

A Case Study of Mobile Robot's Energy Consumption and Conservation Techniques

Yongguo Mei, Yung-Hsiang Lu, Y. Charlie Hu, and C.S. George Lee
School of Electrical and Computer Engineering, Purdue University
{yme, yunglu, ychu, csglee}@purdue.edu

Abstract—Mobile robots are used in many applications, such as carpet cleaning, pickup and delivery, search and rescue, and entertainment. Energy limitation is one of the most important challenges for mobile robots. Most existing studies on mobile robots focus on motion planning to reduce motion power. However, motion is not the only power consumer. In this paper, we present a case study of a mobile robot called Pioneer 3DX. We analyze the energy consumers. We build power models for motion, sonar sensing and control based on experimental results. The results show that motion consume less than 50% power on average. Therefore, it is important to consider the other components in energy-efficient designs. We introduce two energy-conservation techniques: dynamic power management and real-time scheduling. We provide several examples showing how these techniques can be applied to robots. These techniques together with motion planning provide greater opportunities to achieve better energy efficiency for mobile robots. Although our study is based on a specific robot, the approach can be applied to other types of robots.

I. INTRODUCTION

Mobile robots are widely used in many applications. People can buy intelligent robotic vacuum cleaners or lawn movers from stores. Some hospitals are using robots to provide quick and safe medicine delivery [11]. Batteries are often used to provide power for mobile robots; however, they are heavy to carry and have limited energy capacity. A Honda humanoid robot can walk for only 30 minutes with a battery pack they carry on the back [3]; energy is the most important challenge for mobile robots. Rybski et al. [26] show that power consumption is one of the major issues in their robot design.

Mobile robots usually have multiple components, such as motors, sensors, microcontrollers and embedded computers. DC motors transform direct current into mechanical energy and are often used to drive the robots. Sensors collect data from environment and provide information to robots. Most often used sensors are vision, infrared,

sonar and laser rangefinders. Many robots use embedded computers for high-level computation and microcontrollers for low-level controls. This paper differentiates the microcontroller and embedded computer. The microcontroller directly controls motors, sensors, and polls the sensor readings. It hides the hardware details from the embedded computer, and provides an application programming interface (API) for the embedded computer. The embedded computer handles high-level computation, including motion planning, image processing, and scheduling. The separation of the microcontroller and embedded computer makes the designs more flexible.

Existing studies on energy reduction for robots focus on motion planning to reduce motion power. However, other components like sensing, control, communication and computation also consume significant amounts of power. It is important to consider all components to achieve better energy efficiency. This study has two major contributions. Firstly, we study power consumption of a robot called Pioneer 3DX by ActivMedia. This is a robot popular in research community [1] [10] [14]. We build power models from real measurement results for motion and sensing. We also measure the power consumption of the microcontroller. Results show that motion accounts for less than 50% of the total power consumption. Second, we introduce two techniques, dynamic power management and real time scheduling, to reduce the power consumption of mobile robots. These techniques together with motion planning provide greater opportunities for energy-efficient designs of mobile robots. Although this study is based on the data of a specific robot, the methods can be applied to other types of robots.

II. RELATED WORK

Barili et al. [4] provide a method to control the traveling speed of an autonomous mobile robot to avoid frequent speed changes and save energy. Sun et al. [29] present an algorithm for finding the energy efficient paths on terrains. We develop energy models

¹This work was supported in part by the National Science Foundation under Grant IIS-0329061.

for a mobile robot called PPRK and compare the energy consumptions of different routes [22]. Yamasaki et al. [30] present an energy-efficient walk generation algorithm for a humanoid robot. However, all of them consider only the motion power. Some studies analyze energy consumption of different components of robots. Liu et al. [19] present an energy breakdown table of a Mars rover. Michaud et al. [23] estimate the energy consumption of a rover including the communication power. However, they do not build power models for each component. Previous studies indicate that sensing, computation and communication consume significant amounts of power. In this paper, we build power models for each component of a mobile robot. This is important to develop energy-efficient designs considering multiple components together.

As the second major contribution, we introduce two energy-conservation techniques: dynamic power management (DPM) and real-time scheduling (RTS). DPM is developed primarily for portable and embedded computer systems [5] [6] [9] [15] [16] [21] [24] [25]. One of the DPM techniques is dynamic voltage scaling (DVS). It dynamically changes the voltage and clock frequency of a processor to save power. Rybski et al. [26] mention that a small robot called Scout may save about 1 W by reducing CPU's speed from 50MHz to 20MHz. RTS is to schedule multiple tasks and meet their deadlines [8]. Two often used scheduling algorithms are rate monotonic (RM) and earliest deadline first (EDF) [18]. Some studies have been devoted into the RTS of mobile robot systems. George et al. present [12] an RM scheduler for controlling autonomous robots. Laloni et al. [17] describe a scheduling strategy for mobile vehicle guidance in industrial environments. Gergeleit et al. [13] show a fault-tolerant scheduler for timely ranger data fusion of a team of robots. However, these studies do not consider the energy efficiency. Some researchers study energy-efficient RTS. Aydin et al. [2] propose a slacked EDF algorithm to minimize energy consumption of a processor with DVS capability. Sinha et al. [28] present an on-line adaptive voltage adjustment algorithm for processors. The algorithm speculatively anticipates early completions and dynamically reclaim the allocated energy. These scheduling algorithms can be further improved for robots because the timing constraints are affected by the robots' speed.

III. COMPONENTS AND POWER MODELS

Figure 1 shows a common architecture for mobile robots. This architecture includes five major components: batteries, motors, sensors, a microcontroller, and an embedded computer. The most often used energy

sources are rechargeable batteries. The batteries need to be recharged after exhaustion. In some cases, such as a Mars rover, solar powered batteries are used. Motors, sensors, microcontrollers and embedded computers are energy consumers. DC motors transform direct current to mechanical energy, and are used in robots as actuators. As robots become more sophisticated, control, sensing, communication and computation consume higher portions of energy. Robots use many kinds of sensors, such as encoders, vision, sonar, laser and infrared rangers. The microcontroller handles low-level controls, such as directly controlling motors and polling readings from sensors. At the same time, it provides a programming interface for the embedded computer. The embedded computer has better computation ability, and is in charge of high-level controls, such as motion planning and coordination.

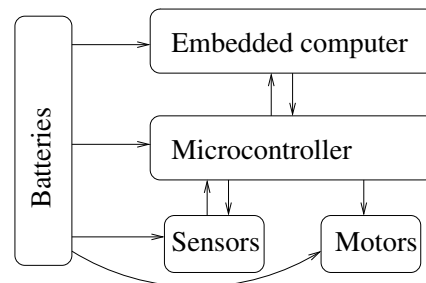


Fig. 1. A common component architecture of a robot.

A. Motion

Motors transform electrical energy into mechanical energy. The power consumption of the motors is the sum of the output mechanical power and the transforming loss. Let m be the robot's mass, and the ground friction constant is μ . When the robot travels with a speed of v and an acceleration of a , it needs a traction force of $m(a + g\mu)$. Therefore, the output mechanical power is $m(a + g\mu)v$, where g is the gravity constant. The motion power can be modeled as a function of the speed, acceleration and mass:

$$p_m(m, v, a) = p_l + m(a + g\mu)v, \quad (1)$$

where p_m is the motion power, and p_l is the transforming loss. For DC motors, the power loss is mainly due to the armature resistance and the internal mechanical friction. The power loss is relatively small when the robot moves at a high speed. At a lower speed, the power loss can be significant. From this model, if the transforming loss and acceleration can be ignored, the power is a linear function of the speed.

B. Sensing

Sensing power varies from different sensors and sensing frequencies. We can denote the sensing frequency by f_s . For video cameras, it is the number of frames per second; for laser rangefinders, it is the firing frequency. A linear function is sufficient to model the power consumption of sensors:

$$p_s(f_s) = c_{s0} + c_{s1} f_s, \quad (2)$$

where p_s is the sensing power, c_{s0} and c_{s1} are two positive constant coefficients. Their values depend on sensors used.

C. Microcontroller and Embedded Computer

The microcontroller periodically sends commands to motors and sensors, polls sensors' readings, and communicates with the embedded computer. The microcontroller's tasks are usually fixed so the power consumption of the microcontroller can be modeled by a constant.

The embedded computer is more complex than the microcontroller. Many studies have been devoted into simulation-based methods to estimate its power consumption [7] [20] [27]. The power consumption of the embedded computer may vary significantly across different programs.

IV. EXPERIMENTAL RESULTS

A. Experimental Setup

The Pioneer 3DX weighs about 9kg, and can carry at most 22.5kg of load. The robot itself is powered by lead-acid rechargeable batteries and the computer is powered by a rechargeable Li-ion battery. The robot has two DC motors driving two wheels. The maximum traveling speed is approximately 1 m/s. The DC motors are assembled with encoders. The robot has two arrays of sonar sensors, one in the front and one in the rear. Each array has 8 transducers. A Hitachi-8S microcontroller is used to control motors and sensors, and it communicates with an embedded computer through a serial port. The microcontroller is managed by a real-time operating system called AROS. We use a data acquisition (DAQ) card from National Instrument to measure the power consumption. DAQ can measure multiple channels simultaneously. Figure 2 is a picture of the Pioneer 3DX robot with a laptop on the top to control the robot. In our experiments, we use two computers: one computer to control the robot and the other computer to measure the power consumption. Both computers are placed in a cart pushed by a person to follow the robot.



Fig. 2. The Pioneer 3DX robot.

B. Motion Power

We measure the motion power of the robot in different scenarios. The robot travels at different speeds on different grounds, and move along straight lines and circles. We also change the load of the robot.

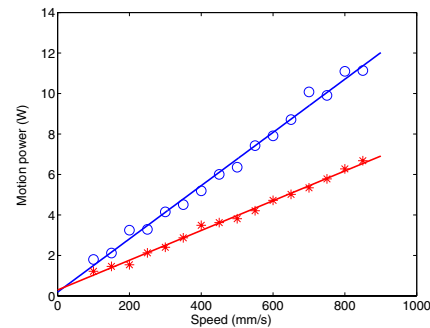


Fig. 3. Motion power at different speeds.

Figure 3 shows the power consumption of the robot runs at different speeds. The robot moves along straight lines in a hallway inside the Electrical Engineering Building of Purdue University. Inside the figure, the lower set of data and the fitting line is for the robot without load; the upper set of data and the fitting line is for robot with a 9kg load. The two energy models are:

$$p_m(v) = 0.29 + 7.4v \quad (3)$$

$$p_m(v) = 0.19 + 13.1v \quad (4)$$

Figure 4 shows the motion power when the robot travels along a circle with a radius of 0.5m. To move along a circle, we control the two wheels at different speeds. The distance between the two wheels is 33 cm. If one wheel's speed is twice of the other wheel's speed, the center of the circle is closer to the low speed wheel with a distance of 33cm so that the two wheels have the same angular velocity. The center of the robot moves

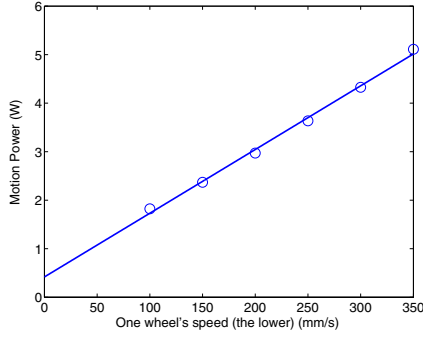


Fig. 4. Motion power at different speeds.

with a radius of $33 + \frac{33}{2} \approx 50\text{cm}$. The horizontal axis shows the speed of the slower wheel.

C. Sonar Sensors' Power

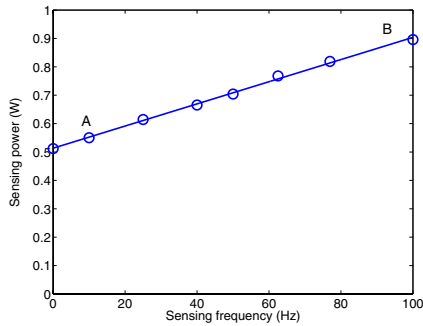


Fig. 5. Sonar sensors' power consumption.

Figure 5 shows the power consumption of the two sonar arrays at different sensing frequencies. The power consumption model is:

$$p_{sensing}(f_s) = 0.51 + 0.0039f_s. \quad (5)$$

The static power is 0.51 W, 76.9% of the total sensing power when the sensing frequency is 40 Hz. The power consumption increases as the sensing frequency increases; the power consumption at 10Hz (point A) is 38.2% lower than that at 100Hz (point B).

D. Microcontroller's and Computer's Power

The power consumption of the microcontroller is very stable at 4.6 W from our measurement. The power consumption of the embedded computer is estimated in the range between 8 W to 15 W. These values are estimated by dividing the battery capacity by the time the computer can run with a fully charged battery when running different programs.

Component	Power	Percentage
Motion	2.8W ~ 10.6W	12.1%~ 44.6%
sensing (sonar)	0.58W ~ 0.82W	1.9%~5.1%
Microcontroller	4.6W	14.8%~28.8%
Embedded Computer	8W ~ 15W	33.3%~65.3%

TABLE I
THE POWER BREAKDOWN OF PIONEER 3DX ROBOT

E. Power Breakdown

Table I is the power breakdown of the Pioneer 3DX robot. We calculate the range of power consumption and the percentage of each component. The motion power is estimated on the speed range from 0.2m/s to 0.8m/s when the robot have a 9kg load. The sensing power is estimated on the frequency range from 20Hz to 80Hz. We calculate the minimum and the maximum percentages of a component to the total power consumption. For the minimum percentage, the component's power is the minimum while the other components consume the maximum power. For the maximum percentage, the component's power is the maximum while the other components consume the minimum power. For example, the maximum percentage of the motion power is computed as follows. The maximum motion power is 10.6 W. The sum of the minimum power of the other components is $0.58 + 4.6 + 8 = 13.18W$. The percentage is $\frac{10.6}{10.6+13.18} = 44.6\%$. This analysis shows that motion accounts for at most 44.6% of the total power.

V. ENERGY-CONSERVATION TECHNIQUES

This section explains two promising techniques for power reduction of mobile robots.

A. Dynamic Power Management

Dynamic power management (DPM) dynamically adjusts power states of components adaptive to the task's need. The purpose is to reduce the power consumption without compromising system performance. Many electronic components have multiple power states; their power consumption is different at different power states. For example, processors can run on different frequencies. To save power, the processors can enter lower frequencies when the workloads are light. Another example is to shut off the power supply to the disk in an embedded computer to save the static power when there is no disk access.

A simple DPM method shuts down a component when it is idle. It is essentially a prediction problem. If we predict there is no access on this component for a reasonably long period of time, the component can be shut down to save static power. Turning on and off the

component takes time and energy. If the idle period is too short, the components may actually consume more energy for turning on and off. One of the widely used prediction methods is timeout: if the component has been idle for a time period longer than the timeout, the component will be shut down. The rationale behind timeout is that the component is likely to keep idle in the near future since it has been idle for a while.

Another widely used DPM technique is dynamic voltage scaling (DVS) by reducing both supply voltage and clock frequency to reduce the power consumption of processors. The dominant power dissipation of a CMOS circuit is its dynamic power, which can be expressed by cv_{dd}^2f , where c is the effective switched capacitance, v_{dd} is the supply voltage and f is the clock frequency.

B. Real-Time Scheduling

Real-time systems handle tasks with deadlines. Real-time scheduling (RTS) schedules multiple tasks and meet the deadlines. If the tasks can be scheduled without missing the deadlines, we say they are schedulable. Mobile robots are real-time systems. When a robot detects an obstacle, it has to timely slow down and decide the next motion. For multiple robots coordinating to accomplish a task, timely information communicating is critical. Two often used scheduling algorithms are rate monotonic (RM) and earliest deadline first (EDF). Many other algorithms are based on these two. RM is a fixed-priority algorithm, assigning a higher priority to a task with a shorter period. EDF executes the task with the earliest deadline among all ready tasks. It has been proved that EDF is optimal with respect to minimizing the maximum lateness.

Besides scheduling tasks to meet their deadlines, RTS can also schedule the tasks such that DPM can save more energy. For example, when the idle periods of a component are too short due to frequent accesses, power cannot be saved by shutting down the component. However, if we can reschedule the tasks and make the component have more long idle periods, the component may be shut down to save power.

C. Examples

In this section, we show some potential applications of DPM and RTS into energy-efficient robot designs using several examples.

1) *Shutdown of Unused Components*: Electric components consume static power in idle states. Shutting down the power supply when a component is idle can save the static power. This technique can be applied to sensors and network cards. The sonar sensors of Pioneer 3DX consumes a static power of 0.51 W; the power is

0.66 W at 40 Hz. When the robot stops, the sensors may be turned off. If half of the time the sensors can be shut down, the average sensing power can be reduced by $\frac{0.66 \times 0.5}{0.66 \times 0.5 + 0.51 \times 0.5} = 43.6\%$.

2) *Sensing Frequency Scaling*: It is intuitive that the sensing frequency should be different when robots move at different speeds. The sensing frequency needs to be higher when the speed is higher. Instead of keeping the sensing frequency that satisfies the highest speed's need, we can reduce the sensing frequency when the robot moves slowly. When the sensing frequency is 40 Hz, the sensing power is 0.66 W. If the robot moves slowly and the sensing frequency can be reduced to 20 Hz, the sensing power becomes 0.59 W. The average sensing power can be reduced by $\frac{0.66 - 0.59}{0.66} = 11\%$.

3) *Dynamic Voltage Scaling*: DVS is very effective in reducing processors' power. The processor inside the Hitachi-8s microcontroller can work at two different frequencies: 20MHz and 10MHz. The current operating system inside the microcontroller doesn't support the frequency scaling. Therefore, we can not measure the power savings. However, if we can dynamically change the working frequency according to the workload, we can reduce the control power. This technique also applies to the embedded computer.

4) *Trade-off between Motion and Communication*: A team of robots may move and cooperatively execute a task, such as exploring an unknown area. Robots need to send sensing data through wireless communication. Consider one robot needs to transfer data to another robot, but the robot is far away. If the robots can move closer, the communication power can be saved. The cost here is the motion power for moving closer. If the volume of the data is large enough, more communication power can be saved than the motion power cost.

5) *Energy-Efficient Real-Time Scheduling for Robots*: A mobile robot is a real-time system. The robot can have many periodic tasks, such as motor and sensor control, sensing data reading, motion planning, and data processing. The robot may also have some aperiodic tasks, such as obstacle avoidance and communication. RTS can work with DPM to more effectively reduce the power consumption. For example, if a scheduler can cluster tasks closer in time and create longer idle periods, shutdown techniques can be more effective. RTS also can work with DVS to reduce processor energy consumption, as we discussed in the related work. For mobile robots, the tasks' deadlines are different at different traveling speeds. At a higher speed, the periodic tasks have shorter periods. Therefore, we should consider both motion planning and RTS together.

VI. CONCLUSION

In this study, we measure the power consumption of different components of a mobile robot called Pioneer 3DX. Power models of motion and sensing have been developed. The experimental results show that motion consumes less than 50% of the total power consumption. In this paper, we introduce two techniques DPM and RTS for energy-efficient designs of mobile robots. These techniques together with motion planning provide greater opportunities for reducing the power consumption and prolonging the operation time of mobile robots. For future work, we plan to extend the current study in two directions. First, we will measure power consumption of more components, such as laser rangefinders and cameras. Second, we will implement the proposed energy conservation techniques into the Pioneer robots, and conduct experiments in real applications.

REFERENCES

- [1] D. Auguelov, D. Koller, E. Parker, and S. Thrun. Detecting and Modeling Doors with Mobile Robots. In *ICRA*, pages 3777–3784, 2004.
- [2] H. Aydin, R. Melhem, D. Mosse, and P. Mejia-Alvarez. Power-Aware Scheduling for Periodic Real-time Tasks. *IEEE Transactions on Computers*, 53(5):584–600, 2004.
- [3] R. Aylett. *Robots: Bringing Intelligent Machines To Life*. Barons, 2002.
- [4] A. Barili, M. Ceresa, and C. Parisi. Energy-Saving Motion Control for An Autonomous Mobile Robot. In *International Symposium on Industrial Electronics*, pages 674–676, 1995.
- [5] L. Benini, A. Bogliolo, and G. D. Micheli. A Survey of Design Techniques for System-Level Dynamic Power Management. *IEEE Transactions on Very Large Scale Integration Systems*, 8(3):299–316, 6 2000.
- [6] L. Benini, A. Bogliolo, G. A. Paleologo, and G. D. Micheli. Policy Optimization for Dynamic Power Management. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 18(6):813–833, 6 1999.
- [7] D. Brooks, V. Tiwari, and M. Martonosi. Watch: A Framework for Architectural-level Power Analysis and Optimizations. In *International Symposium on Computer Architecture*, pages 83–94, 2000.
- [8] G. C. Buttazzo. *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*. Kluwer, 1997.
- [9] E.-Y. Chung, L. Benini, and G. D. Micheli. Dynamic Power Management Using Adaptive Learning Tree. In *International Conference on Computer-aided Design*, pages 274–279, 1999.
- [10] D. Erickson. Non-Learning Artificial Neural Network Approach to Motion Planning for the Pioneer Robot. In *IROS*, pages 112–117, 2003.
- [11] J. Evans. Helpmate: An Autonomous Mobile Robot Courier for Hospitals. In *IROS*, pages 1695–1700, 1994.
- [12] R. George and Y. Kanayama. A Rate-Monotonic Scheduler for the Real-Time Control of Autonomous Robots. In *ICRA*, pages 2804–2809, 1996.
- [13] M. Gergeleit, L. B. Becker, and E. Nett. Robust Scheduling in Team-Robotics. In *International Parallel and Distributed Processing Symposium*, 2003.
- [14] B. P. Gerkey and M. J. Mataric. Sold!: Auction Methods for Multirobot Coordination. *IEEE Transactions on Robotics and Automation*, 18(5):758–768, 2002.
- [15] C.-H. Hwang and A. C.-H. Wu. A Predictive System Shutdown Method for Energy Saving of Event-driven Computation. *ACM Transactions on Design Automation of Electronic Systems*, 5(2):226–241, April 2000.
- [16] T. Ishihara and H. Yasuura. Voltage Scheduling Problem for Dynamically Variable Voltage Processors. In *International Symposium on Low Power Electronics and Design*, pages 197–202, 1998.
- [17] C. Laloni, R. Gutsche, and F. M. Wahl. Hard Real-Time Scheduling of Distance-Constrained Sensor Tasks within a Monitoring System for Mobile Robot Guidance. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 591–596, 1995.
- [18] C. L. Liu. Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. *Journal of the ACM*, 20(1):46–61, January 1973.
- [19] J. Liu, P. H. Chou, N. Bagherzadeh, and F. Kurdahi. Power-Aware Scheduling Under Timing Constraints for Mission-Critical Embedded Systems. In *Design Automation Conference*, pages 840–845, June 2001.
- [20] J. R. Lorch and A. J. Smith. Apple Macintosh’s Energy Consumption. *IEEE Micro*, 18(6):54–63, November 1998.
- [21] Y.-H. Lu and G. D. Micheli. Comparing System-Level Power Management Policies. *IEEE Design and Test of Computers*, 18(2):10–19, March 2001.
- [22] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. S. G. Lee. Energy-Efficient Motion Planning for Mobile Robots. In *International Conference on Robotics and Automation*, pages 4344–4349, 2004.
- [23] S. Michaud, A. Schneider, R. Bertrand, P. Lamont, R. Siegwart, M. V. Winnendael, and A. Schiele. SOLERO: Solar Powered Exploration Rover. In *the 7th ESA Workshop on Advanced Space Technologies for Robotics and Automation, The Netherlands.*, 2002.
- [24] L. S. Nielsen, C. Niessen, J. Spars, and kees van Berkel. Low-Power Operation Using Self-Timed Circuits and Adaptive Scaling of Supply Voltage. *IEEE Transactions on VLSI Systems*, 2(4):391–397, December 1994.
- [25] Q. Qiu and M. Pedram. Dynamic Power Management Based on Continuous-time Markov Decision Processes. In *Design Automation Conference*, pages 555–561, 1999.
- [26] P. E. Rybski, N. P. Papanikolopoulos, S. A. Stoeter, D. G. Krantz, K. B. Yesin, M. Gini, R. Voyles, D. F. Hougen, B. Nelson, and M. D. Erickson. Enlisting Rangers and Scouts for Reconnaissance and Surveillance. *IEEE Robotics and Automation Magazine*, 7(4):14–24, December 2000.
- [27] T. Simunic, L. Benini, and G. D. Micheli. Cycle-accurate Simulation of Energy Consumption in Embedded Systems. In *Design Automation Conference*, pages 867–872, 1999.
- [28] A. Sinha and A. P. Chandrakasan. Energy Efficient Real-Time Scheduling. In *IEEE/ACM International Conference on CAD*, pages 458–463, 2001.
- [29] Z. Sun and J. Reif. On Energy-Minimizing Paths on Terrains for A Mobile Robot. In *International Conference on Robotics and Automation*, pages 3782–3788, 2003.
- [30] F. Yamasaki, K. Hosoda, and M. Asada. An Energy Consumption Based Control for Humanoid Walking. In *IEEE/RSJ International Conference on Intelligent Robots and System*, pages 2473–2477, 2002.