al.[8], which was about the energy modeling of a differential robot (robot with two wheels symmetrical to each other), they built models for all energy behaviors of that robot (DC motors, Gearheads, Motion, Losses due to friction, electronics etc.) but they did not provide any optimization method to optimize one of these behaviors.

## 3 Backgrounds

Power reduction techniques may be applied at all levels of the system design hierarchy. These levels include Algorithmic, Architectural, Logic and Circuit, and Device technology. A brief description of each is given followed by some specific applications

### 3.1 Power Optimization in the Physical Level

The power reduction in this level depends on the types of devices used in a system and their electronic configurations like different ICs types (CMOS or TTLs), CPUs, types of conductors, etc. Many techniques for power reduction are available at the logic and circuit level; most of these circuits are built using CMOS device family. The sources of power dissipation in CMOS devices are represented in (Eq 1-5). [1]

$$P = P_{dynamic} + P_{static} \tag{1}$$

$$P_{static} = I_{leak} V_{DD} \tag{2}$$

$$P_{dynamic} = P_{switching} + P_{short\ circuit} \tag{3}$$

$$P_{switching} = \frac{1}{2} C.V_{DD}^{2}.f.N_{sw} \tag{4}$$

$$P_{short\ circuit} = Q_{SC}.V_{DD}.f.N_{SC} \tag{5}$$

Where:

$I_{leak}$ : Leakage current

$P_{switching}$ : Switching power (power required to charge and/or discharge circuit nodes)

$P_{short\ circuit}$ : Short Circuit power dissipation due to short circuit current

$C$ : Node capacitance.

$N_{sw}$ : Switching activities (number of gate transitions per clock cycle).

$f$ : Frequency of operation.

$V_{DD}$ : Supply voltage.

$Q_{SC}$ : Charge carried by short circuit current per transition.

The leakage power is consumed even if the circuit is idle (standby), the only way to avoid is decoupling from power. Short circuit power can be around 10% of total. Switching power is still the main source of power dissipation. By doing some math, we can easily determine the consumed energy by integrating the dissipated power with respect to time. To achieve energy reduction at this state and based on (Eq. 4), we need to do one of the following: reduce supply voltage; reduce switching activities; reduce the capacitance; reduce the number of cycles.

### 3.2 Power Optimization in the Architectural Level

At the architectural or micro-architectural level, instruction set, data-path, data-length, register structure are very important in minimizing power consumption. Power reductions in this level can be also achieved by minimizing the bus length (i.e. bus segmentation) which means: transform long heavily global bus into a partitioned set of local bus segments. Reducing the number of memory transfers and accesses is also an efficient way to reduce power consumption hence, a number of caches is needed to be adapted however; larger caches consume much power but reduce the number of memory transfers/accesses, so we need to find the right balance. [1]

### 3.3 Power Optimization in System Level

Many portable systems deploy Operating Systems (OS) to manage resources including power. The OS estimates the utilization of a device from each process, if a device is not used by any running process; the OS puts it into a low power state. When processes are properly scheduled, power reduction can be achieved without degrading performances.

The energy consumption optimization using an OS approach is divided into two parts. First, the OS kernel observes the utilization of each device, means: when the utilization is low, the device should be put in sleeping mode. Second, the OS schedules processes to reduce power without degrading the performance.[2] we will see later much details about this approach.

### 3.4 Static Energy Optimization

Fig. 1 shows a thermal image of a motherboard of a given device in the idle state; the red spots indicate that the energy is consumed even though the device is not performing any task, this energy is known as static energy, and it is well represented in Fig. 2 such that the red line indicates that about 50% of the system power is used in the idle state. The efficient way to reduce the static energy is to make all the hardware in the highest performance then race to the completion as quickly as possible then shutdown the unsed devices.

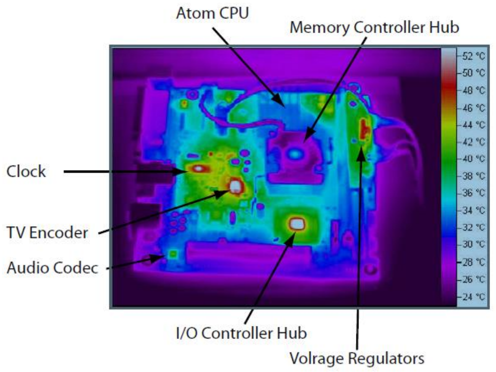

**Fig. 1**. Thermal image of an Intel Atom motherboard in idle state

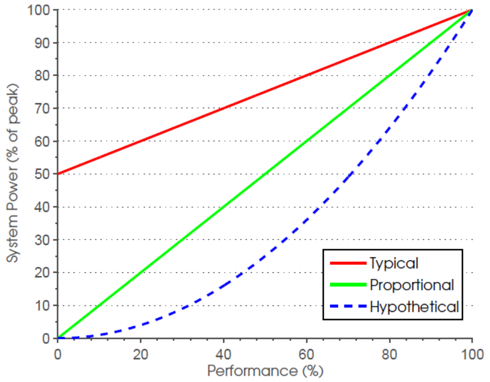

**Fig. 2.** Power-per-performance graph.

### 3.5 Dynamic Power Management

Most computers do not operate at their peak performance continuously; some devices are idle even when others are busy, this idleness provides opportunities for power reduction.

DPM puts idle devices into sleeping mode and provides the illusion that these devices are always ready to serve requests. DPM changes the power sate of a device based on the variations of workloads, the latter consists of the request generated from all process. Workloads for a disk are read/write commands; workloads for a network card are packets etc. [2]

Fig. 3 illustrates the concepts of power management. In this figure, the device is idle between t1 and t3. When the device is idle, it can enter a low-power sleeping state. Changing power states takes time; tsd and twu are the shutdown and wakeup delays.

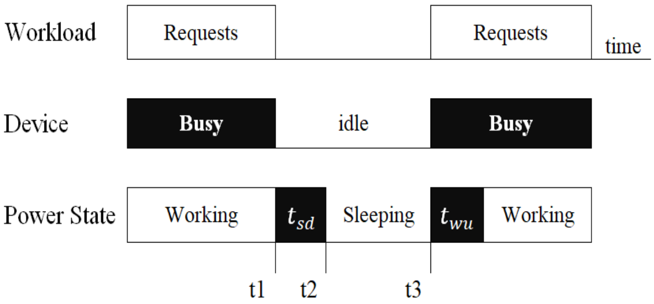

**Fig. 3.** Sleep during an idle state

As shown in Fig. 4, the transition states consumes much power, hence there is a minimum length of an idle period to save energy called Break-even time (tbe). The break-even time makes energy in both cases equal using (Eq. 6) [2]

$$P_w . t_{be} = E_0 + P_s \, (t_{be} - t_0) \tag{6}$$

Where:

Pw: Power in working mode. Ps : Power in sleeping mode. tbe: Break-even time. E0: Transition energy. t0: Transition time. The break-even time has to be larger than the transition delay and given by (Eq.7).

$$t_{be} = max(\frac{E_0 - P_s t_0}{P_w - P_s}, t_0) \tag{7}$$

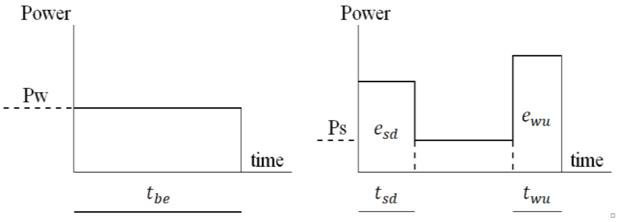

**Fig. 4.** Keeping the Device in the Working State (left). Shutting-down and Waking-up the Device (right)

### 3.6 Real time Scheduling

Running process generates requests for devices, their execution orders directly affect the arrival time of requests hence, the length of idle period. A job is defined as a unit to finish a specific task and can be scheduled to start at a specific time. Consider three processes P1, P2 and P3, and each one has three jobs Ji1, Ji2 and Ji3 so there are nine jobs, Ji1 must execute before Ji2, this called "Precedence Constraint". The precedence can be expressed as Directed Acyclic Graph (DAG) as shown in Fig. 5.

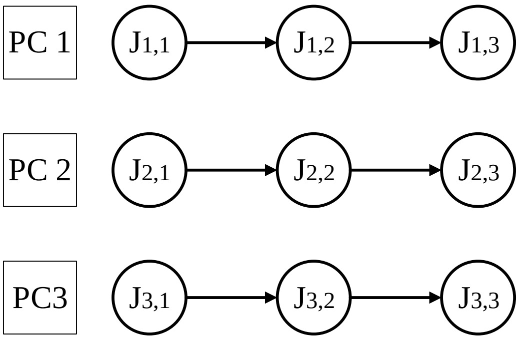

**Fig. 5.** Precedence of Three Independent Processes

Another type of constraint is the "Timing Constraint", a job has to be finished before its deadline, and the latter can be classified into three categories: firm, soft and on-time. [2] Figure 6 illustrates the difference between these three categories.

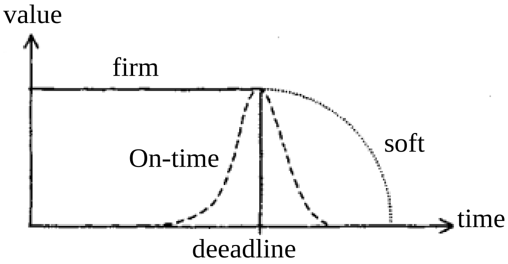

**Fig. 6.** Types of deadlines

Suppose there is a value if the job is finished before its deadline, for a firm deadline: the value drops sharply if the job finishes after its deadline; soft deadline: the value decreases more smoothly after the deadline, and if a job has an on-time deadline constraint, it should finish near the deadline, neither too early nor too late. Suppose three jobs J11, J12 and J23 need a specific device, we assume that each takes t to execute. Fig. 7 shows two schedules, the difference between them is the length of idle period. In the first one, the device is idle three times, each of length 2t. The second one, the device is idle for 6t continuously. The break-even time is between 2t and 6t hence, power management saves power only in the second schedule. [2]

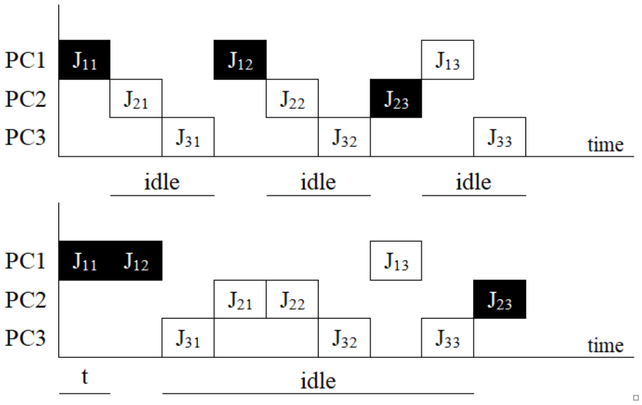

**Fig. 7.** Two schedules of three independent processes

# 4 Static and dynamic energy behaviors modeling of mobile robots

## 4.1 Energy behaviors modeling of two-wheeled robot

Two-wheeled robot also named: differential robot, a robot that has two wheels symmetrical to each other as shown in Fig. 8. In [8], a various energy loss components were investigated and a complete energy model was presented.

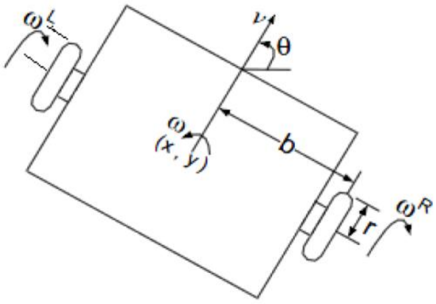

**Fig. 8.** Kinematic model of a differential robot

The energy loss equations are shown through the following:

### 4.1.1 DC motors

The DC motors are attached to the wheels of the robot and their energy loss depends on the armature resistance, friction and windage loss, stray loss and the armature currents. So the loss equations for DC motors are represented in (Eq8-9).

$$E_{dc} = E_{armature} + E_{others} \quad (8)$$

$$E_{armature} = \int ((I_a^R)^2 R_a^R + (I_a^L)^2 R_a^L) dt \quad (9)$$

$$E_{dc} = \int ((I_a^R)^2 R_a^R + (I_a^L)^2 R_a^L) dt \qquad (10)$$

### 4.1.2 Gearheads

Almost every robot uses gearhead motors to increase the torque output and decrease the velocity to a controllable value. Gearheads consume energy and their equations can be defined as shown in (Eq11-13).

$$E_{gearhead} = \int ((P_{mech}^R - \eta_g^R P_{mech}^R) + (P_{mech}^L - \eta_g^L P_{mech}^L)) dt \qquad (11)$$

$$P_{mech}^R = \tau^R \omega^R \qquad (12)$$

$$P_{mech}^L = \tau^L \omega^L \qquad (13)$$

Where:

$\eta_g^R$, $\eta_g^L$ are the efficiencies of worm gears (constant values). $P_{mech}^R$, $P_{mech}^L$ are the motors' output mechanical power which are available at the input of gearheads and they can be got by multiplying the torque $\tau$ times the rotational speed of the DC motors $\omega$.

### 4.1.3 Kinetic energy losses

A part of the available output power should be used to increase the kinetic energy and to accelerate the robot; which depends on the mass of the robot, its linear and rotational speeds $v(t)$, $\omega(t)$ and the moment of inertia of that robot I. This type of energy can be represented by (Eq. 14).

$$E_{Kinetic} = \frac{1}{2}(mv(t)^2 + I\omega(t)^2) \qquad (14)$$

### 4.1.4 Losses in electronics

All mobile robots have on-chip boards which includes DC motor drivers, sensors, embedded computers etc. The whole energy consumed by the electronic system is given by (Eq. 15).

$$E_{elect} = \int (I_{elect} V_{elect}) dt \qquad (15)$$

Where $I_{elect}$ and $V_{elect}$ are input current and voltage respectively.

### 4.1.5 Losses due to friction

The remaining output energy of the DC motors will be used to move the wheels against friction; the energy loss due to the friction is given by (Eq 16-18).

$$E_{friction} = \int (P^R_{friction} + P^L_{friction})dt \qquad (16)$$

$$P^R_{friction} = \mu mg(v(t) + b\omega(t)) \qquad (17)$$

$$P^L_{friction} = \mu mg(v(t) - b\omega(t)) \qquad (18)$$

Where:

$P^R_{friction}$, $P^L_{friction}$ are the power lost against friction for the right and left motors respectively, $\mu$ is the coefficient of rolling friction.

### 4.1.6 Overall Energy Model

After analyzing all loss components, the complete energy model equation is represented in (Eq 18).

$$E_{battery} = E_{dc} + E_{Kinetic} + E_{friction} + E_{elect} \qquad (18)$$

Fig. 9 and Fig. 10 show the amount of energy consumed by the robot and the dissipated power obtained theoretically using the above models and experimentally.

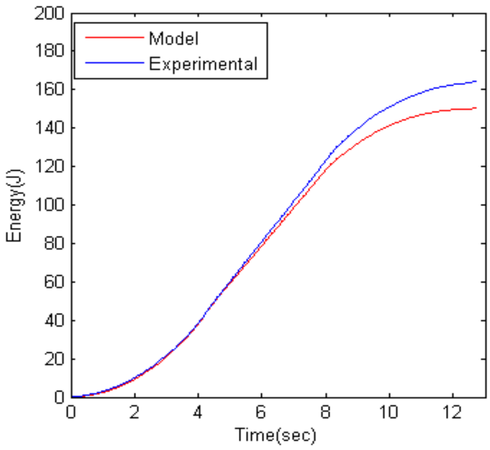

**Fig. 9.** Modeling vs. experimental result (energy)

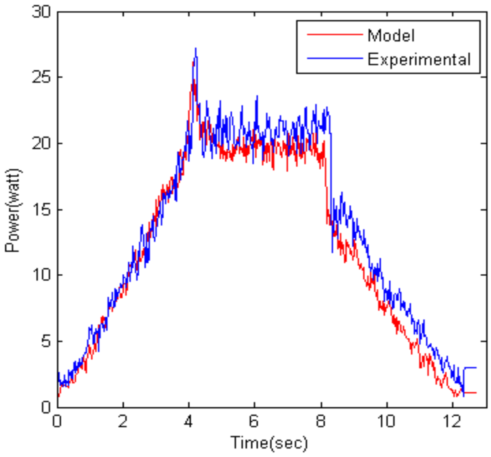

**Fig. 10.** Modeling vs. experimental result (power)

## 4.2 Energy behaviors modeling of three-wheeled robot

The energy consumption of a three-wheeled mobile robot was developed, as shown in Fig. 11, the robot composed of three wheels of the same radius r and driven by three identical DC motors. The kinematic equations relating linear and angular velocities of the robot to the linear velocities of the wheels are given in (Eq. 19):

$$\begin{bmatrix} V1 \\ V2 \\ V3 \end{bmatrix} = \begin{bmatrix} 0 & b \\ -\sin\frac{\pi}{3} & b \\ \sin\frac{\pi}{3} & b \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{19}$$

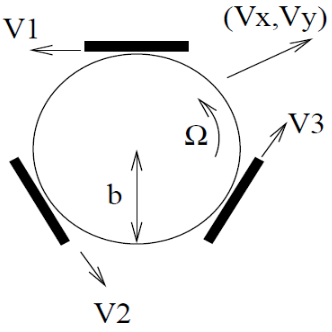

**Fig 11.** Kinematic model of 3-wheeled mobile robot

The energy consumption model of differential drive mobile robot has been developed in [8], based on [8], an energy consumption model for 3-wheeled robot can be easily derived (Eq.20-21).

$$E_T = E_k + E_F + E_e \tag{20}$$

$$\begin{aligned} E_T &= \frac{1}{2}(mv(t)^2 + I\omega(t)^2) \\ &+ \mu mg \int (|b\omega(t)| + 2\max(|b\omega(t)|, \left|\frac{\sqrt{3}}{2} v(t)\right|)dt \\ &+ \int P dt \end{aligned} \tag{21}$$

## 5 Energy-efficient motion planning for mobile robots

### 5.1 Energy-efficiency measurement for three motion plans

In [4], an approach has been presented to find the energy-efficient motion plans for a three-wheeled mobile robot shown in the previous section, after building energy behaviors model of that robot, the energy consumption was measured in different routes at different velocities. Three motion plans were suggested: scan lines, spiral and square spiral as shown in Fig. 12, to scan a given area. The energy efficiency for each scenario is illustrated in Fig. 13 which is found by dividing the area to be scanned over the required energy to scan that area, such that for small areas, best energy-efficiency is achieved when robot moves along straight lines, when the area becomes larger, spirals become the most energy-efficient plan.

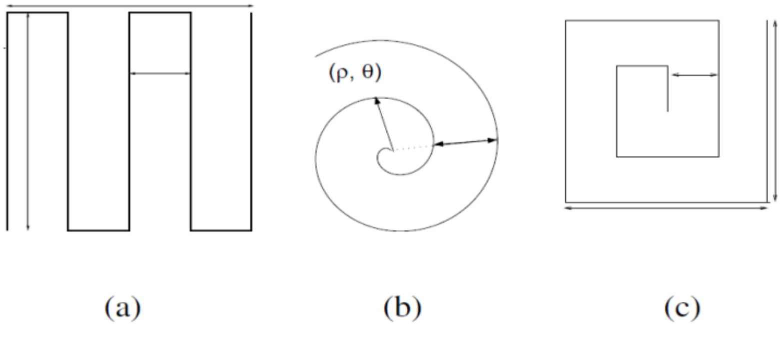

**Fig. 12.** (a) Scan lines (b) Spiral (c) Square spiral

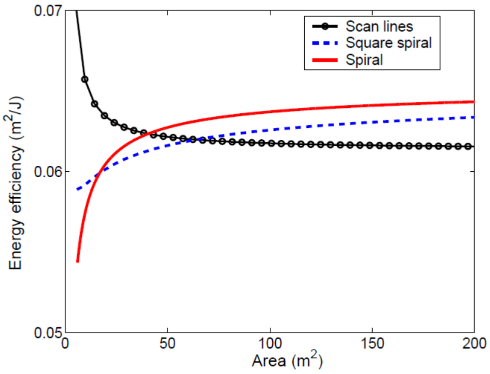

**Fig. 13.** Energy efficiency of three paths with different covering areas

### 5.2 Using A* Algorithm

As mentioned in [4-6], motion planning plays an important role to reduce the amount of energy consumed by mobile robots. So A* or A-start is an algorithm used to determine the path from a starting to a target node of a given robot for which the robot consumes as few energy as possible. This algorithm has been used in [5] to determine the optimal path with respect to the energy consumption of a three-wheeled mobile robot. An optimal path under a set of conditions (based on [5]) has been obtained which contains less number of rotations, accelerations, decelerations etc; this path is well illustrated in Fig. 14.

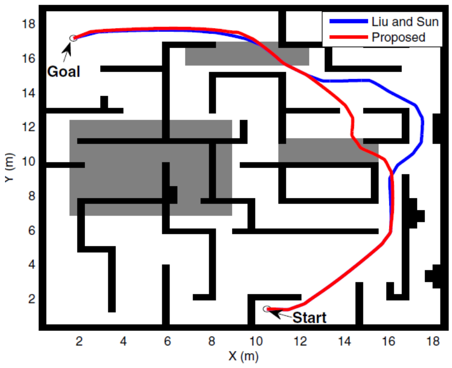

**Fig.14.** Optimal path found using A* algorithm

## 6 Conclusion

In this paper, we have given some state of the art's applications and optimization methods that are being used to reduce the static and dynamic energy consumption of mobile robots in mechanical and electrical points of view such as: motion planning, DPM, RST, race to sleep etc. We have introduced also the general concept of energy consumption and modeling of some mobile robots often used by researchers like differential robot, two- or three-wheeled robots etc.

## References

1. B. Moyer, "Low-power design for embedded processors," *IEEE,* 2001.
2. Y. H. Lu, "Power aware operating system for interactive systems," *IEEE,* 2002.
3. S. Dawson-Haggerty, "Power optimization - a reality check," 2009.
4. Y. Mei, "Energy-Efficient motion planning for mobile robots," *IEEE,* 2005.
5. R. Datouo, "Optimal motion planning for minimizing energy consumption of wheeled mobile robot," *IEEE,* 2017.
6. S. Liu, "Minimizing energy consumption of wheeled mobile robots via optimal motion planning," *IEEE,* 2014.
7. Y. Mei, "A case study of mobile robot's energy consumption and conservation techniques," *IEEE,* 2005.
8. M. Wahab, "Energy modeling of differential drive robots," *IEEE,* 2015.