Multiobjective generation scheduling using multicore processing-based continuous genetic algorithm

Kiran Babu Vakkapatla and Srinivasa Varma Pinni

Cogent Engineering (2020), 7: 1767019
Multiobjective generation scheduling using multicore processing-based continuous genetic algorithm

Kiran Babu Vakkapatla* and Srinivasa Varma Pinni†

Abstract: Algorithms used for day ahead generation scheduling are crucial for a power system operator to balance conflicting objectives and the network constraints. Practically feasible algorithms using parallel computing, low-cost hardware and open-source software in power system parlance are rarely attempted in the literature. In this paper, a multicore processing-based genetic algorithm is proposed for finding the optimum solution of economic emission dispatch considering the reliability indices. Continuous genetic algorithm is used to improve the speed of the algorithm. Cost minimization and emission minimization are considered as the objectives to find the set of pareto-optimal solutions. The final solution is selected from the pareto-optimal set based on the reliability of generating stations. The insight used to improve the search space is the usage of two cores of a dual core processor in parallel, with different parameters of genetic algorithm. The constraints are handled using repair function and penalty factors, based on the feasibility of implementation. The algorithm is tested on IEEE 30 Bus, 6 generator system and IEEE 57 Bus system. The results show that the multicore processing using different parameters of genetic algorithm has improved the performance.

Keywords: generation scheduling; multiobjective optimization; genetic algorithm; power system reliability; parallel computing

ABOUT THE AUTHOR
Kiran Babu Vakkapatla received M.Tech degree in electrical power engineering from JNTUH College of Engineering, India in 2012. He has diverse experience in software design and testing in the context of electric power systems. He is currently pursuing doctoral studies at Koneru Lakshmaiah Education Foundation (KLEF), India. The power systems research group at KLEF works actively in the areas of optimization, renewable energy integration and electric vehicles.

PUBLIC INTEREST STATEMENT
The continuity of electric power supply has become a necessity in these days of industrialization. The depletion of fossil fuels and increase in the penetration of wind and solar power has posed challenges to power system operator. To use the existing resources economically and with the emission within limits, research on the optimization of generation scheduling has emerged. In this paper, a parallel computing-based multi-objective generation scheduling algorithm is proposed. This algorithm has been implemented using low-cost hardware and open-source software. These features make it suitable to use for day ahead scheduling by the power system operator. The principle used to improve the performance of genetic algorithm can be used in other nonlinear optimization problems also.
1. Introduction
The depletion of fossil fuels and concern for environment have resulted in the search for alternatives and optimum utilization of resources for electric power generation. DC Optimal Power Flow (DCOPF) is used to find the optimal schedule of generation for day ahead scheduling (Wood et al., 2013). Penalty factor-based methods are used to convert two or more conflicting objectives to a single objective (Basu, 2018). Another way of handling conflicting objectives is, to find the pareto-optimal front and the best solution is found from the set of pareto-optimal solutions (Deb, 2001).

Genetic algorithms are effective in finding optimum solution of a nonlinear objective function (Ahmed & ALhialy, 2019). Metaheuristic methods are used to solve the economic emission dispatch (EED) problem due to the nonlinear nature of objectives and constraints. The performance of these algorithms depends on the control parameters of the metaheuristic method (Zaman et al., 2016). Non Dominated Sorting Genetic Algorithm -II (NSGA-II) is used to solve the dynamic EED problem in (Basu, 2008). The final solution is taken from the first pareto front after the fixed number of iterations.

The increase in penetration of renewables has an impact on the reliable operation of power system due to the inherent uncertainty of renewable energy. To improve the reliability of power system operation, reliability indices are also considered in the algorithms used for unit commitment and generation scheduling (Alsac & Stott, 1974; Vakkapatla & Pinni, 2019).

Parallel processing and multithreaded applications have shown improvement in performance when compared with a program running as a single process (Hadizadeh & Tanghatari, 2017; Razian & MahvashMohammadi., 2017). High-performance computing using supercomputers or a cluster of connected computers for grid operations was reported in literature (Epperly et al., 2012; Tippayachai et al., 2002). Security constrained unit commitment using Gurobi optimizer and 28,800 Intel Xeon Phi cores was investigated in (Gong et al., 2019). The advancement in computation comes with higher cost of hardware, difficulty of replacement in case of hardware failure and possibility of platform dependence of code.

Parallel processing using low-cost hardware and open-source software in the context of multi-objective generation scheduling has been rarely attempted. This paper fills the gap partially by using a dual core processor and different parameters of genetic algorithm to find the optimum solution of multiobjective generation scheduling.

This paper presents a multicore-based continuous genetic algorithm for finding the optimum solution of conflicting objectives. The overall pareto-optimal set is found by combining the two sets obtained from individual cores of a dual core central processing unit (CPU). As the search proceeds on two cores of the processor in different directions, improvement in the pareto-optimal solutions is observed with no additional computation time and cost.

The remaining paper is organized as follows. The problem formulation is explained in Section 2. Mathematical representation of the objectives is discussed in Section 2.1. The equality and inequality constraints that need to be satisfied are explained in Section 2.2. Flow chart and details of the algorithm are presented in Section 3. The simulation results are tabulated in Section 4 along with the interpretation. The conclusion and future scope of the proposed algorithm are presented in Section 5.

2. Problem formulation
The nonlinear objectives that need to be minimized are the cost of generation and emission from the thermal power plants.

2.1. Objectives
The minimization of generation cost (GC) and emission minimization are considered as primary objectives and their mathematical representation is presented as follows.
2.1.1. Generation cost minimization
The GC of a thermal power plant by considering the valve point loading effect (Basu, 2008) is represented by
\[
GC = \sum_{k=1}^{n} a_k P_k + b_k P_k + c_k P_k^2 + d_k \sin(e_k (P_k^{\text{min}} - P_k))
\] (1)

2.1.2. Emission minimization
The emission of atmospheric pollutants (Basu, 2008) from a thermal power plant is represented by
\[
\text{Emission} = \sum_{k=1}^{n} \alpha_k + \beta_k P_k + \gamma_k P_k^2 + \eta_k \exp(\delta_k P_k)
\] (2)

2.1.3. Secondary objective
The secondary objective is used to filter the pareto-optimal set of solutions obtained by simultaneous optimization of GC and emission. Aggregate Forced Outage Rate (AFOR) (Vakkapatla & Pinni, 2019) is used to find the most reliable solution from the pareto-optimal set of solutions. It is defined as
\[
\text{AFOR} = \frac{\sum_{k=1}^{n} F_k P_k}{\sum_{k=1}^{n} P_k}
\] (3)

Of the pareto-optimal solutions available, the one with lowest AFOR is chosen as the most reliable solution of the multiobjective optimization.

2.2. Constraints
The different equality and inequality constraints that need to be satisfied by the decision variables and line flows are presented as follows.

2.2.1. Generation and load balance
The power generated by the generators at any moment should be equal to the sum of power consumed by the consumers and the power dissipated as losses. This is represented by the equality constraint as shown in the following equation:
\[
\sum_{i=1}^{n} P_{ik} = \sum_{j=1}^{m} P_{jk} + \text{losses}
\] (4)

As the loss compensation can be handled using ancillary services (Shahidehpour et al., 2003), they are not considered in this case.

2.2.2. Generation limits
The minimum and maximum limits of active power generation are expressed as
\[
P_{k}^{\text{min}} \leq P_k \leq P_{k}^{\text{max}}
\] (5)
The minimum limit is based on the flame stability and the maximum limit is based on the design of generator.

2.2.3. Line flow limits
The maximum limit of MVA flow in a branch is represented as
\[
S_{M}^{\text{max}} \leq S_M
\] (6)
The maximum power that can flow through a transmission line is limited by the design specification of the transmission line.
3. Multiobjective generation scheduling using parallel continuous genetic algorithm
The cost of generation and emission due to generating plants are considered for simultaneous minimization of both objectives. As both the objectives are conflicting, a set of pareto-optimal solutions are obtained. From the set of solutions that fall within the limits of cost and emission, the most reliable solution is obtained by using the reliability indices of the generating stations. The initial population of real powers is generated using a random number generator and the objective functions are evaluated using them. CGA is preferred to binary coded genetic algorithm, as it improves speed of computation (Haupt & Haupt, 2004).

The flow chart of the algorithm for dual core processor is presented in Figure 1. The flow chart of the algorithm that is scheduled on each core of the processor is presented in Figure 2. The parameters of continuous genetic algorithm for two cores of CPU are shown in Table 1.

Two different ways of obtaining pareto-optimal sets of solutions are shown in Figures 3 and 4. In the case of Figure 3, it is clear that the curve AB is the final solution set because all the points on that curve are nearer to the axes than that of curve CD. If the curves obtained are as in Figure 4, the segments AE and ED form the final solution set. This is because the other points are dominated by the points on the segments AE and ED. This case can be extended to the curves with multiple intersection points.

After finding the overall pareto-optimal set of solutions, the one with least AFOR is selected as the final solution. The solution selected in this manner has the power scheduled according to the reliability of generating stations.

4. Results and discussion
The proposed algorithm is validated on IEEE 30 Bus, 6 Generator system (Alsac & Stott, 1974) and IEEE 57 Bus system (Zimmerman et al., 2011). The improvement in results due to multicore
processing is shown graphically. The software used are Ubuntu GNU/Linux, GNU Octave (Eaton et al., 2018) and the CPU is AMD E2 processor. The results of two case studies are presented as follows.

4.1. Case-1: multiobjective generation scheduling on IEEE 30 Bus, 6 generator system
The plot in Figure 5 shows the points obtained from core 1 and core 2 in red color (curve 1) and blue color (curve 2), respectively. In the left half of the plot, it can be observed that curve 1 is dominating and in the right half, curve 2 is dominating. The final pareto-optimal front is shown in

Figure 2. Flow chart of multi-objective generation scheduling.

| Table 1. Parameters of genetic algorithm |
|----------------------------------------|
| Parameter                | Core 1 | Core 2 |
| Population               | 500    | 500    |
| Number of generations    | 10     | 10     |
| Crossover rate           | 0.65   | 0.7    |
| Mutation rate            | 0.1    | 0.1    |
This clearly shows that parallel processing using both cores has improved the solutions in the final pareto-optimal front.

The results of single objective and multiobjective optimization with different crossover rates are presented in Tables 2 and 3. It is observed that in the case of single objective like cost minimization, crossover rate of 0.65 is better with an optimum value of 21,198 $/hr. In the case of emission minimization, crossover rate of 0.7 is better with the minimum value of 1642 $/hr. This shows that search proceeds in different directions in the search space with different control parameters of the genetic algorithm.

The final result of multiobjective optimization using parallel multicore processing is 22,395 $/hr (GC), 1638 lb/hr (emission) and 0.052219 (AFOR). This is obtained from the final pareto-optimal set of both the cores and it lies within the limits of both cost and emission. The GC and emission obtained using single core processing and a single crossover rate of genetic algorithm are 22,586.02 $/h and 1673.89 lb/h,
respectively (Vakkapatla & Pinni, 2019). As the cost and emission are higher when compared with the results of the proposed method, multicore processing using different parameters of genetic algorithm can be concluded to be superior in performance.

4.2. Case-2: multiobjective generation scheduling on IEEE 57 Bus System
The proposed algorithm is tested on IEEE 57 bus system also to verify its scalability. The results with different crossover rates of 0.65 and 0.7 are presented in Tables 4 and 5, respectively.
It is observed that in the case of single objective like cost minimization, crossover rate of 0.65 gives a solution of 82,617 $/hr and crossover rate of 0.7 gives 84,083 $/hr. This confirms that the search for optimum proceeds in a different direction with different crossover rate of the genetic algorithm. When this principle is extended to multiobjective optimization, the pareto-optimal sets obtained from the two cores of a processor can be observed in Figure 7. The two curves intersect at multiple points, which shows that the overall pareto-optimal set obtained from both the sets is better than each individual set. The overall pareto-optimal set of solutions is presented in Figure 8.

| Table 2. Results of single and multiobjective optimization with crossover rate of 0.65 (IEEE 30 Bus System) |
|-----------------------------------------------|
| Objective | Cost ($/hr) | Emission (lb/hr) | AFOR  | Time of computation |
|-----------|-------------|------------------|-------|---------------------|
| Cost minimization | 21,198 | 2252 | 0.065195 | 1.84 s |
| Emission minimization | 23,077 | 1647 | 0.050662 | 1.55 s |
| Multiobjective optimization | 22,471 | 1642 | 0.052343 | 19 s |

| Table 3. Results of single and multiobjective optimization with crossover rate of 0.7 (IEEE 30 Bus System) |
|-----------------------------------------------|
| Objective | Cost ($/hr) | Emission (lb/hr) | AFOR  | Time of computation |
|-----------|-------------|------------------|-------|---------------------|
| Cost minimization | 21,332 | 2310 | 0.066126 | 1.88 s |
| Emission minimization | 22,744 | 1642 | 0.052323 | 1.66 s |
| Multiobjective optimization | 22,395 | 1638 | 0.052219 | 18 s |

| Table 4. Results of single and multiobjective optimization with crossover rate of 0.65 (IEEE 57 Bus System) |
|-----------------------------------------------|
| Objective | Cost ($/hr) | Emission (lb/hr) | AFOR  | Time of computation |
|-----------|-------------|------------------|-------|---------------------|
| Cost minimization | 82,617 | 11,080 | 0.055052 | 1.58 s |
| Emission minimization | 86,753 | 11,607 | 0.053647 | 1.66 s |
| Multiobjective optimization | 82,690 | 10,031 | 0.044267 | 3.85 s |

| Table 5. Results of single and multiobjective optimization with crossover rate of 0.7 (IEEE 57 Bus System) |
|-----------------------------------------------|
| Objective | Cost ($/hr) | Emission (lb/hr) | AFOR  | Time of computation |
|-----------|-------------|------------------|-------|---------------------|
| Cost minimization | 84,083 | 12,570 | 0.054227 | 1.61 s |
| Emission minimization | 87,698 | 12,600 | 0.054455 | 1.7 s |
| Multiobjective optimization | 86,296 | 9979 | 0.044095 | 3.91 s |
The final result of multiobjective optimization using multicore processing is $86,296$ $\$/hr (GC), 9979 lb/hr (emission) and 0.044095 (AFOR), which lies within the limits of cost and emission. This is obtained from the final pareto-optimal set of both the cores.

As the time of computation varies from 1.58 s to 19 s in the case studies, the proposed algorithm can be used by power system operator for day ahead scheduling.

5. Conclusion
Metaheuristic techniques like genetic algorithms are used to solve optimization problems that are nonlinear and with multiple constraints. The performance of genetic algorithms can be improved by
using parallel processing on different cores of a single CPU. A multicore processing-based generation scheduling algorithm is proposed in this paper. Conflicting objectives like cost minimization and emission minimization are handled by finding the pareto-optimal front. Two threads are scheduled on the two cores of CPU with different crossover rates of the genetic algorithm. The overall pareto-optimal set of solutions is found from the individual sets obtained from each core. The final solution is selected based on the reliability of generating stations. Low-cost hardware and open-source software are used for validation of the algorithm. The proposed algorithm is validated on IEEE 30 Bus and 57 Bus systems. The improvement in results has been observed due to the parallel processing by the two cores of a dual core processor and the usage of different control parameters of genetic algorithm. This algorithm can be used for day ahead scheduling by the power system operator. The proposed algorithm can be extended to systems with renewable sources by considering the uncertainty. In the cases of higher renewable penetration in future, inclusion of reliability indices will improve the quality of power supply to the consumers. The algorithm can be extended to a CPU with higher number of cores using different parameters of a metaheuristic technique. The future scope of the algorithm includes hybridization of two or more metaheuristic methods on different cores of a processor, thereby improving the performance of the overall algorithm.

Funding
The authors received no direct funding for this research.

Author details
Kiran Babu Vakkapatla
E-mail: kiran.vakkapatla.in@ieee.org
ORCID ID: http://orcid.org/0000-0003-4213-6785

Srinivasa Varma Pinni
E-mail: pinnivarma@gmail.com
ORCID ID: http://orcid.org/0000-0001-9734-0201
1 Department of Electrical and Electronics Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India.

Nomenclature
\( \alpha_k, \beta_k, \gamma_k, \eta_k, \delta_k \): The emission coefficients of \( k \)th generating station.
\( a_k, b_k, c_k, d_k, e_k \): The cost coefficients of \( k \)th generating station.
\( F_k \): The forced outage rate of \( k \)th generating station.
\( P_k \): Power scheduled on \( k \)th generating station.
\( p_{\text{max}}^{G_k} \): The maximum active power limit of \( k \)th generating station.
\( p_{\text{min}}^{G_k} \): The minimum active power limit of \( k \)th generating station.
\( P_{ik} \): The power generation at \( k \)th bus.
\( P_{jk} \): The power demand at \( k \)th bus.

Nomenclature
\( S_{kl} \): The apparent power flowing in the line connecting buses \( k \) and \( l \).
\( S_{\text{max}}^{kl} \): The maximum limit of apparent power flow in the line connecting buses \( k \) and \( l \).

Citation information
Cite this article as: Multiobjective generation scheduling using multicore processing-based continuous genetic algorithm, Kiran Babu Vakkapatla & Srinivasa Varma Pinni, Cogent Engineering (2020), 7: 1767019.

Cover image
Source: Author.

References
Ahmed, B. M., & ALhialy, N. F. F. (2019). Optimum efficiency of pv panel using genetic algorithms to touch proximate zero energy house. Civil Engineering Journal, 5(8), 1832–1840. https://doi.org/10.28991/ciej-2019-03091375
Alsac, O., & Stott, B. (1974, May). Optimal load flow with steady-state security. IEEE Transactions on Power Apparatus and Systems, PAS-93(3), 745–751. https://doi.org/10.1109/TPAS.1974.293972
Basu, M. (2008). Dynamic economic emission dispatch using nondominated sorting genetic algorithm-ii. International Journal of Electrical Power & Energy Systems, 30 (2), 140–149. https://doi.org/10.1016/j.ijepes.2007.06.009
Basu, M. (2018). Fast convergence real-coded genetic algorithm for short-term solar-wind-hydro-thermal generation scheduling. Electric Power Components and Systems, 46(11–12), 1239–1249. https://doi.org/10.1080/15325008.2018.1486475
Deb, K. (2001). Multi-objective optimization using evolutionary algorithms. John Wiley & Sons, Inc.
Eaton, J. W., Bateman, D., Houberg, S., & Rik, W. (2018). GNU Octave. Free Software Foundation.
Epperly, T., Edmunds, T., Lamont, A., Meyers, C., Smith, S., Yao, Y., & Drayton, G. (2012, July). High-performance computing for electric grid planning and operations. 2012 IEEE Power and Energy Society General Meeting (pp. 1–4). IEEE. https://doi.org/10.1109/PESGM.2012.6345086

Gong, L., Wang, C., Zhang, C., & Fu, Y. (2019, March). High-performance computing based fully parallel security-constrained unit commitment with dispatchable transmission network. IEEE Transactions on Power Systems, 34(2), 931–941. https://doi.org/10.1109/TPWRS.2018.2876025

Hadizadeh, A., & Tanghatari, E. (2017). Parallel processor architecture with a new algorithm for simultaneous processing of mips-based series instructions. Emerging Science Journal, 1(4), 226–232. https://doi.org/10.28991/ijse-01126

Haupt, R. L., & Haupt, S. E. (2004). Practical genetic algorithms. John Wiley & Sons, Inc.

Razian, S. A., & MahvashMohammadi., H. (2017). Optimizing raytracing algorithm using cuda. Emerging Science Journal, 1(3), 167–178. https://doi.org/10.28991/jse-01119

Shahidehpour, M., Yamin, H., & Li, Z. (2003). Market operations in electric power systems: Forecasting, scheduling, and risk management. Wiley -IEEE. Wiley.

Tippayachai, J., Ongsokul, W., & Ngamroo, I. (2002). Parallel micro genetic algorithm for constrained economic dispatch. IEEE Transactions on Power Systems, 17(3), 790–797. https://doi.org/10.1109/TPWRS.2002.800948

Vakkapatla, K. B., & Pinni, S. V. (2019). Generation rescheduling using multi-objective bi-level optimization. Turkish Journal of Electrical Engineering & Computer Sciences, 27(3), 2086–2096. https://doi.org/10.3906/elk-1804-77

Wood, A. J., Wollenberg, B. F., & Sheblé, G. B. (2013). Power generation, operation, and control. Wiley.

Zaman, M. F., Elsayed, S. M., Ray, T., & Sarker, R. A. (2016, March). Evolutionary algorithms for dynamic economic dispatch problems. IEEE Transactions on Power Systems, 31(2), 1486–1495. https://doi.org/10.1109/TPWRS.2015.2428714

Zimmerman, R. D., Murillo-Sanchez, C. E., & Thomas, R. J. (2011, February). Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. IEEE Transactions on Power Systems, 26(1), 12–19. https://doi.org/10.1109/TPWRS.2010.2051168