Chapter 10
A Taxonomy of Evolutionary Inspired Solutions for Energy Management in Green Computing: Problems and Resolution Methods

Joanna Kołodziej, Samee U. Khan and Albert Y. Zomaya

Abstract Over the last years, the engineers, researchers, and vendors have teamed up to design and develop the intelligent models and algorithms that constrict the use of electrical energy in computing devices in the large-scale heterogeneous systems. This chapter realizes the need to present to the scientific community a current state of the art on research, current trends, and future work on evolutionary inspired solutions for green computing.

10.1 Introduction

In modern highly parametrized large-scale computing systems the quest for more powerful computational resources has enabled significant scientific discoveries. These systems are usually composed of thousands of various computing devices, data centers and numerous services with different routing and communication protocols and various local access policies. However, such advancement in the modernization and development of intelligent and scalable distributed computing systems has significantly strained the electrical energy resources, distribution, and protection systems. For all these reasons there is

Joanna Kołodziej
Department of Mathematics and Computer Science, University of Bielsko-Biała , 43-309 Bielsko-Biała, Poland
e-mail: jkolodziej@ath.bielsko.pl

Samee U. Khan
Department of Electrical and Computer Engineering, North Dakota State University, ND 58108, USA;
e-mail: samee.khan@ndsu.edu

Albert Y. Zomaya
School of Information Technologies, University of Sydney, Sydney, NSW 2006, Australia;
e-mail: albert.zomaya@sydney.edu.au

a great need to design, develop, and test devices, procedures, methodologies, and algorithms that constrict the use of electrical energy in computing devices.

This chapter approaches the surveying of the field from the perspective of evolutionary inspired solutions for energy management in energy-aware computing. The survey presented here will serve as a stepping stone for young researchers and also combines two research disciplines, namely evolutionary and "green" computing.

The remainder of the chapter is structured as follows. Sec. 10.2 presents a simple taxonomy of energy and resource management methods in large-scale heterogeneous computing systems. Evolutionary inspired optimization techniques in static and dynamic energy management are surveyed in Sec. 10.3 and Sec. 10.4. The chapter ends in Sec. 10.5 with summative analysis of reviewed evolutionary approaches, conclusions and discussion on possible further research directions.

10.2 Taxonomy of Energy Management in Future Generation Distributed Computing Systems

A lot of research projects have been done in the domain of energy aware resource management in today's large-scale computing system. Based on the taxonomy for cloud computing proposed in [11] the management methods in modern distributed computing systems can be classified into two main categories: static energy management (SEM) and dynamic energy management (DEM), as shown in Fig. 10.1.

At the hardware level of the class of the static management methods the system devices can be replaced by the low-power battery machines or nano-processors and the system workload can be effectively distributed. It allows to optimize the energy utilized for computing the applications, storage and data transfer by reducing the number of idle devices and idle periods of active processors. It is important to carefully consider the implementation of programs that are executed in the system in order to achieve a high and fast reduction in the energy usage. Even with perfectly designed hardware, poor software design can lead to significant power and energy losses. Therefore the process of compilation or code generation and the order of instructions in application source code may lead to the optimization of the energy management.

A wide class of the dynamic energy management methods is composed of the strategies for dynamic adaptation of the system performance to the current resource requirements and other parameters of the system's state. In this case the systems experience variable workloads in the working periods which allows the dynamic adjustment of power states according to current performance requirements. Similarly to static solutions, dynamic management methodologies can be distinguished by their application levels into hardware and software local categories. Hardware tools can be classified as Dynamic Performance

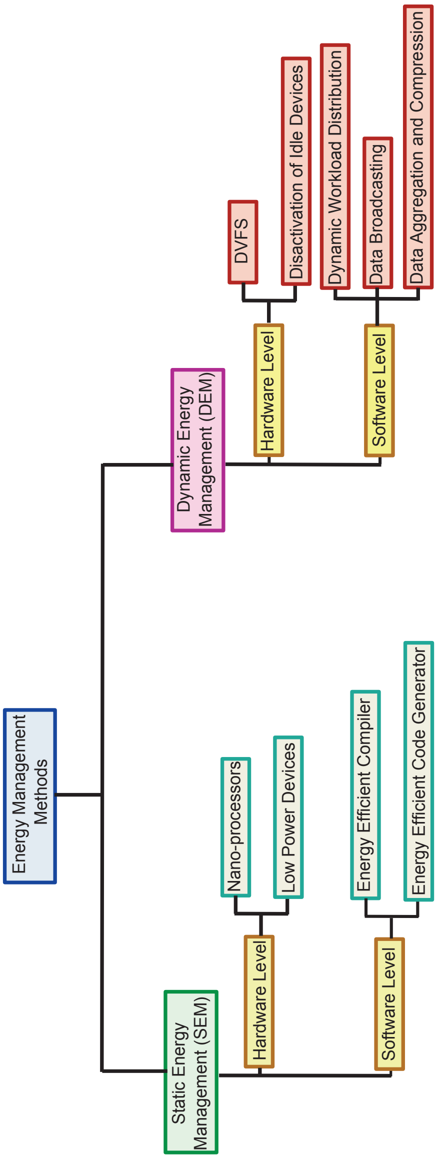

Fig. 10.1: Taxonomy of energy management in large-scale distributed computing systems

Scaling (DPS), such as Dynamic Voltage and Frequency Scaling (DVFS); and partial or complete dynamic deactivation of inactive processors. The software techniques class includes all optimization techniques connected with dynamic workload distribution, efficient data broadcasting, data aggregation and dynamic data (and memory) compression.

The fast development of global communication technologies enables in fact unlimited access of the computing system users to the highly distributed resources. However, the resource management methods in such systems may be very difficult due to the incoherent different local policies at the system's and operational levels, and high parametrization and dynamics of the whole structure. Evolutionary techniques are very effective in finding optimal or near optimal solutions in such complex dynamic environments.

Still not so large family of energy-aware genetic-based optimization methods are presented in literature. Basically classical single population strategies are used as the energy optimizers. The adaptation of such methodologies for solving the optimization problems in large-scale dynamic environments requires an application of specialized genetic operators, such as partially matching or cycle crossovers and swap or rebalancing mutation mechanisms [30], [22]. The energy consumed by the system is usually just one component of a multi-objective fitness function. In such a case the Multi-objective Genetic Algorithm (MOGA) framework [10] seems to be a key solution to tackle the complexity of the optimization process. Ant Colony Optimization (ACO) [7] and Particle Swarm Optimization (PSO) [15] algorithms are useful in generating the optimal paths and tree structures in graph-based models of networks, multi-processor machines and parallel applications. Finally, just few approaches in grid and cloud scheduling show that island, hierarchical and cellular parallel GAs can essentially speed up the optimization process and improve the exploration and exploitation of the large search space.

This chapter surveys the recent and most promising evolutionary inspired solutions to the static and dynamic energy-aware resource management in modern large-scale distributed computing systems. The presented algorithms are characterized by the following three attributes:

- type of the algorithm;
- objective function;
- application area.

10.3 Static Energy Management: Code Optimizers in Embedded Systems

Static methods of energy management in distributed systems are usually employed at the hardware level. Energy absorbing devices can be replaced by the machines with low power batteries and processors. However, this is not so good area for the implementation of the evolutionary meta-heuristics. Some recent projects are focused on the extension of the system architecture by future generation nano-processors, which successfully reduce the energy consumption in the system. The most effective architecture optimization methodologies for such systems are driven by the evolutionary inspired meta-heuristics. The high effectiveness of those techniques is demonstrated mainly in the optimization of compilers and the generators of the source codes of applications at the software level.

Fig. 10.2 presents the basic genetic-based approaches to static energy management in embedded systems.

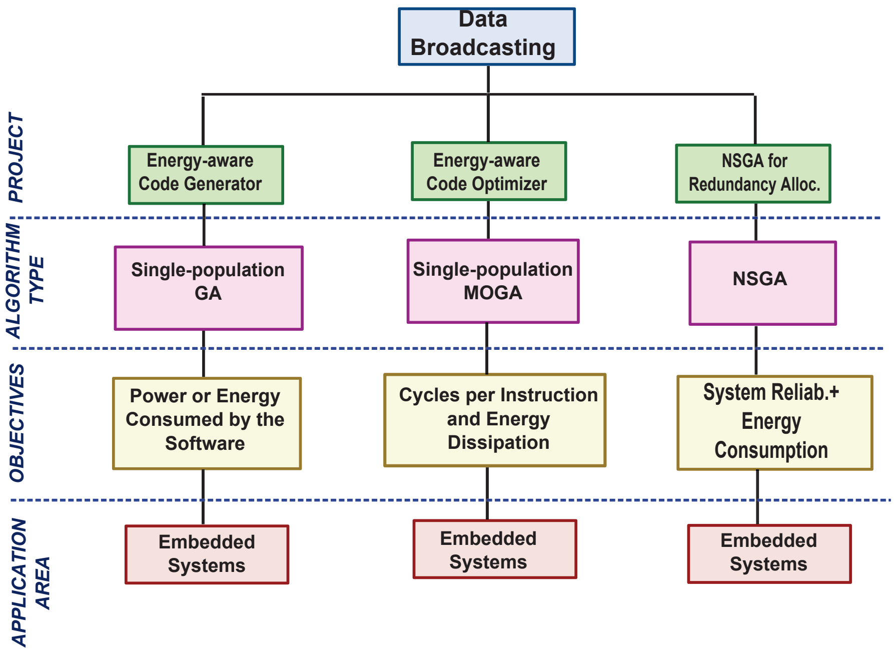

Fig. 10.2: Genetic-based methods for static resource management in embedded systems

Most of the embedded systems are composed of the digital signal processors (DSPs) that flexibly account for the modification of the system specification. However, many embedded applications are still prepared in assembly code. An implementations of such codes are time-consuming and inefficient in system energy utilization. For such reasons, there is a need for optimizing compilers and and application source codes to adapt them to the special architectures and thus to make them capable of exploiting the irregular architecture features of DSPs.

Lorenz et al. define in [24] an energy-aware code generator (GCG) based on single population genetic algorithm. This code generator reduces the energy consumption by suitable instruction selection and instruction scheduling. Energy aware compilation is done with respect to an instruction level energy cost model which is integrated into the code generator and simulator. The genetic algorithm module works on specialized chromosomes, that encode a set of basic blocks, which are created by using a simple decomposition procedure to the source program. Each such a block is represented by a node in data flow graph (DFG). Each gene of the chromosome represents an operation like a load or an addition. The values of a gene express an information about used registers, performed processor instruction, execution cycle (etc.), which are necessary to code generation. An objective function is defined as a consumed power or energy of a program. It is represented by values of average power dissipation of certain combinations of instructions. The authors used their method for SIMD instructions (SIMD=single instruction multiple data). The obtained results show the 30% of the energy reduction with 8% of reduction of the application code.

Similar GA-inspired approach to code optimization is proposed by Azzemi in [2]. The author considers the multimedia DSP processors and define an architecture-based parametric optimization of `C` source code for iterative compilation. Successive source-level, code transformations are applied in order to evaluate an application expression profile. The optimization criteria are defined by cycles per instruction and energy dissipation objectives. This bi-objective function is optimize by a simple multi-objective genetic algorithm (MOGA). The achieved energy reduction is in the range of 17%.

Optimal utilization and reliability of resources connected with the data consolidation are the key quality attributes in various types of today's complex embedded systems. The energy consumption may be reduced by replication of computational and data nodes. This problem is referred to as *redundancy allocation in embedded systems*. The main drawback of using this method may lie in additional energy requirements for supplying the replicated nodes. However, the key solution of the problem may be a trade-off analysis of resource reliability and energy consumption at an architectural level, and the employment of the redundancy allocation, which has a significant effect on both system quality attributes. In [27] Meedeniya et al. try to solve the redundancy allocation problem in the embedded systems by using the Markov Reward Model [14] for system representation. The authors optimized a bi-objective function with system reliability and energy consumption components by using the non-dominated sorting GA (NSGA). Each chromosome encodes a single redundancy allocation. Each allele in a chromosome represents a redundancy level for a system component. The achieved empirical results show that the proposed method can significantly reduce the energy consumption for a very small trade-off of reliability, which would definitely be an interesting information for the system designer.

10.4 Evolutionary Inspired Dynamic Data and Resource Management in Green Computing

Evolutionary-based solutions to dynamic energy management in large-scale distributed systems are primarily proposed as scalable and robust methodologies for scheduling and data processing in networking, cluster and grid computing. This section highlights the recent research in "evolutionary-driven" energy optimization in data management, namely data broadcasting and aggregation in wireless sensor networks, grid and cloud scheduling, where the voltage supply of the devices may be modulated in the system.

10.4.1 Energy Efficient Data Transmission

Data transmission (or data broadcasting) is, beyond the resource allocation and scheduling, a fundamental problem in large-scale data centers, intelligent networks and grid and cloud environments. The types of the implemented communication protocols have a great impact on the whole system performance. The problem of an efficient and energy-aware data broadcasting is especially essential in today's large-scale wireless networks such as ad-hoc and sensor networks, where the nodes, acting potentially both as routers and hosts, are equipped with antennas for sending and receiving information. Communication may be performed by one-to-one transmissions (single-hop) or using other nodes as relay stations (multi-hop). In both cases each sender node must adjust its emission power in order to reach the respective receiver node. Additionally, in the cases where energy is supplied by batteries, the network lifetime is limited by the batteries of the wireless devices. Therefore, energy saving is critical in all network operations.

Minimum Energy Broadcast (MEB) is defined as a problem of minimizing the energy during the data transfer. Formally, it can be formulated as the minimal spanning tree task ($T = (V, E_T)$) in the fully connected graph $G = (V, E)$ representing the system structure. The root of the tree is a source node for the data (signals) emission and the following energy emission function is minimized:

$$P(T) = \sum_{i \in V} \max_{(i,j) \in V_T} d(i, j)^{\alpha}, \tag{10.1}$$

where $d(i, j)$ is the Euclidean distance between the nodes i and j and α is a parameter that, depending on the environment, takes typically values between 2 and 4. It is assumed that the graph G for wireless networks is directed and $d(i, j)^{\alpha} < p_{max}$, where p_{max} is a maximal emission power in the system. If the antennas in the network nodes are directional, a beam width and a beam direction must be chosen for each node $i \in V$.

In classical cluster and grid systems the energy utilized for the data files transfer between two connected nodes is summarized (and then optimized) for all possible nodes pairs.

Figure 10.3 presents few recent evolutionary inspired approaches to energy aware data transfer in cluster system and wireless sensor networks.

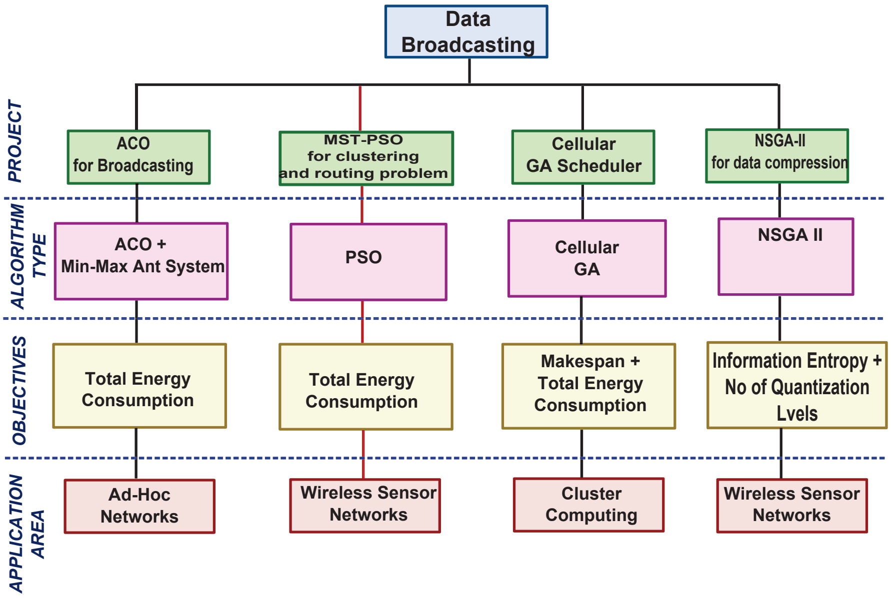

Fig. 10.3: Evolutionary solutions to energy-aware data broadcasting

In [13] Hernández, Blum and Francès address the problem of signal broadcasting in the ad-hoc networks. They consider the system with omni-directional and directional antennas. The emission energy defined in Eq. 10.1 is an objection function, which is globally optimized by using a specialized Ant Colony Optimization algorithm – Min-Max Ant System in the Hyper-Cube Framework [35]. At each iteration of the algorithm artificial ants construct a broadcasting tree rooted at the emission source node. Local search r-shrink algorithm is applied to each of these trees and the pheromone values may be updated by using also the best-so-far solutions. The power saving rates achieved in the experimental analysis is ab. 85 %, which makes the methodology spectacular solution for improving the network effectiveness in the reduction of the energy emission.

Cao et al. in [4] have considered a routing problem in wireless sensor networks (WSNs) and a case, in which a node and its cluster-head engage in a multi-hop communication. They used a particle swarm optimization (PSO) algorithm for the nodes clustering. A distance based minimum spanning tree of the weighted graph of the network is generated and the best connection between a node and its cluster-head is searched from all the optimal spanning trees on the criterion of energy consumption. Cluster-heads are elected based on the energy available to the nodes and the Euclidean distance to its neighbor node in the optimal tree. The results show that the PSO-based clustering methods ensure longer network life.

An interesting approach of cellular GA-based schedulers to cluster computing is presented by Guzek et al. in [12]. The authors consider a general scheduling problem of parallel application modeled by a directed acyclic graph (DAG) in a cluster of heterogeneous machines. The cellular algorithm is used primarily for sub-tasks clustering (the number of clusters copes with the number of processors in the machine) and scheduling. The privilege objection is makespan and the second criterion – total energy consumed during the inter-processor communication. This communication model is based on the classical delay model [32] and the energy utilized for a data transfer is measured for each CPUs connection in a parallel machine.

Dynamic data compression in the application codes seems to be a promising software tool for saving the energy used for the data propagation in wireless sensor networks. Compression methods exploit the data structure and reduce the data size. Marcelloni and Vecchio [26] perform a data compression on a network (single) node based on a differential pulse code modulation scheme with quantization of the differences between consecutive codes of the signal samples. The trade-off between a performance of compression algorithm and the amount of the lost information is determined by the set of quantization parameters. The authors employ the Non-dominated Sorting Genetic Algorithm II (NSGA–II) for optimizing the combinations of these parameters corresponding to different optimal trade-offs. The chromosomes in this approach encode quantizers defined by using the following parameters: (1)a width of the dead zone, (2) a width of the cell in the first granular subregion, (3) a number of cells in the first granular subregion, (4) a width of the cell in the second granular subregion, and (5) a number of cells in the second granular subregion.

The chromosomes are encoded as binary Gray strings. The granular regions are the regions with quantization levels. Information entropy and the number of distinct quantization levels (used in the quantizer are the optimization criteria. The evaluation analysis of the proposed method shows the 62% reduction of the energy consumed in data transmission.

10.4.2 Energy-Aware Data Aggregation in Grids, Clouds and Wireless Sensor Networks

Data aggregation is the combination of data from different datasets by using a specified aggregation function. This function can be defined as a duplicate data suppression, minima, maxima and average data indicator. A big amount of the energy in data centers is the idle power wasted when servers run at low utilization. Multiple data center applications may be hosted on a common set of servers. Also sensor nodes in wireless networks may generate significant redundant data. This allows for consolidation of application workloads on a smaller number of servers and aggregation of similar data packets from multiple network nodes that may increase the system utilization by save the energy.

In grid and cloud computing the problem of loading servers to a desired utilization level for each resource may be modeled as a multi-dimensional bin packing problem where servers are bins with each resource (CPU, disk, etc.) being one dimension of the bin. The bin size along each dimension is given by the energy optimal utilization level. Each hosted application with known resource utilizations can be treated as an object with a given size in each dimension. The ultimate goal of the consolidation algorithm is to pack all items to possible minimal number of bins. An objective function for such a problem can be defined as follows(see also [9]:

$$f = \sum_{v=0}^{n-1} y_v, \tag{10.2}$$

and is minimized subject to the following constraints:

$$\sum_{i=0}^{m-1} \bar{r}_{i,k} x_{i,v} \leq C_{v,k} y_v, \forall v \in \{0,\ldots,n-1\}, \forall k \in \mathbb{R} \tag{10.3}$$

$$\sum_{v=0}^{n-1} x_{i,v} = 1, \forall i \in \{0,\ldots,m-1\} \tag{10.4}$$

where

- n is the number of bins;
- m is the number of items;
- y_v is the bin variable which is 1 if the bin v is selected and 0 otherwise;
- $x_{i,v}$ is the allocation variable equals 1 if the item i is assigned to the bin v, and 0 otherwise;
- $C_{v,k}$ is the capacity of bin v of resource $k \in \mathbb{R}$;
- $\bar{r}_{i,k}$ is the i-th item maximum demand for resource $k \in \mathbb{R}$ over the last measurement period.

The condition 10.3 ensures that the capacity of each bin is not exceeded and constraint 10.4 guarantees that each item is assigned to at most one bin.

In wireless sensor networks signal processing methods may be used for data aggregation. In this case, it is referred to as data fusion where each node is capable of producing an accurate output signal by using some techniques such as beam forming to combine the incoming signals and reducing the noise in the output ones.

Selected genetic-based methods for data aggregation are reported in Fig. 10.4.

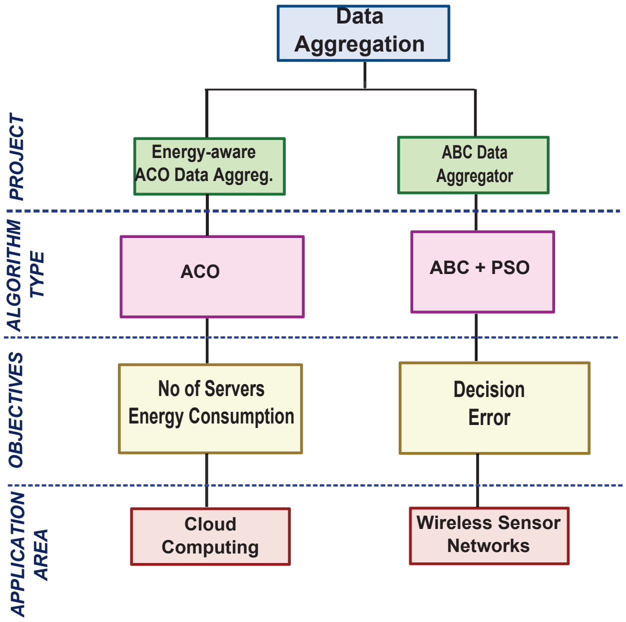

Fig. 10.4: Genetic-based methods for data aggregation in large-scale distributed systems

One of the most recent Ant Colony Optimization (ACO) approaches in data aggregation in cloud computing is presented by Feller et al. in [9]. The authors used the ACO algoriithm for the consolidation of virtual machines on the least number of physical nodes in the cloud system. The problem is interpreted as an instance of the multi-dimensional bin-packing (MDBP) problem. The fitness function is defined as a sum of boolean bin variables given by the Eq. 10.2. The authors follow the MAX-MIN Ant System (MMAS) Framework for updating the pheromone trials [35] for the ants. The power function is specified for each host in the system for estimating the energy consumed by a workload placement. This power function is defined as a linear function $P(u)$ of the host utilization u, that is to say:

$$P(u) = (P_{max}, \ldots, PPidle) \times u + P_{idle}, \tag{10.5}$$

where P_{idle} and P_{max} stand for the average power values when the system is idle and fully utilized, respectively. Computational results show the 4.1% of energy conservation on average 4.7% of hosts.

An efficient data-aggregation in wireless sensor networks can be achieved by the determination of optimal local thresholds in the decisions made by the networks fusion center for detecting the events. Each sensor node in the network collects local observations corrupted by noise and sends a summary to a fusion center, which is responsible for making the final decision. Thresholding may lead to a gain in terms of bandwidth and energy consumed by the system. Veeramachaneni et al. [37] present a hybrid of ant-based control and PSO (ABC-PSO) method for the local threshold management to achieve an optimal decision route. Partial solutions to the optimization problem are constructed by artificial ants that move from a node to another and define the paths of network nodes. Then PSO algorithm identifies the thresholds and achieves the minimum error for the sequence. A feedback on this is presented to ants to help them to improve the qualities of node sequences to achieve optimal thresholds on all nodes and an optimal decision route (hierarchy) that assure minimum energy expenditure.

10.4.3 Dynamic Voltage and Frequency Scaling in Energy-Aware Scheduling and Resource Allocation Problems

Scheduling in traditional distributed systems has been mainly studied for system performance parameters without data transmission requirements. With the emergence of data grids (DGs) and data centers, data-aware scheduling has become a major research issue. Today's data centers arise quite naturally to support needs of scientific communities to share, access, process, and manage large data collections geographically distributed.

Computing devices (CPUs) are the major energy "consumers" in a data center. The energy of the system is utilized for the tasks execution, data storage at the data hosts, data transmission, decoupling of data from processing and data replication.

Power and total energy consumption can be reduced by lowering the supply voltage of CPUs using Dynamic Voltage Scaling (DVS) or Dynamic Voltage/Frequency Scaling (DVFS) methods [25]. It is assumed that each machine in the system (it can be a data or/and computing node) is equipped with a DVS module, which allows to modulate its supply voltage and operating frequency. Instead of complete deactivation of the processor, its clock frequency along with adjustments of the supply voltage can be gradually reduced or increased in cases when the resource is not fully utilized.

The energy consumption model in a data center is usually based on the power consumption model in complementary metal-oxide semiconductor (CMOS) logic circuits. The power consumption of a CMOS-based microprocessor is defined as a sum of the capacitive, short-circuit and leakage power. The most significant factor is the capacitive power, which can be interpreted as the dynamic power consumption P_d of a CPU i in a data center and can be calculated in the following way:

$$P_d^i = A_i \cdot C_i \cdot v_i^2 \cdot f_i, \tag{10.6}$$

where A_i is the number of switches per clock cycle, C_i is an effective switched capacitance of the circuits, v_i is the supply voltage and f_i is the clock frequency. For constant values of parameters A_i and C_i for a given CPU it can be assumed that power P_d^i is proportionate to v_i^2. Since the DVS mechanism has been considered to scale up and down CPU frequency, the execution time of a task on a given machine will significantly vary according to the CPU frequency. The decrease in the execution time usually is a result of approximately proportional increase in CPU frequency. The energy consumption of the processor i can be expressed as processor power multiplied by execution time of the operation (application). In the general case the energy can be defined as follows[1]:

$$E_i = \int_0^{completion[i]} P_d^i(t)dt, \tag{10.7}$$

where $completion[i]$ is a completion time of the processor i.

It can be observed from the above equations that the decrease in voltage supply can reduce the energy per operation in a quadratic manner, and unfortunately, it may significantly slow down the completion time of the operation.

Table 10.1 shows the parameters for 16 typical DVFS levels and three main "energetic" categories for CPUs considered in the most of the publications on DVFS approaches.

Table 10.1: DVFS levels for three CPU classes

	Class I		**Class II**		**Class III**	
Level	**Volt.**	**Rel.Freq.**	**Volt.**	**Rel.Freq.**	**Volt.**	**Rel.Freq.**
0	1.5	1.0	2.2	1.0	1.75	1.0
1	1.4	0.9	1.9	0.85	1.4	0.8
2	1.3	0.8	1.6	0.65	1.2	0.6
3	1.2	0.7	1.3	0.50	1.9	0.4
4	1.1	0.6	1.0	0.35		
5	1.0	0.5				
6	0.9	0.4				

The detailed energy model description in grid and cloud data centers can be found i.e. in [17] and [11].

The DVFS technique is classified as an effective hardware dynamic energy optimizer in resource allocation and scheduling problems in large-scale

[1] The integration is replaced by the summation in the case of the discrete time process.

distributed systems. The energy-aware scheduling is usually considered as a multi-objective global optimization problem with makespan and total energy consumption as the main criteria. In most of the DVFS approached the scheduling has been defined as classical or dynamic load balancing problem. In such cases linear, dynamic and goal programming are the major optimization techniques (see i.e. [23], [38], [19], [17], [20]). Recent evolutionary-based approaches that apply DVFS to reduce energy consumption are presented in Fig. 10.5. The total energy utilization in the system is a component of the fitness functions.

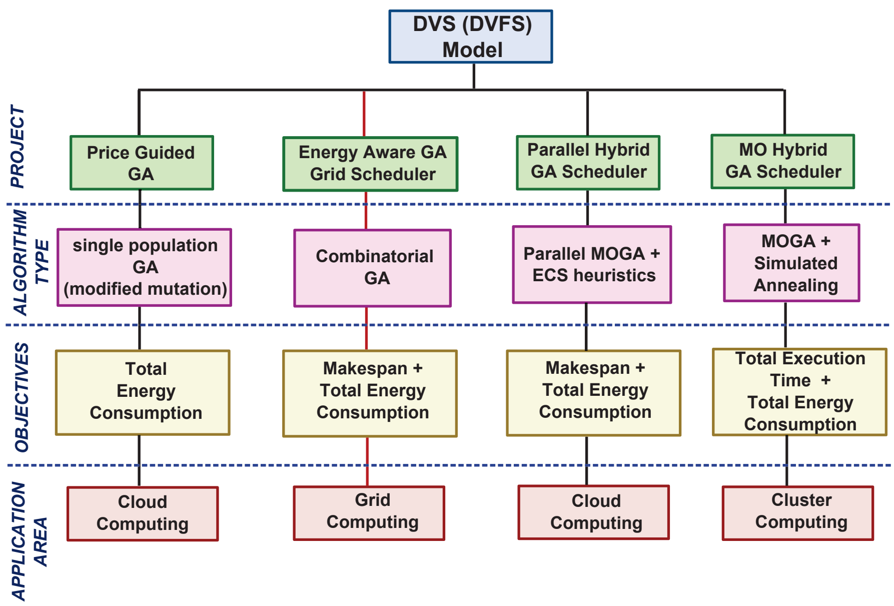

Fig. 10.5: Evo-based meta-heuristic for energy aware scheduling with modular voltage supply

In [33] and [34] Shen et al. present a *shadow price* technique for improving the genetic operations in standard GA used as a scheduler in computational cloud. The “shadow price” for a pair task-machine is defined as an average energy consumption per instruction for the processor that can operate at different voltage levels. Then the classical move and swap mutation operations are used for an optimal mapping of tasks to machines. The fitness function for such GA scheduler is expressed as a total energy consumption.

Total energy consumed by a computational grid is the key criterion in independent batch scheduling problem adressed by Kołodziej, Khan and Xhafa in [21]. The expected times of the execution of tasks on the machines in the system are estimated by using the *Expected Time to Compute* matrix model [1]. There are two GA-based schedulers developed for makespan and energy consumption optimization. The authors consider two scenarios, where all machines works at the highest voltage level and are switched to the sleep mode in idle periods, and the case of operating at different voltage levels under optimal makespan constraint. The schedulers were experimentally evaluated in static and dynamic grid environment. In both cases the modulation of the voltage supply of the machines reduced the energy consumption by 25–30 % in average.

Kessaci et al. in [16] present two versions of multi-objective parallel Genetic Algorithm (MOPGA) hybridized with energy-conscious scheduling heuristics (ECS). The GA engine is based on the concepts of island GA and multistart GA models. The authors consider parallel applications represented by a directed acyclic graph (DAG), which are mapped onto multi-processors machines. The voltage and frequencies of the processors are scaled up at 16 discrete levels and genes in GA chromosomes are defined by the task-processor labels and processor voltage. The objective function is composed of two criteria: privileged makespan and total energy consumption in the system. The reduction of the energy utilization achieved in the experimental analysis is about 47.4%.

The solution presented in [16] is dedicated to general computing and embedded systems. An application of such methodology in computational cloud is demonstrated by Mezmaz et al. in [28]. The energy conservation rate in cloud system is very similar to the results achieved in the general large-scale cluster models.

Another hybrid GA approach is presented by Miao et al. in [29]. The authors propose a multi-objective genetic algorithm which is hybridized with simulated annealing for the improvement of the local solutions for the scheduling problem in cluster computing.

10.5 Conclusions

This chapter surveyed the recent research results related to the implementations of evolutionary inspired methodologies supporting the energy and power management in modern large-scale distributed computing systems, such as wireless networks, grid, cloud and cluster computing systems. Although, the genetic meta-heuristics are still not the most popular solutions to key green computing problems, the results of the empirical analysis presented by a wide community of authors confirmed high effectiveness of these methods in fast and significant reduction (in the range of 6 % to 85%) of the energy consumed in the system.

Fig. 10.6 shows a simple summative analysis of the evolutionary approaches to energy management in today's most popular distributed computing systems.

The algorithms are classified into single- and multi-population methods. The first group contain all popular single population genetic techniques for global optimization. They may be used for solving all problems addressed in

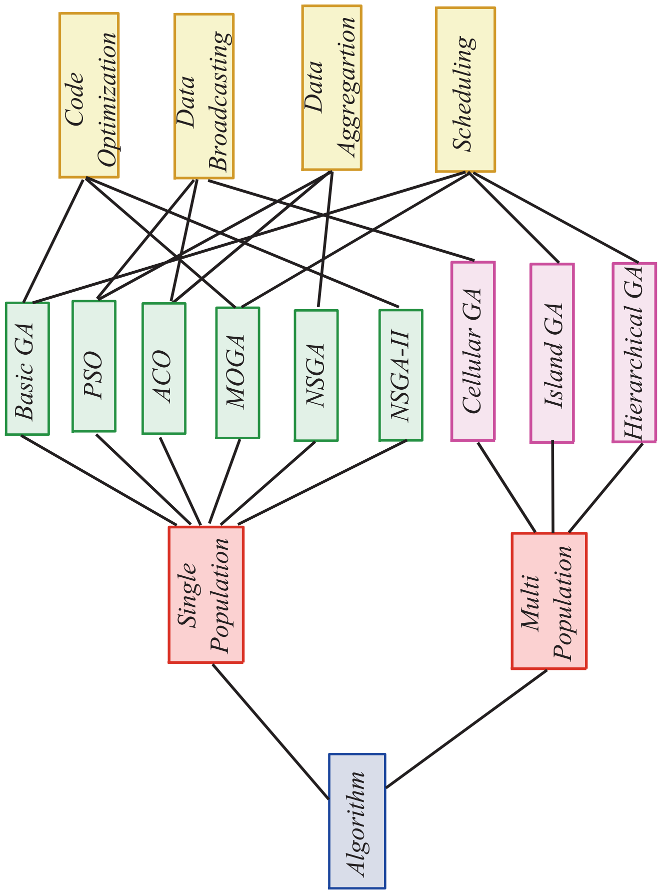

Fig. 10.6: Summative analysis of evolutionary-based approaches to energy management in distributed computing systems

this chapter and they work in all types of dynamic environments we considered in this study. The second class of multi-population algorithms is very small, that confirms an early stage of the research on such approaches.

An emergence of new generation IT systems, grow environmental concerns and imply new challenges in efficient management of huge packages of highly parameterized data. An incorporation of new additional criteria into the energy-aware data and resource management in future generation distributed systems may expose limitations in effectiveness of existing solutions at both hardware and software levels. It certainly implies a need of development of new models and meta-heuristic optimization techniques which can tackle the higher complexity of the system components, new access polices and conditions and users' preferences and requirements. A promising research direction, which can make a significant progress in green computing, may be an utilization of game-theoretical models and evolutionary-based resolution methods for supporting the decisions of the system users and resource providers. Simple cooperative games with Nash-bargained solutions have been already developed and successfully applied in energy-aware scheduling in grids and data centers (see [18], [36]), which can be a strong background for an expansion of evolutionary inspired solvers of such models.

References

1. Ali, S., Siegel, H.J., Maheswaran, M., and Hensgen, D. (2000). "Task execution time modeling for heterogeneous computing systems", *Proceedings of Heterogeneous Computing Workshop*, pp. 185–199.
2. Azzemi, N.Z. (2006). "A Multiobjective Evolutionary Approach for Constrained Joint Source Code Optimization", in Proc. of *ISCA 19th International Conference on Com- puter Application in Industry (CAINE 2006), Las Vegas, Nevada, USA*, pp. 175–180.
3. Beloglazov, A., Buyya, R., Lee, Y.C., and Zomaya, A.Y. (2011). "A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems," *Advances in Computers*, 82, pp. 47–111.
4. Cao, X., Zhang, H., Shi, J., and Cui, G. (2008). "Cluster heads election analysis for multi-hop wireless sensor networks based on weighted graph and particle swarm optimization", in Proc. of the *4th International Conference on Natural Computation (ICNC)*, vol. 7, pp. 599—603.
5. Chandran, J.J.G., and Victor, S.P. (2010). " Optimized Energy Efficient Localization Technique in Mobile Sensor Networks", *IACSIT International Journal of Engineering and Technology*, Vol.2, No.2, pp. 149–156.
6. Diaz, C.O., Guzek, M., Pecero, J.E., Danoy, G., Bouvry, P., and Khan, S.U. (2001). "Energy-aware Fast Scheduling Heuristics in Heterogeneous Computing Systems", in Proc. of *ACM/IEEE/IFIP International Conference on High Performance Computing and Simulation (HPCS), Istanbul, Turkey, July 2011*, pp. .
7. Dorigo, M., Stützle, T. (2004). *Ant Colony Optimization*, MIT Press, Cambridge (2004).
8. Fard, G.H.E., and Monsefi, R. (2010). "A Fast Multi-objective Genetic Algorithm based Approach for Energy Efficient QoS-Routing in Two-tiered Wireless Multimedia Sensor Networks", *Modern Applied Science*, Vol. 4, No. 6., pp. 101–112.
9. Feller, E., Rilling, L., and Morin, C. (2011). "Energy-Aware Ant Colony Based Workload Placement in Clouds", INRIA Report RR-7622, Rennes, France.
10. Fonseca, C. M., and Fleming, P. J. (1995). "An overview of evolutionary algorithms in multiobjective optimization", *Evolutionary Computation*, 3(1), pp. 1–16.
11. Garg, S.K., Yeo, C.S., Anandasivam, A., and Buyya, R. (2009). "Energy-Efficient Scheduling of HPC Applications in Cloud Computing Environments", CoRR abs/0909.1146.
12. Guzek, K., Pecero, J.E., Dorrosoro, B., Bouvry, P., and Khan, S.U. (2010). "A Cellular Genetic Algorithm for Scheduling Applications and Energy-aware Communication Optimization," in *ACM/IEEE/IFIP International Conference on High Performance Computing and Simulation (HPCS)*, Caen, France, pp. 241–248.
13. Hernández, H., Blum., C., and Francès, G. (2008). "Ant Colony Optimization for Energy-Efficient Broadcasting in Ad-Hoc Networks", in Proc. of the *ANTS 2008*, M. Dorigo et al. (Eds.) , LNCS 5217, pp. 25—36.

14. Katoen, J.P., Khattri, M., and S.Zapreev, I. (2005). "A Markov reward model checker", in Proc. of the Q*EST:International Conference on the Quantitative Evaluation of Systems*, IEEE Computer Society, pp. 243–244.
15. Kennedy, J., and Eberhart, R. (1995). "Particle swarm optimization", in Proc. of the *IEEE International Conference on Neural Networks*, vol. 4, 27 Nov.–1 Dec. 1995, pp. 1942—1948.
16. Kessaci, Y., Mezmaz, M., Melab, N., Talbi, E-G, and Tuyttens, D. (2011). "Parallel Evolutionary Algorithms for Energy Aware Scheduling", in *Intelligent Decisions Systems in Large-Scale Distributed Environments"*, P.Bouvry, H. Gonzalez-Velez and J. Kołodziej (Eds), "Studies in Computational Intelligence" Series, Springer Vlg, vol. 362, Chapter 4, pp. 75–100.
17. Khan, S.U. (2009). "A Goal Programming Approach for the Joint Optimization of Energy Consumption and Response Time in Computational Grids", in Proc. of the *28th IEEE International Performance Computing and Communications Conference (IPCCC), Phoenix, AZ, USA*, pp. 410-417.
18. Khan, S.U., and Ahmad, I. (2009). "A Cooperative Game Theoretical Technique for Joint Optimization of Energy Consumption and Response Time in Computational Grids", *IEEE Transactions on Parallel and Distributed Systems*, 20(3), pp. 346–360.
19. Khan, S. U. (2009). "A Self-adaptive Weighted Sum Technique for the Joint Optimization of Performance and Power Consumption in Data Centers", in Proc. of the *22nd International Conference on Parallel and Distributed Computing and Communication Systems (PDCCS)*, Louisville, KY, USA, September 2009, pp. 13–18.
20. Kliazovich, D., Bouvry, P., and Khan, S. U. (2010). "DENS: Data Center Energy-Efficient Network-Aware Scheduling," in Proc. of *ACM/IEEE International Conference on Green Computing and Communications (GreenCom)*, Hangzhou, China, December 2010, pp. 69–75.
21. Kołodziej, J., Khan, S.U., and Xhafa, F. (2011). "Genetic Algorithms for Energy-aware Scheduling in Computational Grids", in Proc. of the 6th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC2011), 26-28 October, 2011, Barcelona, Spain (article in perss).
22. Kołodziej, J., and Xhafa, F. (2011). "Enhancing the genetic-based scheduling in computational Grids by a structured hierarchical population", *Future Generation Computer Systems*, vol. 27, pp. 1035–1046, DOI:10.1016/j.future.2011.04.011, 2011.
23. Lee, Y.C., and Zomaya, A.Y. (2009). "Minimizing Energy Consumption for Precedence-Constrained Applications Using Dynamic Voltage Scaling", in Proc. of the *9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid)*, Shanghai, China, pp. 92–99.
24. Lorenz, M., Wehmeyer, L., and Dräger T. (2002). "Energy aware Compilation for DSPs with SIMD instructions", in Proc. of *Languages, compilers and tools for embedded systems: software and compilers for embedded systems LCTES/SCOPES '02*, pp. 94—101.
25. Lorch, J.R., and Smith, A.J. (2001). "Improving dynamic voltage scaling algorithms with pace", In *2001 ACM SIGMETRICS international conference on measurement and modeling of computer systems*, pp. 50—61.
26. Marcelloni, F., and Vecchio, M. (2010). "Enabling energy-efficient and lossy-aware data compression in wireless sensor networks by multi-objective evolutionary optimization", *Information Sciences*, vol. 180, pp. 1924—1941.
27. Meedeniya, I., Buhnova, B., Aleti, A., and Grunske, L. (2010). "Architecture-Driven Reliability and Energy Optimization for Complex Embedded Systems", in Proc. of the *International Conference on the Quality of Software Architectures (QoSA'10),* Springer Verlag, ISBN 978-3-642-13820-1, pp. 52–67.

28. Mezmaz, M., Melab, N., Kessaci, Y., Lee, Y.C., Talbi, E.-G., Zomaya, A.Y., and Tuyttens, D (2011). "A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems", *J. Parallel Distrib. Comput. (in press)*, doi:10.1016/j.jpdc.2011.04.007.
29. Miao, L., Qi, Y., Hou, D., Dai, Y.H., and Shi, Y. (2008). "A multi-objective hybrid genetic algorithm for energy saving task scheduling in CMP system", in Proc. of *IEEE Intl. Conf. on Systems, Man and Cybernetics (ICSMC2008)*, pp. 197—201, doi:10.1109/ICSMC.2008.4811274
30. Michalewicz, Z. (1992). *"Genetic Algorithms + Data Structures = Evolution Programs"*, Springer, 1992.
31. Min, R., Furrer, T., and Chandrakasan, A. (2000). "Dynamic voltage scaling techniques for distributed microsensor networks', In *Proc. IEEE Workshop on VLSI*, pp. 43—46.
32. Rayward-Smith, V.J. (1987). "UET scheduling with unit interprocessor communication delays", *Discrete Applied Mathematics*, Vol. 18, No. 1, pp. 55—71.
33. Shen, G., and Zhang, Y.Q. (2011). "A New Evolutionary Algorithm Using Shadow Price Guided Operators", *Applied Soft Computing*, 11(2), pp. 1983—1992.
34. Shen, G., and Zhang, Y.Q. (2011). "A Shadow Price Guided Genetic Algorithm for Energy Aware Task Scheduling on Cloud Computers", in Proc. of the *3rd International Conference on Swarm Intelligence - (ICSI-2011)*, pp. 522-529.
35. Stützle, T., and Hoos, H. (1996). "Improvements on ant-system: Introducing max-min ant system", in Proc. of the *Artificial Neural Networks and Genetic Algorithms Conference*, Wien, 1996, Springer Verlag, pp. 245—249.
36. Subrata, R., Zomaya, A. Y., and Landfeldt, B. (2010). "Cooperative power-aware scheduling in grid computing environments", *J. Parallel Distrib. Comput.*, vol. 70, pp. 84—91.
37. Veeramachaneni, K., and Osadciw, L.A. (2008). "Swarm intelligence based optimization and control of decentralized serial sensor networks", in Proc. of the *IEEE Swarm Intelligence Symposium*, 2008, pp. 1—8.
38. Zomaya, A.Y. (2009). "Energy-Aware Scheduling and Resource Allocation for Large-Scale Distributed Systems", in Proc. of the *11th IEEE International Conference on High Performance Computing and Communications (HPCC)*, Seoul, Korea, pp. .