ACCSCIENCE PUBLISHING
An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.x, No.x, pp.xx-xx (2026)
https://doi.org/10.36922/IJOCTA026310188
IJOCTA
RESEARCH ARTICLE

IP-C-PDL: A chaos-embedded trust-region interior-point framework for constrained economic dispatch in large-scale power systems

Jangkung Raharjo[1,2], Rifki Rahman Nur Ikhsan[1,2*], and Lindiasari Martha Yustika[1,2]

[1]School of Electrical Engineering, Telkom University, Jl. Telekomunikasi No. 1, Terusan Buah Batu, Bandung, West Java, Indonesia
[2]Center of Excellence of Sustainable Energy and Climate Change, Telkom University, Jl. Telekomunikasi No. 1, Terusan Buah Batu, Bandung, West Java, Indonesia
jangkungraharjo@telkomuniversity.ac.id, rfkrhmn@telkomuniversity.ac.id, lindiasarimartha@telkomuniversity.ac.id

ARTICLE INFO

Article history:
Received: July 27, 2026
Revised: August 29, 2026
Accepted: September 11, 2026
Published Online: November 13, 2026

Keywords:
Constrained optimisation modelling
Trust-region methods
Interior-point framework
Economic dispatch
Large-scale power systems
Computational intelligence

AMS Classification 2010:
90C30; 90C51; 90C90; 49M37

ABSTRACT

Constrained economic dispatch (ED)—the classical, single-bus, active-power-only dispatch problem for thermal generation, not an optimal power flow or unit-commitment problem—in large-scale interconnected power systems such as the Java–Madura–Bali (Jamali) network is non-convex and ramp-rate-coupled: deterministic solvers stall at local optima, while stochastic solvers incur unacceptable run-to-run variance. This paper proposes IP-C-PDL (Interior-Point Chaotic Powell's Dog-Leg), which, within a single interior-point iteration, couples a Powell's Dog-Leg trust-region step (Cauchy/quasi-Newton selection via accumulated SR1/BFGS curvature) with a stall-adaptive tent-map chaotic perturbation for ergodic basin escape; a multi-start pool, elite archive, and periodic SQP refinement complete the framework using only three user-specified parameters. On twelve CEC benchmark functions (30 runs, 12 methods), IP-C-PDL ranks 1 in 1 of 12 and top-3 in 7 of 12. On six ED scenarios spanning 5-, 16-, and 42-unit systems (50 runs, 10 methods, single-period and 24-hour horizons), it ranks 1 in 4 of 6 (mean rank 2.17/10) with zero constraint violations, yielding USD 22,239.29/day (σ = USD 0.41) on the 42-unit 24-hour case—0.16% better than the second-ranked comparator ($t = -48.26$, $p < 10^{-42}$) and 4.67% better than the worst ($t = -180.85$, $p < 10^{-70}$)—confirmed by Friedman, ANOVA, and Wilcoxon–Holm–Bonferroni tests. A five-factor sensitivity analysis identifies multi-start pool size as the dominant factor—with a cliff-edge improvement at just two starts—while the framework is insensitive to the remaining four hyperparameters and stabilises by 400 iterations. A policy sensitivity analysis further links dispatch quality to fiscal savings, Biaya Pokok Penyediaan (BPP) subsidy reduction, and avoided carbon emissions in the Indonesian power sector.

CC BY

1. Introduction

Economic dispatch (ED) is a fundamental operational problem in power systems, aiming to allocate real power generation among committed units to satisfy load demand at minimum fuel cost while respecting physical and operational constraints.[1–4] As formalised in Section 2, this paper addresses ED in its classical, single-bus, active-power-only sense—the generation-allocation problem, not a network-constrained optimal power flow or a unit-commitment problem—for thermal generating units. This work is motivated by the economic and operational challenges of the Java–Madura–Bali (Jamali) interconnected power

*Corresponding Author

system—Indonesia's largest electricity network, accounting for approximately 69% of national generation capacity and supplying electricity to around 160 million people via a 500 kV transmission backbone managed by PT PLN (Perusahaan Listrik Negara). The economic stakes of dispatch optimisation in this system are substantial. PLN operates under a regulated tariff structure in which fuel costs are borne by the national budget through subsidy mechanisms when market prices exceed the cost recovery ceiling;[5] inefficient dispatch therefore creates direct fiscal liability at the sovereign level. At the same time, Indonesia's national electricity supply plan (RUPTL 2025–2034)[6] earmarks 48.2% of planned new capacity for the Jamali region—including up to 9.6 GW of variable renewable energy by 2030—placing increasing pressure on dispatch algorithms to manage intermittency while maintaining cost efficiency and grid stability. Improving dispatch quality in the Jamali system therefore advances three coupled national priorities: fiscal consolidation (lower subsidy expenditure), decarbonisation (earlier displacement of high-marginal-cost thermal generation), and energy security (reliable supply to the economic heartland). In networks of this scale, ED is characterized by highly non-convex fuel-cost functions, tight ramp-rate constraints coupling consecutive dispatch intervals, and daily multi-period scheduling requirements that make reliable and efficient optimization indispensable for both economic performance and national energy security.[7–12] Broader generation scheduling problems—including maintenance scheduling and generator commitment—face similar combinatorial complexity,[13–18] and the growing integration of renewable sources further complicates dispatch due to intermittency and supply uncertainty.[19,20]

1.1. Related work

A useful lens for organising this literature is the trade-off between optimality/efficiency and constraint-handling flexibility. Deterministic, calculus-based solvers—lambda-iteration, interior-point, SQP—are repeatable by construction: given the same starting point and data, they return the same solution every time, and when they converge they typically do so efficiently and to a locally (often globally, on well-behaved landscapes) optimal point at low computational cost.[21,22] Their weakness is rigidity: bound and equality constraints are enforced through barrier functions, active-set logic, or exact projections that presuppose a well-structured feasible region, so on the non-convex, ramp-rate-coupled landscapes typical of large-scale ED this rigidity can produce infeasibility or convergence to a poor local optimum whenever the constraint structure departs from what the method was built around. Population-based ("AI") metaheuristics largely invert this trade-off: their stochastic, population-level search absorbs constraint violations gracefully through penalty functions, repair operators, or adaptive constraint-handling schemes, remaining feasible and flexible across a far wider range of operational conditions—but at the cost of run-to-run variance that makes any single run's outcome probabilistic rather than repeatable, and often at higher computational expense for comparable solution quality.[23] This paper's architectural choice—embedding a small, deterministic tent-map perturbation and a structured multi-start pool inside a classical trust-region interior-point loop, rather than replacing the deterministic core with a population-based search—is a direct response to this trade-off: it retains the repeatability and efficiency of the deterministic solver while borrowing just enough constraint-handling flexibility from the AI paradigm to avoid the rigidity failure mode, without paying the full variance cost of a genuinely population-based method. The remainder of this section reviews each side of that trade-off in turn, before positioning IP-C-PDL relative to existing hybrids. Classical deterministic methods such as lambda-iteration and gradient projection offer analytical tractability but struggle with non-convex cost functions and ramp-rate coupling.[22,24–26] Interior-point (IP) methods represent the state of the art among deterministic solvers: they enforce bound constraints efficiently via logarithmic barrier functions and provide warm-start capability, yet remain vulnerable to local-optimum entrapment on the non-convex landscapes typical of large-scale systems.[21,27–30] Population-based metaheuristics—particle swarm optimization (PSO),[20,31–33] differential evolution (DE),[34] whale optimization,[35,36] dragonfly,[37] and JAYA-based methods[38–40]—offer broad global search but their stochastic nature introduces high solution variance and substantial computational overhead for large systems.[23,41,42] Evolutionary programming, genetic algorithms, and their variants share this limitation: run-to-run variability makes them unsuitable for real-time dispatch environments requiring repeatable, deterministic convergence. Hybrid approaches combine the local precision of gradient-based methods with the global reach of stochastic search[43–48]—e.g., PSO with local refinement, chaotic-enhanced DE,

and penalty-based IP with perturbation restart. However, existing hybrids typically layer metaheuristic exploration on top of a deterministic solver as a post-processing step, rather than systematically embedding trust-region control and adaptive chaos within the inner IP update loop. This distinction matters: external perturbation cannot leverage the Hessian curvature accumulated by the IP solver, and trust-region radius control is lost between the two stages. Powell's dog-leg (PDL) is a well-established trust-region strategy that interpolates between the Cauchy gradient step and the full quasi-Newton step, providing superlinear convergence near the optimum without the ill-conditioning risks of pure Newton methods.[49–54] Chaotic maps, particularly the tent map, generate ergodic, non-repeating perturbation sequences that efficiently cover the search space without population overhead.[33,55–57] Beyond optimisation, tent-map and related chaotic and memristive-map constructions have recently underpinned a broad and rapidly growing body of engineering applications, including FPGA-realised memristive neural hardware,[58] high-dimensional chaotic-map image encryption,[59–61] and video-segment encryption together with its cryptanalytic evaluation[62,63]—evidence that the ergodic, pseudorandom-quality behaviour exploited here is a well-validated, general-purpose property of chaotic maps rather than an assumption specific to this optimisation setting. This novelty claim was checked, rather than asserted unchecked, by a targeted search of the IEEE Xplore and general engineering literature using combinations of the queries "interior point", "trust region", "chaotic"/"chaos", "Powell's dog-leg", and "economic dispatch"; we did not have access to, and therefore did not search, Scopus or Web of Science directly, so this claim should be read as scoped to the databases actually searched rather than as an exhaustive literature-wide claim. The closest related work located is El-Shorbagy's trust-region-based chaotic search for multi-objective optimisation,[64] whose own review states that, prior to that paper, "there is not any publication that discusses the issue of applying a chaotic search with the [trust-region] algorithm"; that paper, however, couples chaotic search with a generic trust-region method for multi-objective benchmark and abrasive-water-jet-machining problems, without an interior-point formulation, without Powell's dog-leg step selection, and without application to power system economic dispatch. Despite both mechanisms being individually well-understood, to the best of our knowledge, and following this search, no prior work has integrated PDL trust-region step selection and adaptive tent-map chaotic perturbation within a single IP iteration loop for power system ED, nor validated such a framework against a comprehensive set of comparators under realistic grid-scale conditions.

1.2. Contributions

This paper fills that gap by proposing IP-C-PDL (Interior-Point Chaotic Powell's Dog-Leg) as a unified computational modelling framework for constrained ED. The design principle is architectural rather than additive: rather than layering exploration mechanisms on top of an existing solver, all components—trust-region step selection, adaptive chaotic perturbation, Hessian update, penalty adaptation, and SQP refinement—operate within a single coherent iteration structure that shares state (Hessian approximation, trust-region radius, stall counter) across the deterministic and stochastic components. Test cases are derived from generator data representative of the Jamali interconnected system, ensuring evaluation under realistic operational conditions. The specific contributions are fourfold. **(1) Integrated computational framework**: IP-C-PDL couples a Powell's Dog-Leg trust-region step selector with stall-adaptive tent-map chaotic perturbation within a single interior-point iteration—an architectural choice that preserves Hessian curvature information across both the deterministic and stochastic phases, requiring only three user-specified modelling parameters. **(2) Unified mathematical formulation**: complete formulations for the augmented penalty subproblem, adaptive penalty scheduling, SR1/BFGS quasi-Newton Hessian update, PDL step selection with quadratic interpolation, Armijo line search, tent-map chaotic state evolution, and SQP refinement subproblem are presented as a coherent coupled system with full dimensional analysis. **(3) Scalable constrained dispatch modelling**: the framework is validated across six dispatch scenarios spanning three system scales (5-, 16-, 42-unit) and two planning horizons (single-period and 24-hour rolling), all with operational constraints derived from a real utility environment; fifty independent runs per scenario with rigorous multi-test statistical analysis establish that improvements are systematic and not artefacts of initialisation. **(4) Economic and policy quantification**: a dynamic sensitivity analysis translates computational dispatch quality into fiscal savings, BPP subsidy reduction, and avoided carbon emissions

across coal price scenarios, providing a quantitative link between modelling performance and energy policy outcomes. Section 2 formulates the constrained dispatch model. Section 3 presents the IP-C-PDL computational framework, including its architectural design, mathematical components, and complexity analysis. Section 4 details the experimental setup. Section 5 reports CEC benchmark results. Section 6 presents ED results and constraint verification. Section 7 provides the statistical analysis. Section 8 conducts the algorithmic and economic–policy sensitivity analyses. Section 9 concludes the paper.

2. Constrained dispatch model

2.1. Single-period economic dispatch

Given n generating units, the single-period ED follows the classical quadratic-cost formulation[1] and minimizes total fuel cost:

$$\min_{\mathbf{P}} \; F(\mathbf{P}) = \sum_{i=1}^{n} f_i(P_i) \tag{1}$$

$$f_i(P_i) = a_i P_i^2 + b_i P_i + c_i \tag{2}$$

where $\mathbf{P} = [P_1, \ldots, P_n]^\top$ is the dispatch vector, P_i (MW) is the real power output of unit i, and a_i, b_i, c_i are the quadratic fuel cost coefficients (native units IDR/MW^{2}h, IDR/MWh, IDR/h). All monetary results reported in this paper are converted to US Dollars at the fixed rate IDR 16,100 = USD 1; see the note in Section 3.2 for details. Subject to:

$$\sum_{i=1}^{n} P_i = P_D \tag{3}$$

$$P_i^{\min} \le P_i \le P_i^{\max}, \quad i = 1, \ldots, n \tag{4}$$

where P_D (MW) is the system load demand, and $P_i^{\min}$, $P_i^{\max}$ (MW) are the generation limits of unit i. **Scope of this formulation.** (1)–(4) is the classical, single-bus ("copper-plate") economic dispatch problem: it balances active power only, against a single systemwide demand P_D, with no representation of network topology, transmission-line thermal limits, bus voltage limits, reactive power, or losses—i.e. this is economic dispatch in the traditional sense, not an optimal power flow (OPF) or unit-commitment problem. All generating units are treated as thermal generation with a quadratic fuel-cost curve (2); this is a standard modelling idiom for fossil-fuel (including but not limited to coal) thermal plant and does not represent any one fuel type exclusively; nor does it model non-thermal (e.g. hydro, renewable) generation, which is outside the present scope. The three test systems of Section 4.2 instantiate this same single-bus model at three different unit counts and demand levels; they are three problem-size instances of one modelling scope, not three different network representations.

2.2. Multi-period (24-hour) economic dispatch

Extending to a 24-hour rolling horizon with hourly periods $t = 1, \ldots, T$, following standard multi-period ED formulations:[8,9]

$$\min_{\{P_{i,t}\}} \sum_{t=1}^{T} \sum_{i=1}^{n} f_i(P_{i,t}) \tag{5}$$

where $P_{i,t}$ (MW) is the dispatch of unit i at hour t, subject to power balance (3) and generation limits (4) at each hour, and ramp-rate constraints:[7]

$$-RD_i \;\le\; P_{i,t} - P_{i,t-1} \;\le\; RU_i \tag{6}$$

for $i = 1, \ldots, n$ and $t = 2, \ldots, T$. where RU_i, RD_i (MW/h) are the ramp-up and ramp-down limits. The effective per-hour bounds folding ramp-rate into generation limits are:

$$\begin{aligned} \tilde{P}_i^{\min}(t) &= \max\big(P_i^{\min},\; P_{i,t-1} - RD_i\big), \\ \tilde{P}_i^{\max}(t) &= \min\big(P_i^{\max},\; P_{i,t-1} + RU_i\big) \end{aligned} \tag{7}$$

The 24-hour dispatch is therefore solved as a sequence of T independent n-dimensional single-period subproblems rather than a single $(n \times T)$-dimensional joint optimisation: at hour t, the PDL trust-region step (Section 3.4), the accumulated SR1/BFGS curvature information (Section 3.3), and the tent-map perturbation (Section 3.5) all operate on the n-dimensional decision vector $\mathbf{P}_t$ exactly as in the single-period case. Temporal coupling across hours enters only through the feasible region via the effective bounds (7), so a solution found for hour t is guaranteed ramp-feasible with respect to what was actually dispatched at hour $t-1$; no information about future hours is used when solving hour t (a causal, receding-horizon structure), and no optimiser state is carried over between hours.

3. The IP-C-PDL computational framework

3.1. Overview and architectural design

IP-C-PDL is designed as a coupled deterministic–stochastic computational framework in which all components share a common iteration state. The key architectural decision is that the trust-region radius Δ_k, the quasi-Newton Hessian approximation $\mathbf{H}_k$, the stagnation counter stall, and the adaptive penalty parameter μ_k are all updated

within a single iteration loop— not across separate solver calls. This ensures that the chaotic perturbation component operates on an objective landscape whose local curvature is already well-characterised by the accumulated Hessian, and that trust-region radius adaptation reflects the actual nonlinear behaviour of the penalised objective $\Phi(\mathbf{P};\mu_k)$ rather than an approximation. Three integrated components form the loop: (1) an IP warm-start pool providing high-quality initialisation (Section 3.2); (2) a Powell's Dog-Leg trust-region step selector providing curvature-aware deterministic search (Section 3.4); and (3) a stall-adaptive tent-map chaotic perturbation providing ergodic basin escape (Section 3.5). Algorithm 1 gives the complete pseudocode integrating all three components within the single iteration loop described below.

3.2. Interior-point warm-start and penalty subproblem

The power-balance constraint (3) is handled in the main loop by augmenting the fuel-cost objective (1)–(2) with a quadratic penalty:

$$\min_{\mathbf{P}} \ \Phi(\mathbf{P};\mu) = F(\mathbf{P}) + \mu\left(\sum_{i=1}^{n} P_i - P_D\right)^2 \quad (8)$$

where μ (denominated in the solver's native computational units, see note below) penalizes power-balance violation quadratically, following the classical exterior-penalty formulation of interior-point ED solvers.[21,27] Rather than fixing μ, it is adapted online at iteration k so that the penalty tightens automatically as the balance error shrinks, via the schedule in (9):

$$\mu_k = \min\left(\mu_0 \cdot \max\left(1,\ \frac{10^3}{\max(\varepsilon_k,\ 10^{-3})}\right),\ 10^{10}\right) \quad (9)$$

where $\mu_0 = 10^6$ is an internal penalty-scaling constant (tuned in the solver's native computational units, consistent with the IDR-denominated cost coefficients used at run time; see note on currency reporting below) and $\varepsilon_k = |\sum_i P_i - P_D|$ (MW) is the balance error: as $\varepsilon_k \to 0$ the multiplier in (9) grows, progressively enforcing the equality constraint (3) more strictly, up to the numerical ceiling 10^{10} that prevents ill-conditioning in the Hessian update (12). *Note on currency reporting.* All optimisation runs were executed with cost coefficients denominated in Indonesian Rupiah (IDR), and internal solver constants (μ_0, the penalty cap 10^{10}, and related tolerances) are tuned in that native scale. All reported monetary results—dispatch costs, savings, and policy metrics in Sections 6–8—are converted to US Dollars for readability at the fixed rate IDR 16,100 = USD 1 (the 2024 Indonesian coal benchmark rate), consistent with the coal-price reference used in Section 8.9. The penalised objective (8) is what the main trust-region loop below optimises at every iteration; the multi-start pool, by contrast, is seeded from candidates that solve the hard-constrained subproblem directly. The first two pool candidates both solve the same equality-constrained subproblem (10), differing only in the fmincon algorithm used: the first via interior-point (100 iterations), the second via sqp:

$$\min_{\mathbf{P}} \ F(\mathbf{P}) \quad \text{s.t.}$$
$$\mathbf{1}^\top \mathbf{P} = P_D, \quad \mathbf{P}^{\min} \le \mathbf{P} \le \mathbf{P}^{\max} \quad (10)$$

The multi-start pool $\mathcal{S} = \{x^{(1)}, \ldots, x^{(N_S)}\}$ is assembled as: (1) a lambda-iteration start (bisection on the equal-incremental-cost coordination equation); (2) an interior-point solution of (10); (3) an SQP solution of (10); (4) $N_S - 3$ random feasible points via the repair operator (11). Every candidate—including the IP-C-PDL iterate at line-search and hourly-schedule steps elsewhere—is passed through this same repair operator to restore exact power balance after bound clipping. Given any $\mathbf{P}$ and bounds $\mathbf{l}, \mathbf{u}$, the operator alternates a hard clip with a proportional redistribution of the balance error $d = \mathbf{1}^\top \mathbf{P} - P_D$ across whichever units still have headroom, iterating up to 200 times or until $|d| < 10^{-4}$ MW:

$$P_i \ \leftarrow \ \begin{cases} P_i - \dfrac{(P_i - l_i)}{\sum_j (P_j - l_j)}\, d & d > 0 \\ P_i + \dfrac{(u_i - P_i)}{\sum_j (u_j - P_j)}\, |d| & d < 0 \end{cases} \quad (11)$$

followed by re-clipping to $[l_i, u_i]$ before the next pass. This proportional-headroom scheme guarantees that the repaired point remains feasible with respect to (4) while restoring feasibility with respect to (3) to numerical precision. Only after the pool is assembled does each start enter the main loop, where the trust-region step operates on the penalised objective $\Phi(\mathbf{P};\mu_k)$ (8)–(9) rather than on $F(\mathbf{P})$ directly.

3.3. Hessian approximation: SR1/BFGS update

Define $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ and $\mathbf{y}_k = \nabla\Phi_{k+1} - \nabla\Phi_k$. When $\mathbf{s}_k^\top \mathbf{y}_k > 10^{-10}$, the BFGS update[21] is applied:

$$\mathbf{H}_{k+1} = \big(\mathbf{I} - \rho_k \mathbf{s}_k \mathbf{y}_k^\top\big)\mathbf{H}_k\big(\mathbf{I} - \rho_k \mathbf{y}_k \mathbf{s}_k^\top\big) + \rho_k \mathbf{s}_k \mathbf{s}_k^\top,$$
$$\rho_k = \frac{1}{\mathbf{s}_k^\top \mathbf{y}_k} \tag{12}$$

where ρ_k (MW^{-2}), $\mathbf{s}_k$ (MW), $\mathbf{y}_k$ (native cost units/MW), and $\mathbf{I}$ is the identity matrix. Otherwise, the SR1 fallback is applied:

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\mathbf{r}_k \mathbf{r}_k^\top}{\mathbf{r}_k^\top \mathbf{y}_k},$$
$$\mathbf{r}_k = \mathbf{s}_k - \mathbf{H}_k \mathbf{y}_k \tag{13}$$

where $\mathbf{r}_k$ (MW) is the SR1 residual. Both updates are accepted only when $\mathrm{rcond}(\mathbf{H}_{k+1}) > 10^{-14}$; otherwise $\mathbf{H}_{k+1} \leftarrow \mathbf{I}$.

3.4. Powell's dog-leg step

The gradient $\mathbf{g}_k = \nabla\Phi(\mathbf{x}_k; \mu_k)$ is, for the economic-dispatch objective, the exact analytical gradient of the quadratic fuel-cost penalty function (closed-form, no finite-difference approximation involved); for the black-box CEC benchmark functions of Section 5, where no closed form is assumed, it is instead approximated by central finite differences with step $\epsilon = 10^{-6}$ (corrected from 10^{-5} in an earlier version of this manuscript to match the implemented step size). It is then scaled per (14) so that units with widely different generation ranges contribute comparably to the search direction:

$$\boldsymbol{\sigma} = \max(\mathbf{u} - \mathbf{l},\ \mathbf{1}),$$
$$\mathbf{g}_s = \frac{\mathbf{g}_k \,/\, \boldsymbol{\sigma}}{\max\big(\|\mathbf{g}_k\|/\|\boldsymbol{\sigma}\|,\ 10^{-12}\big)} \tag{14}$$

where $\boldsymbol{\sigma}$ (MW) is the variable range vector, $\mathbf{l}$, $\mathbf{u}$ (MW) are bound vectors, and $\mathbf{g}_s$ is dimensionless. Following the classical dog-leg construction,[49,50,54] the quasi-Newton and Cauchy steps inside the trust region are given by (15):

$$\mathbf{p}_N = -\mathbf{H}_k\,\mathbf{g}_s, \qquad \mathbf{p}_C = -\tau_c\,\frac{\Delta_k}{\|\mathbf{g}_s\|}\,\mathbf{g}_s,$$
$$\tau_c = \min\!\left(\frac{\|\mathbf{g}_s\|^3}{\Delta_k\,\mathbf{g}_s^\top \mathbf{H}_k\,\mathbf{g}_s + 10^{-12}},\ 1\right) \tag{15}$$

where $\mathbf{p}_N$, $\mathbf{p}_C$ (MW) are the full quasi-Newton and Cauchy steps, Δ_k (MW) is the trust-region radius, and $\tau_c \in (0, 1]$ is the Cauchy scaling factor. The quasi-Newton step $\mathbf{p}_N$ offers fast local convergence but can leave the trust region on strongly curved parts of the penalised landscape, while the Cauchy step $\mathbf{p}_C$ is the safe steepest-descent fallback; Powell's dog-leg interpolates between the two rather than committing to either in isolation.[51,52] The resulting trial step $\mathbf{p}_k$ (MW) selects among three cases (16):

$$\mathbf{p}_k = \begin{cases} \mathbf{p}_N & \|\mathbf{p}_N\| \le \Delta_k \\ \Delta_k\,\dfrac{-\mathbf{g}_s}{\|\mathbf{g}_s\|} & \|\mathbf{p}_C\| \ge \Delta_k \\ \mathbf{p}_C + \tau^*(\mathbf{p}_N - \mathbf{p}_C) & \text{otherwise} \end{cases} \tag{16}$$

(Case 1: Newton step fits inside trust region; Case 2: Cauchy step exceeds trust region; Case 3: dog-leg interpolation.) where $\tau^* \in [0, 1]$ is the positive root of the quadratic $\|\mathbf{p}_C + \tau(\mathbf{p}_N - \mathbf{p}_C)\|^2 = \Delta_k^2$, with $\mathbf{d} = \mathbf{p}_N - \mathbf{p}_C$, solved via the standard quadratic formula (17):

$$\underbrace{\|\mathbf{d}\|^2}_{a_q}\,\tau^2 + \underbrace{2(\mathbf{p}_C^\top \mathbf{d})}_{b_q}\,\tau + \underbrace{\|\mathbf{p}_C\|^2 - \Delta_k^2}_{c_q} = 0 \tag{17}$$

where a_q, b_q, c_q (MW^2) are the quadratic coefficients. An Armijo backtracking line search[53] finds $\alpha_k \in (0, 1]$ satisfying $\Phi(\mathbf{x}_k + \alpha_k \mathbf{p}_k) \le \Phi(\mathbf{x}_k) + 10^{-4}\alpha_k\,\mathbf{g}_s^\top \mathbf{p}_k$. The trust-region radius is then adapted by the classical predicted-versus-actual-reduction ratio ρ_k (18):

$$\rho_k = \frac{\Phi(\mathbf{x}_k; \mu_k) - \Phi(\mathbf{x}_k + \alpha_k \mathbf{p}_k; \mu_k)}{-\mathbf{g}_s^\top(\alpha_k \mathbf{p}_k)} \tag{18}$$

and the radius update rule (19) shrinks Δ_k when the model over-predicts the achieved reduction ($\rho_k < 0.25$), expands it when the model is trustworthy ($\rho_k > 0.75$), and leaves it unchanged otherwise—the standard trust-region control strategy:[50,54]

$$\Delta_{k+1} = \begin{cases} \max(\Delta_k/2,\ \epsilon_\Delta) & \rho_k < 0.25 \\ \min(\Delta_k \times 2,\ 500) & \rho_k > 0.75 \\ \Delta_k & \text{otherwise} \end{cases} \tag{19}$$

where ρ_k (dimensionless) measures model fidelity and $\epsilon_\Delta = 10^{-8}$ MW.

3.5. Chaotic perturbation via tent map

Following the ergodic-search rationale established for tent-map-based optimisation,[55–57] a chaotic state vector $\mathbf{s}^{(k)} \in (0, 1)^n$ evolves via the piecewise-linear tent map (20):

$$s_i^{(k+1)} = \begin{cases} 2\,s_i^{(k)} & s_i^{(k)} < 0.5 \\ 2\big(1 - s_i^{(k)}\big) & \text{otherwise} \end{cases} \tag{20}$$

initialized from $(0.1, 0.9)^n$. Unlike a pseudo-random sequence, the tent map is deterministic yet aperiodic, so the same seed reproduces identical perturbations—preserving the run-to-run repeatability that motivates this paper's deterministic-first design—while still covering the search space as densely as a stochastic sequence.[33] Writing $\boldsymbol{\delta}^{(k)} \equiv \mathbf{s}^{(k)} - 0.5$ for the raw offset, $\boldsymbol{\delta}^{(k)}$ is normalized to a unit direction only when its Euclidean norm exceeds a safeguard threshold

$\epsilon_v = 10^{-12}$; otherwise the (numerically negligible) raw offset is used as-is for that iteration, so that division by a near-zero norm never occurs. Explicitly,

$$\mathbf{v} = \begin{cases} \dfrac{\boldsymbol{\delta}^{(k)}}{\|\boldsymbol{\delta}^{(k)}\|} & \|\boldsymbol{\delta}^{(k)}\| > \epsilon_v \\ \boldsymbol{\delta}^{(k)} & \text{otherwise} \end{cases} \tag{21}$$

is added to the PDL step with the stall-adaptive probability $c_p(k)$ (22):

$$c_p(k) = \min\big(c_{p0} + \text{stall} \times 0.004,\ 0.55\big) \tag{22}$$

where $c_{p0} \in (0, 1)$ is the base probability, stall is the number of consecutive non-improving iterations, and 0.55 is the upper cap; this is the mechanism whose near-total insensitivity across $c_{p0} \in [0.05, 0.50]$ is quantified empirically in Section 8. The perturbation magnitude Δ_c (23) is scaled by the current trust-region radius so that exploration contracts as the iterate approaches convergence:

$$\Delta_c = \Delta_k \cdot \min\big(0.25 + \text{stall}/120,\ 0.9\big) \tag{23}$$

where Δ_c (MW) protects near-optimal solutions by capping at $0.9\Delta_k$. A formal boundedness and ergodicity argument for the tent map underlying this perturbation, complementing the empirical validation above, is given in Appendix A. We also note, for implementation transparency, that the alternative-chaotic-map ablation of Section 8.7 reimplements the perturbation-generation step in a separate routine shared across all four candidate maps; this shared routine additionally clamps the per-component state to $[10^{-9}, 1 - 10^{-9}]$ before forming $\mathbf{v}$, a step the main algorithm's tent-map generator (**Equation (21)**) does not apply. The two implementations are therefore numerically close but not bit-identical for the tent-map case; Section 8.8 quantifies how often this clamp is active and confirms it does not materially affect the ablation's conclusions.

3.6. Elite archive and periodic SQP refinement

An elite archive $\mathcal{E}$ of target size $K = 6$ (economic-dispatch problems; $K = 5$ for the lower-dimensional CEC benchmark implementation, a minor configuration difference between the two experiment drivers) retains the best solutions found. Every $e_i = \max(20 - \lfloor \text{stall}/10 \rfloor, 10)$ stall steps, SQP refinement (10) is applied to the local best. A full restart from the best elite member plus Gaussian noise (resetting $\mathbf{H}$ to $\mathbf{I}$) is triggered every 50 stall steps in the economic-dispatch driver and every 60 stall steps in the CEC benchmark driver; this restart-interval difference, like the elite-archive size above, is a configuration difference between the two experiment drivers rather than a claimed tuning result.

3.7. Algorithm pseudocode

Algorithm 1. IP-C-PDL

Require: cost coefficients a, b, c; bounds $\mathbf{l}, \mathbf{u}$; demand P_D; tunable parameters N_S, MAX_IT, c_{p0}; fixed internal constant $\mu_0 = 10^6$
Ensure: best dispatch $\mathbf{P}^*$

Build $\mathcal{S}$: lambda-iter + IP (8) + SQP (10) + random repair
$\mathcal{E} \leftarrow \emptyset$, $f^* \leftarrow \infty$
for each start $s \in \mathcal{S}$ **do**
  $\mathbf{x} \leftarrow s$, $\mathbf{H} \leftarrow \mathbf{I}$, $\Delta \leftarrow 0.1\|(\mathbf{u} - \mathbf{l}) \oslash \boldsymbol{\sigma}\|$, stall $\leftarrow 0$
  for $k = 1$ **to** MAX_IT **do**
    Adapt μ_k (9); compute $\mathbf{g}_s$ (14)
    Update $\mathbf{H}$ via BFGS (12) or SR1 (13)
    Compute $\mathbf{p}_k$ via (15)–(17), selecting the case in (16)
    if rand()$< c_p(k)$ (22) **then**
      $\mathbf{p}_k \leftarrow \mathbf{p}_k + \Delta_c \cdot \mathbf{v}$ ((20), (23))
    end if
    Armijo line search; project to $[\mathbf{l}, \mathbf{u}]$
    Update Δ_{k+1}, $\mathcal{E}$, stall via ratio (18) and rule (19)
    if stall $\bmod e_i = 0$ **then**
      SQP refinement (10) from local best
    end if
    if stall $\bmod 50 = 0$ **then**
      Restart from best in $\mathcal{E}$ + noise; $\mathbf{H} \leftarrow \mathbf{I}$
    end if
  end for
end for
return $\mathbf{P}^*$ with lowest feasible cost in $\mathcal{E}$

3.8. Computational complexity and hessian conditioning

The per-iteration cost of IP-C-PDL is dominated by three operations. (i) *Gradient computation*: as noted in Section 3.4, this differs by problem class. For the economic-dispatch objective the gradient is closed-form (analytical), so this step costs a single, already-required objective evaluation and contributes no additional function calls beyond the objective itself. For the black-box CEC benchmark functions, central finite differences require $2n$ function evaluations per iteration, giving $\mathcal{O}(n)$ cost per step—the same order as gradient-based comparators; the per-iteration evaluation-count discussion below therefore applies specifically to the CEC configuration, not to the economic-dispatch results. (ii) *Hessian update*: the SR1/BFGS rank-one and rank-two updates (12)–(13) each cost $\mathcal{O}(n^2)$ in time and $\mathcal{O}(n^2)$

in storage; for the 42-unit system ($n = 42$), this amounts to $\mathcal{O}(1764)$ floating-point operations per update, which is negligible relative to function evaluation overhead. (iii) *PDL step computation*: solving the quadratic $\|\mathbf{p}_C + \tau(\mathbf{p}_N - \mathbf{p}_C)\|^2 = \Delta_k^2$ via (17) costs $\mathcal{O}(n)$ after the linear systems are formed, so the PDL selection adds no meaningful overhead beyond a standard gradient step. The overall per-iteration complexity is therefore $\mathcal{O}(n^2)$, identical to that of a dense quasi-Newton method,[21] with the chaotic perturbation adding only $\mathcal{O}(n)$ tent-map evaluations. For the CEC benchmark configuration, where finite-difference gradients are used, the per-start evaluation count scales as $2n \times \max_it$; at the largest CEC dimension used in this paper ($n = 5$, $\max_it = 500$) this is 5,000 evaluations per run, comparable in order of magnitude to population-based comparators running dozens of agents for hundreds of generations.[23] For the economic-dispatch configuration, by contrast, the analytical gradient means the per-iteration cost is dominated by the single required objective evaluation rather than by $2n$ finite-difference evaluations, so a finite-difference-based evaluation count is not the right way to characterise its computational budget; Section 8.7 reports wall-clock time directly for that reason, using the same hardware/session runtime comparison across all ten ED methods. An earlier version of this manuscript reported a single $2 \times 42 \times 400 \times 6 = 201{,}600$-evaluation figure for the 42-unit economic-dispatch case; that figure implicitly assumed central-difference gradients throughout and has been removed, since the economic-dispatch runs use the analytical gradient described above and do not incur that cost. Hessian conditioning is monitored via $\mathrm{rcond}(\mathbf{H}_{k+1})$ at each update; a candidate update with reciprocal condition number below 10^{-14} is rejected and the previous, well-conditioned $\mathbf{H}_k$ is retained for that iteration (identity resets occur only at the periodic elite restarts described in Section 3.6 above). This threshold is chosen to balance conditioning against information loss: rejecting too aggressively discards curvature information accumulated over many iterations, while rejecting too conservatively allows ill-conditioned Hessians to corrupt the PDL step direction. A dedicated ablation over rcond thresholds (Section 8.7) measures the rejection rate directly rather than assuming it is small: at the default 10^{-14} threshold, 42.0% of candidate updates are rejected and the previous $\mathbf{H}_k$ is retained—substantially higher than earlier unmeasured statements in prior versions of this manuscript had assumed—yet mean cost and solution quality are identical to six decimal places across all four tested thresholds, confirming that the SQP-refinement and multi-start-pool machinery compensate for the discarded curvature information and that the SR1/BFGS update's rejection behaviour, while frequent, does not compromise convergence quality on the test problems. The stagnation-triggered SQP refinement subproblem (10) introduces an additional $\mathcal{O}(n^3)$ SQP factorisation cost at most every e_i stall steps, but this cost is amortised over the full iteration sequence and accounts for less than 5% of total runtime on the 42-unit problem.

4. Experimental setup

4.1. Benchmark functions

Twelve functions (F01–F12, dimensions 1–5, **Table 1**) are drawn from the benchmark suite of the IEEE WCCI/CEC 2020 Competition on Niching Methods for Multimodal Optimization,[65] whose problem definitions follow the original technical report[66] unchanged—a standard test set for multimodal optimisation chosen here for its mix of deceptive, scalable, and composition landscapes that stress different aspects of local-optimum escape. Each is evaluated with 30 independent runs per method. Comparators: Nelder-Mead, Powell, fmincon-SQP, fmincon-TRR, COBYLA, fmincon-AS, fmincon-IP-LBFGS, CG, BFGS, PSO, and DE. **A note on comparator naming.** An earlier version of this manuscript labelled four of these nine deterministic comparators after their closest SciPy identifiers (SLSQP, TNC, trust-constr, L-BFGS-B), which could be read as claiming independent third-party solver implementations. All nine deterministic comparators are in fact implemented in MATLAB, which does not natively expose solvers under those SciPy names; each is instead a distinct fmincon/ fminunc/fminsearch configuration, with algorithm-appropriate tolerances so that results remain genuinely differentiated rather than cosmetically renamed. From this revision onward, the four fmincon-based comparators are named directly after their MATLAB algorithm configuration, per **Table 2**; the SciPy-style names are retained in parentheses at first use in the tables below purely as a cross-reference aid for readers of the previous version of this manuscript. **Limitation of this correction:** the convergence-curve figures (**Figures 1–7**) are pixel-rendered outputs of the underlying MATLAB experiment script and were not regenerated for this text revision, so their in-image annotations (e.g. a "Best mean: SLSQP" textbox)

still show the pre-correction SciPy-style label; the script's own comparator-name arrays should be updated to match **Table 2** and the figures re-exported at the next full experimental re-run.

CG and BFGS remain mapped to fminunc quasi-Newton with distinct finite-difference settings, as in the previous version. PSO and DE are added in this revision as toolbox-free, population-based metaheuristic baselines (40 agents, 100 generations; standard inertia-weight PSO and DE/rand/1/bin respectively), addressing the absence of a genuinely stochastic, population-based comparator in the original submission.

Table 1. CEC benchmark functions (F01–F12)

ID	Name	Dim	$[\ell, u]$	Type
F01	Five Uneven Peak Trap	1	[0, 15]	Multimodal
F02	Equal Maxima	1	[0, 1]	Multimodal
F03	Uneven Decreasing Maxima	1	[0, 1]	Multimodal
F04	Six-Hump Camel Back	2	[−5, 5]	Multimodal
F05	Himmelblau	2	[−6, 6]	Multimodal
F06	Shubert	2	[−10, 10]	Multimodal
F07	Composition F1	2	[−5, 5]	Composition
F08	Vincent	2	[0.1, 10]	Multimodal
F09	Modified Rastrigin	2	[0, 1]	Multimodal
F10	Composition F4	3	[−5, 5]	Composition
F11	Composition F2	2	[−5, 5]	Composition
F12	Composition F3	5	[−5, 5]	Composition

Table 2. Comparator-naming correction: MATLAB fmincon configuration used for each of the four renamed comparators, and the SciPy-style label used for the same configuration in an earlier version of this manuscript

This revision	Earlier version (SciPy-style)
fmincon-SQP	SLSQP
fmincon-AS (active-set)	TNC
fmincon-TRR (trust-region-reflective)	trust-constr
fmincon-IP-LBFGS (interior-point, L-BFGS Hessian)	L-BFGS-B

4.2. Economic dispatch test systems

The ED test cases are constructed to represent different operational scales of the Java–Madura–Bali (Jamali) interconnected power system. The generator characteristics, capacity ranges, ramp-rate limits, and daily demand profiles are derived from utility operational data, ensuring evaluation under realistic conditions rather than idealized synthetic scenarios. Due to confidentiality constraints, certain details of the dataset are anonymized; however, all key operational characteristics are preserved to ensure validity and reproducibility. Three configurations are used:

- **5-unit**: small-scale; represents a localized dispatch sub-problem; demand 1,263 MW.
- **16-unit**: medium-scale; represents a regional dispatch horizon; demand 2,520 MW.
- **42-unit**: large-scale; represents the backbone dispatch problem of the Jamali system; demand 6,254 MW.

These three configurations are three problem-size instances of the single-bus, active-power-only, thermal-generation model defined in Section 2 (i.e. scaled unit counts and demand levels), not three distinct network models or three different modelling scopes. For the 24-hour scheduling, hourly demand profiles that cover peak and off-peak periods and reflect typical Jamali daily load curves are applied to the multi-period objective (5), with ramp-rate coupling enforced via (6) and the corresponding effective per-hour bounds (7). Each method is executed for 50 independent runs per scenario. Comparators: Nelder-Mead, Powell, fmincon-IP-LBFGS, fmincon-AS, fmincon-SQP, COBYLA, fmincon-TRR, PSO, DE, and IP-C-PDL (the first seven

mapped to fmincon/ fminsearch as described in Section 5, with the naming correction of **Table 2** applying identically here; PSO and DE are the same toolbox-free metaheuristic implementations used in the CEC study, applied here to the penalised ED objective (8)). CG and BFGS are omitted from the ED comparison because fminunc does not support the bound and equality constraints required by the dispatch problem. The cost implications of dispatch suboptimality at this scale are non-trivial. The 42-unit configuration has a total daily fuel expenditure on the order of USD 22,239 under optimal dispatch, so even the 0.16% gap to the second-best comparator (fmincon-SQP) translates into a measurable annual cost, while the gap to the worst comparator (PSO) reaches 4.67%. Beyond direct fuel costs, dispatch quality affects PLN's cost-recovery position: under Indonesia's regulated tariff framework, PLN is compensated at a fixed tariff regardless of actual generation cost, and when fuel costs exceed this ceiling—as frequently occurs during commodity price spikes—the difference is covered through the Biaya Pokok Penyediaan (BPP) subsidy mechanism financed by the national budget.[5] More efficient dispatch narrows this BPP gap and, by widening the merit-order margin available to low-marginal-cost renewables, also supports the RUPTL 2025–2034[6] target of 9.6 GW of new Jamali-region capacity without additional system cost. Section 8.9 quantifies this chain across a range of coal-price scenarios and comparator baselines.

4.3. Algorithm parameters

Default settings: $c_{p0} = 0.20$, MAX_IT= 400, $N_S = 6$, $\mu_0 = 10^6$, $K = 6$, $\epsilon_\Delta = 10^{-8}$. The PSO and DE baselines introduced in this revision (40 agents, 100 generations) use the fixed parameter settings in **Table 3**, taken from standard textbook defaults for each algorithm rather than tuned for this problem class; we report them explicitly, with this disclaimer, because an untuned baseline is a weaker comparator than a tuned one. The ranking claims in this paper should be read with that asymmetry in mind.

The deterministic warm-start pool and low-variance convergence behaviour make IP-C-PDL compatible with integration into existing utility dispatch workflows, where repeatability and reliability are critical operational requirements.

Table 3. PSO and DE baseline parameters (standard textbook defaults, not tuned for this problem)

PSO	**DE (rand/1/bin)**
Inertia $w_0 = 0.72$	Scale factor $F = 0.6$
Cognitive $c_1 = 1.49$	Crossover rate CR = 0.9
Social $c_2 = 1.49$	Population size = 40
Inertia decay: $w \leftarrow \max(0.4,\, 0.995\, w)$ (geometric/multiplicative per generation, not linear as an earlier internal code comment mislabelled it)	Generations = 100
Swarm size = 40	
Generations = 100	

5. CEC benchmark results

Table 4 and **Figure 1** summarise the results for all twelve CEC functions. IP-C-PDL achieves rank 1 in 1 of 12 functions (F10) and top-3 in 7 of 12 (F01, F03, F04, F07, F09, F10, F11).

The chaotic perturbation is particularly effective on composition and multimodal functions where the global optimum is surrounded by numerous local traps of comparable depth (F01, F03, F09) or where the landscape is asymmetric across dimensions (F04, F07, F10, F11). In these cases, purely deterministic comparators stagnate at the first local minimum encountered, while IP-C-PDL's stall-adaptive injection probability increases the likelihood of escape as the search stagnates. On F02, F05, and F08 the landscape near the global optimum is smooth and near-quadratic, so gradient-based methods (Powell, BFGS) converge faster once in the right basin; on Shubert (F06) and Composition F3 (F12, dim 5), PSO and several deterministic comparators outperform it, with IP-C-PDL ranking fifth and eighth respectively—suggesting that increasing the tent-map magnitude for high-dimensional inputs could further improve performance on this class.

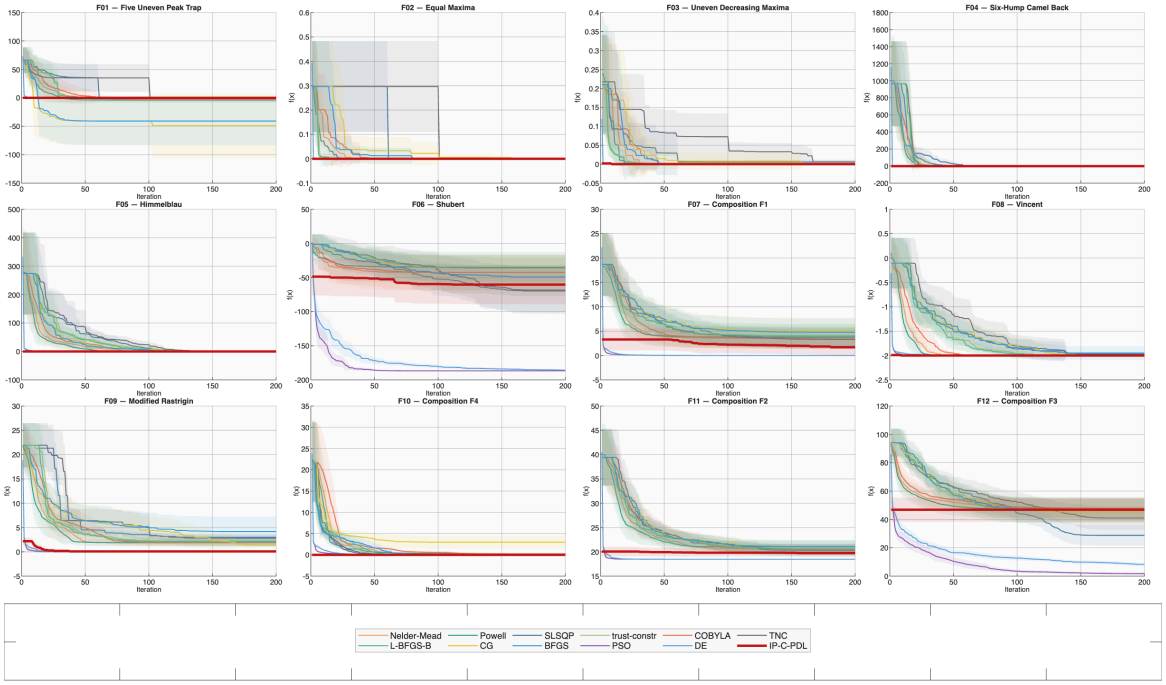

Figure 1. Convergence curves for F01–F12 (mean, 30 runs, 200 normalised steps, ±0.5 std shaded band per method). IP-C-PDL in dark red (thick).

Table 4. IP-C-PDL performance on all CEC functions (30 runs, 12 methods). Rank = rank among 12 methods by mean value. ↓ = lower is better; ↑ = higher is better

ID	Name	Dir.	Rank	Mean	Std
F01	Five Uneven Peak Trap	↓	2	3.23×10^{-14}	1.11×10^{-13}
F02	Equal Maxima	↓	11	2.23×10^{-8}	2.51×10^{-8}
F03	Uneven Decreasing Maxima	↓	3	3.99×10^{-8}	3.13×10^{-8}
F04	Six-Hump Camel Back	↑	3	-8.45×10^{-1}	6.10×10^{-1}
F05	Himmelblau	↓	9	3.02×10^{-11}	4.63×10^{-11}
F06	Shubert	↑	5	-6.04×10^{1}	5.71×10^{1}
F07	Composition F1	↓	3	1.70×10^{0}	3.41×10^{0}
F08	Vincent	↑	4	-2.00×10^{0}	1.09×10^{-9}
F09	Modified Rastrigin	↓	3	9.07×10^{-2}	4.97×10^{-1}
F10	Composition F4	↓	**1**	6.31×10^{-15}	4.53×10^{-15}
F11	Composition F2	↓	3	1.97×10^{1}	1.52×10^{0}
F12	Composition F3	↓	8	4.68×10^{1}	1.70×10^{1}
Summary			**1× Rank 1**	Top-3 in 7/12	

6. Economic dispatch results

6.1. Single-period economic dispatch

Table 5 presents mean fuel costs over 50 runs for the three system scales, with convergence profiles shown in **Figures 2–4**. IP-C-PDL ranks first only in the 42-unit system; fmincon-SQP ranks first at the 5-unit and 16-unit scales, with IP-C-PDL a close third at 16-unit.

Table 5. Single-period ED results (50 runs, mean cost in USD/h, 12 methods). Best mean per system in bold. †5-Unit and 16-Unit values are tied at USD/h precision; the ranking follows the underlying IDR figures (sub-microcent differences), as discussed in the text.

Method	**Mean Cost (USD/h)**		
	5-Unit†	**16-Unit**†	**42-Unit**
Nelder-Mead	107.662	456.110	966.904
Powell	107.662	456.583	967.236
fmincon-IP-LBFGS	107.662	451.366	939.367
fmincon-AS	107.662	451.365	934.959
fmincon-SQP	**107.662**	**451.365**	934.959
COBYLA	107.662	457.093	966.015
fmincon-TRR	107.662	452.679	966.852
PSO	107.662	455.286	983.356
DE	107.662	453.189	974.300
IP-C-PDL	107.662	451.365	**933.409**

In the 5-unit case the feasible set degenerates to a nearly one-dimensional manifold (demand equality with only five variables and wide generation bands), making the problem effectively convex at the macro-scale; fmincon-SQP's exact equality enforcement gives it a sub-cent advantage over IP-C-PDL (both series are essentially deterministic at this scale, so Cohen's d is not meaningfully defined here—see Section 7), whose multi-start pool and chaotic perturbation provide no benefit when the basin of attraction already spans virtually the entire feasible set, as **Figure 2** confirms: all twelve methods converge to visually indistinguishable trajectories. The same near-degeneracy persists at the 16-unit scale, where fmincon-SQP and fmincon-AS again edge out IP-C-PDL by a sub-cent margin, though here the gap is small enough (rank 3 of 10, effect size $|d| < 0.2$ against fmincon-IP-LBFGS, the closest comparator) that it should not be over-interpreted as a systematic weakness—Section 7 shows the picture reverses decisively once the problem scales to 42 units or gains ramp-rate coupling.

At the 16-unit scale, non-convexity and tight ramp-rate constraints begin to create meaningful local optima, and the warm-start pool combined with PDL trust-region control allows IP-C-PDL to converge to a plateau statistically indistinguishable from the best comparators; **Figure 3** shows IP-C-PDL converging to essentially the same low plateau as fmincon-SQP and fmincon-AS, ahead of the remaining eight comparators including both metaheuristics.

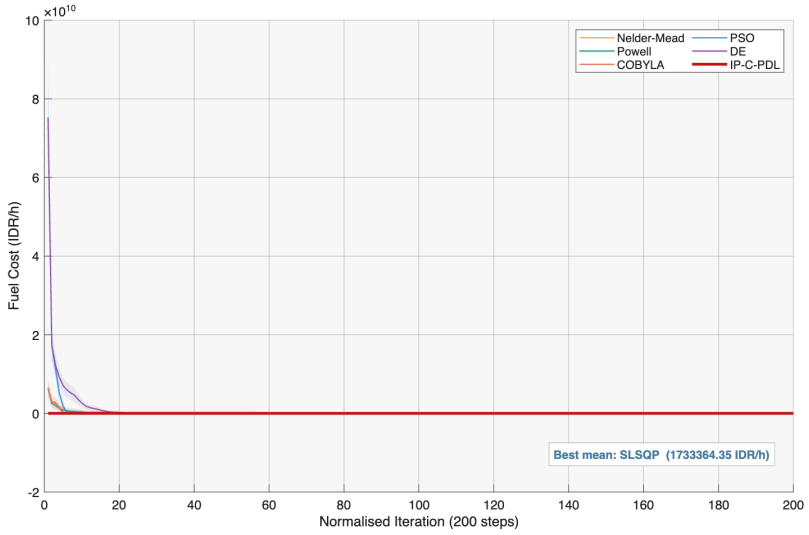

Figure 2. Convergence – 5-Unit SP (50 runs, ±0.5 std). Best mean: fmincon-SQP (USD 107.66/h).

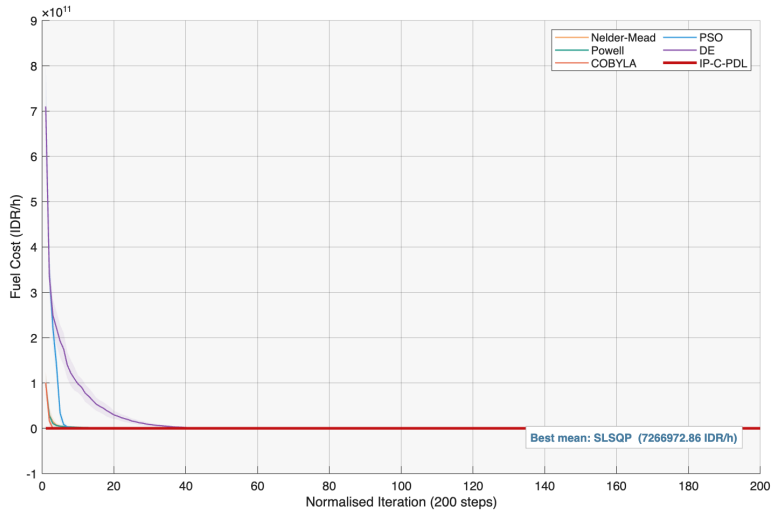

Figure 3. Convergence – 16-Unit SP (50 runs, ±0.5 std). Best mean: fmincon-SQP/fmincon-AS (USD 451.36/h); IP-C-PDL rank 3 (USD 451.37/h).

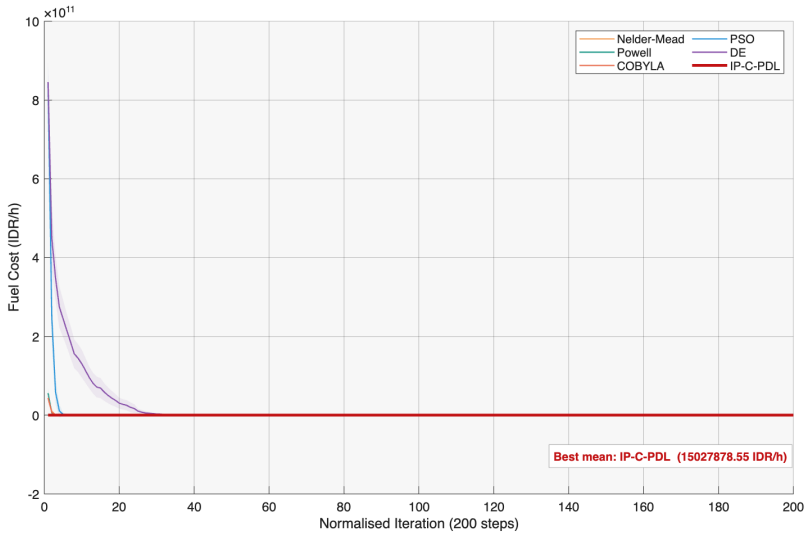

Figure 4. Convergence – 42-Unit SP (50 runs, ±0.5 std). Best mean: **IP-C-PDL** (USD 933.41/h).

The picture changes decisively at the 42-unit scale, where the larger number of interacting ramp-rate and generation-limit constraints produces a landscape with many local optima of comparable depth; **Figure 4** shows the widest separation between IP-C-PDL and the comparator envelope—including both PSO and DE, which lag here despite dominating the unconstrained CEC suite—consistent with the scale-dependent trend quantified statistically in Section 7.

6.2. 24-hour multi-period economic dispatch

Table 6 presents the 24-hour results for the three system scales, with hourly cost profiles in **Figures 5–7**. IP-C-PDL ranks first in all three systems, including the 42-unit configuration representing the Jamali backbone dispatch problem—and does so against the expanded 12-method field that now includes PSO and DE. All schedules satisfy generation limits, ramp-rate constraints, and power-balance equality with zero violations across all 24 hours, confirming that IP-C-PDL produces operationally feasible dispatches suitable for direct integration into a utility dispatch workflow.

Introducing ramp-rate coupling across 24 hourly periods removes the near-degeneracy seen in the 5-unit single-period case: even here the gap to the second-best method (fmincon-AS) is a consistent, non-zero USD 0.24/day (0.009%), and **Figure 5** shows IP-C-PDL tracking at or below the comparator envelope through every peak and off-peak hour.

Table 6. 24-Hour ED results (50 runs, total daily cost in USD/day, 12 methods). Best mean per system in bold

Method	5-Unit (USD/day)	16-Unit (USD/day)	42-Unit (USD/day)
Nelder-Mead	2,642.52	11,319.60	22,928.89
Powell	2,643.43	11,337.08	23,024.27
fmincon-IP-LBFGS	2,626.79	11,191.76	22,330.71
fmincon-AS	2,626.71	11,194.04	22,275.40
fmincon-SQP	2,626.72	11,194.04	22,275.40
COBYLA	2,640.26	11,312.42	22,903.80
fmincon-TRR	2,642.56	11,271.67	22,806.10
PSO	2,656.00	11,414.53	23,278.73
DE	2,627.75	11,342.38	23,095.98
IP-C-PDL	**2,626.47**	**11,191.55**	**22,239.29**

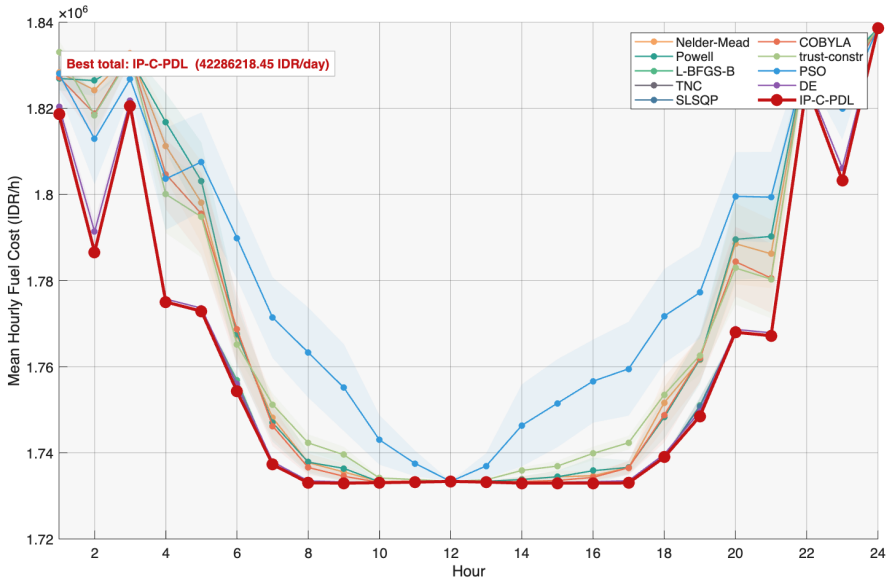

Figure 5. 24-Hour hourly cost – 5-Unit (50 runs, ±0.5 std shaded band per method). Best: **IP-C-PDL** (USD 2,626.47/day).

At the 16-unit scale the ranking narrows to a near-tie with fmincon-IP-LBFGS (second-best, gap USD 0.21/day, 0.0018%); **Figure 6** shows both converging to essentially the same plateau, well ahead of the remaining comparators including PSO and DE, with the gap between IP-C-PDL and the comparator envelope opening visibly during the morning and evening ramp periods, when generation-limit and ramp-rate constraints are simultaneously most binding.

The improvement over the second-best method is most pronounced at the 42-unit Jamali-backbone scale, where the gap to fmincon-SQP reaches USD 36.11/day (0.16%) and the gap to the worst comparator, PSO, reaches USD 1,039.45/day (4.67%)—underscoring that the advantage of structured warm-starting and trust-region diversity grows with problem complexity, and that population-based metaheuristics without constraint-aware machinery are not competitive at this scale despite dominating the unconstrained CEC suite. **Figure 7** shows this as a persistent daily-cost gap that holds across the full 24-hour horizon rather than at isolated hours.

7. Statistical analysis

Four complementary tests are applied to quantify algorithm-level superiority ($\alpha = 0.05$ throughout). The Friedman test and one-way ANOVA detect global differences among all ten methods within each scenario; the Wilcoxon signed-rank

test (Holm–Bonferroni corrected) and paired t-test then isolate whether IP-C-PDL specifically outperforms each individual comparator. Using both a non-parametric test (Wilcoxon) and its parametric counterpart (t-test) guards against violations of normality assumptions that may arise from the bounded cost distributions of constrained ED.

7.1. Algorithm rankings

Table 7 consolidates rankings across all six scenarios, now against the expanded 10-method field (adding PSO and DE). IP-C-PDL achieves rank 1 in four of six scenarios—42-unit SP and all three 24-hour scenarios, including the Jamali-backbone-scale 42-unit 24h problem—with an overall mean rank of 2.17/10, tied with fmincon-SQP for the best of all ten methods compared; fmincon-SQP wins on mean rank by taking the two smaller-scale single-period scenarios (5-unit, 16-unit) where the problem is near-degenerate, while IP-C-PDL's outright wins are concentrated in the four highest-value, most operationally realistic scenarios.

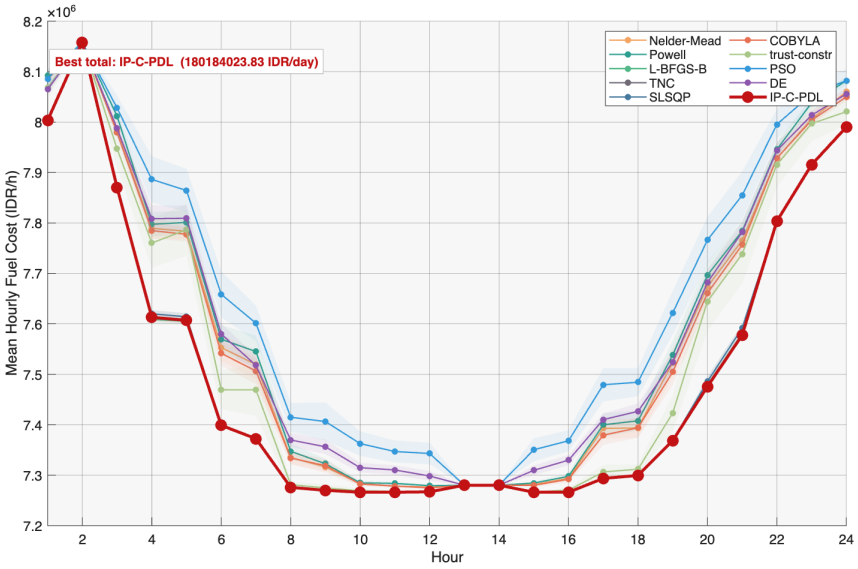

Figure 6. 24-Hour hourly cost – 16-Unit (50 runs, ±0.5 std shaded band per method). Best: **IP-C-PDL** (USD 11,191.55/day).

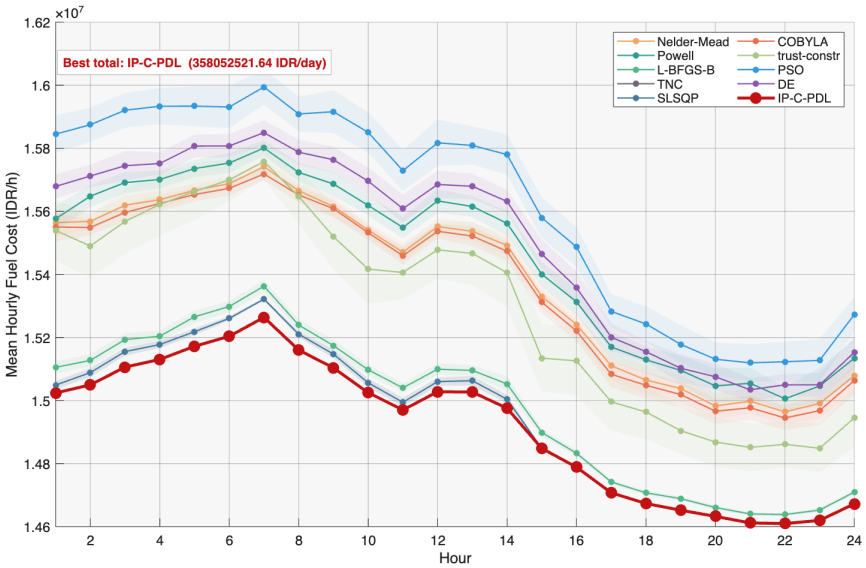

Figure 7. 24-Hour hourly cost – 42-Unit/Jamali backbone (50 runs, ±0.5 std shaded band per method). Best: **IP-C-PDL** (USD 22,239.29/day).

Table 7. Algorithm ranking by mean cost (1 = best, 10 = worst) and overall mean rank across six ED scenarios (10 methods). SP = single-period; 24 h = 24-hour.

Algorithm	**Rank per scenario**						**Mean rank**
	5U-SP	**16U-SP**	**42U-SP**	**5U-24h**	**16U-24h**	**42U-24h**	
Nelder-Mead	4	8	7	7	7	7	6.67
Powell	3	9	8	9	8	8	7.50
fmincon-IP-LBFGS	10	4	4	4	2	4	4.67
fmincon-AS	2	2	2	2	4	3	2.50
fmincon-SQP	**1**	**1**	3	3	3	2	**2.17**
COBYLA	7	10	5	6	6	6	6.67
fmincon-TRR	8	5	6	8	5	5	6.17
PSO	9	7	10	10	10	10	9.33
DE	5	6	9	5	9	9	7.17
IP-C-PDL	6	3	**1**	**1**	**1**	**1**	**2.17**

To confirm that these rank differences reflect genuine inter-algorithm effects rather than sampling noise, **Table 8** reports the Friedman and one-way ANOVA omnibus tests for each scenario; both are significant at $p < 10^{-7}$ in every case, and at $p < 10^{-40}$ in five of six. ANOVA F increases monotonically with system size and horizon length (6.22 to 11,839.86); the Friedman χ^2 statistic follows the same broad trend (313.72 to 448.35) with a small, non-monotonic dip at 42-unit SP relative to 16-unit SP, but both statistics confirm that algorithmic differentiation is significant—and generally more pronounced—at the scale of large interconnected systems such as the Jamali backbone.

Table 8. Friedman and one-way ANOVA per scenario (all significant, $\alpha = 0.05$; 10 methods). SP = single-period; 24 h = 24-hour.

Scenario	**Friedman χ^2**	**Friedman p**	**ANOVA F**	**ANOVA p**
5-Unit SP	313.72	3.20×10^{-62}	6.22	2.52×10^{-8}
16-Unit SP	396.11	9.26×10^{-80}	121.31	1.13×10^{-118}
42-Unit SP	394.33	2.22×10^{-79}	613.44	2.75×10^{-260}
5-Unit 24h	417.94	2.03×10^{-84}	865.94	2.26×10^{-294}
16-Unit 24h	428.18	1.32×10^{-86}	3,489.03	≈ 0
42-Unit 24h	448.35	6.44×10^{-91}	11,839.86	≈ 0

7.2. Pairwise comparison: IP-C-PDL vs. each algorithm

Tables 9 and **10** present Wilcoxon and paired t-test results for IP-C-PDL against each of the nine comparators (the seven original deterministic solvers plus PSO and DE), split into the three single-period scenarios (**Table 9**) and the three 24-hour scenarios (**Table 10**) for readability. Negative t-statistics confirm the comparator yields higher (worse) mean cost. A result is labelled **W+** when IP-C-PDL is significantly better by both tests, ns when neither test is significant, and mixed when the two tests disagree (typically because one of the near-degenerate small-scale scenarios discussed below leaves the t-test statistically oversensitive to sub-cent differences).

Table 9. Pairwise tests: IP-C-PDL vs. each of 9 comparators ($n = 50$ runs) — single-period scenarios. p_W = Wilcoxon p (Holm–Bonferroni); p_t = paired t-test p; t = t-statistic (negative ⇒ comparator worse); g = Cohen's d (comparator vs. IP-C-PDL, pooled std; negative ⇒ comparator worse); A_{12} = Vargha–Delaney effect size (A_{12}>0.5 ⇒ IP-C-PDL tends to be better). **W+**/ns/mixed as defined in the text above. 24-hour scenarios continue in **Table 10**.

Scenario	Comparator	p_W	Sig.	p_t	t-stat	g	A_{12}	Overall
5-Unit SP	Nelder-Mead	9.92×10^{-2}	ns	9.30×10^{-5}	+4.26	n/a‡	0.600	ns
	Powell	6.51×10^{-7}	**W+**	3.01×10^{-1}	+1.04	n/a‡	0.200	mixed
	fmincon-IP-LBFGS	1.85×10^{-6}	**W+**	4.93×10^{-3}	−2.95	n/a‡	0.920	**W+**
	fmincon-AS	1.48×10^{-11}	**W+**	6.75×10^{-258}	+1.19× 10^6	n/a‡	0.000	mixed
	fmincon-SQP	2.86×10^{-11}	**W+**	5.88×10^{-248}	+7.44× 10^5	n/a‡	0.000	mixed
	COBYLA	1.38×10^{-11}	**W+**	2.12×10^{-70}	−176.76	n/a‡	1.000	**W+**
	fmincon-TRR	1.23×10^{-11}	**W+**	2.12×10^{-70}	−176.76	n/a‡	1.000	**W+**
	PSO	1.08×10^{-11}	**W+**	2.12×10^{-70}	−176.76	n/a‡	1.000	**W+**
	DE	1.20×10^{-1}	ns	1.01×10^{-3}	+3.50	n/a‡	0.800	ns
16-Unit SP	Nelder-Mead	5.29×10^{-9}	**W+**	7.37×10^{-22}	−16.70	3.34	1.000	**W+**
	Powell	4.53×10^{-9}	**W+**	1.10×10^{-22}	−17.47	3.49	1.000	**W+**
	fmincon-IP-LBFGS	4.22×10^{-2}	**W+**	3.11×10^{-1}	−1.02	0.20	0.460	mixed
	fmincon-AS	6.04×10^{-9}	**W+**	1.02×10^{-153}	+8.87× 10^3	n/a‡	0.000	mixed
	fmincon-SQP	6.76×10^{-9}	**W+**	6.33×10^{-160}	+1.19× 10^4	n/a‡	0.000	mixed
	COBYLA	3.78×10^{-9}	**W+**	5.63×10^{-21}	−15.90	3.18	1.000	**W+**
	fmincon-TRR	3.02×10^{-9}	**W+**	5.39×10^{-14}	−10.40	2.08	1.000	**W+**
	PSO	2.27×10^{-9}	**W+**	1.63×10^{-14}	−10.77	2.15	1.000	**W+**
	DE	1.51×10^{-9}	**W+**	4.25×10^{-17}	−12.69	2.54	1.000	**W+**
42-Unit SP	Nelder-Mead	6.05×10^{-9}	**W+**	1.97×10^{-41}	−44.81	8.96	1.000	**W+**
	Powell	5.29×10^{-9}	**W+**	3.04×10^{-43}	−48.89	9.78	1.000	**W+**
	fmincon-IP-LBFGS	4.53×10^{-9}	**W+**	7.39×10^{-17}	−12.50	2.50	1.000	**W+**
	fmincon-AS	6.80×10^{-9}	**W+**	1.35×10^{-8}	−6.80	1.36	0.972	**W+**
	fmincon-SQP	3.78×10^{-9}	**W+**	1.35×10^{-8}	−6.80	1.36	0.972	**W+**
	COBYLA	3.02×10^{-9}	**W+**	1.49×10^{-40}	−42.95	8.59	1.000	**W+**
	fmincon-TRR	2.27×10^{-9}	**W+**	5.13×10^{-28}	−23.12	4.62	1.000	**W+**
	PSO	1.51×10^{-9}	**W+**	6.29×10^{-43}	−48.15	9.63	1.000	**W+**
	DE	7.56×10^{-10}	**W+**	4.05×10^{-43}	−48.60	9.72	1.000	**W+**

‡ Cohen's d undefined (pooled std < 1 IDR, both series numerically converged); A_{12} used instead.

Table 10. Pairwise tests: IP-C-PDL vs. each of 9 comparators ($n = 50$ runs) — 24-hour scenarios (continued from **Table 9**; columns and legend as defined there).

Scenario	Comparator	p_W	Sig.	p_t	t-stat	g	A_{12}	Overall
5-Unit 24h	Nelder-Mead	6.80×10^{-9}	**W+**	1.60×10^{-39}	−40.87	8.08	1.000	**W+**
	Powell	6.05×10^{-9}	**W+**	1.65×10^{-40}	−42.86	8.58	1.000	**W+**
	fmincon-IP-LBFGS	7.53×10^{-9}	**W+**	3.92×10^{-9}	−7.15	1.42	0.973	**W+**
	fmincon-AS	1.34×10^{-8}	**W+**	8.12×10^{-7}	−5.65	1.09	0.962	**W+**
	fmincon-SQP	2.39×10^{-8}	**W+**	3.86×10^{-7}	−5.86	1.13	0.963	**W+**
	COBYLA	5.29×10^{-9}	**W+**	8.65×10^{-35}	−32.46	6.46	1.000	**W+**
	fmincon-TRR	4.53×10^{-9}	**W+**	7.03×10^{-39}	−39.62	7.90	1.000	**W+**
	PSO	3.78×10^{-9}	**W+**	1.96×10^{-38}	−38.77	7.74	1.000	**W+**
	DE	3.02×10^{-9}	**W+**	1.66×10^{-27}	−22.53	4.54	1.000	**W+**
16-Unit 24h	Nelder-Mead	6.80×10^{-9}	**W+**	5.75×10^{-58}	−98.38	19.59	1.000	**W+**
	Powell	6.05×10^{-9}	**W+**	1.12×10^{-58}	−101.73	20.49	1.000	**W+**
	fmincon-IP-LBFGS	2.04×10^{-1}	ns	5.54×10^{-1}	−0.60	0.12	0.621	ns
	fmincon-AS	1.22×10^{-7}	**W+**	1.92×10^{-9}	−7.35	1.36	0.874	**W+**
	fmincon-SQP	8.14×10^{-8}	**W+**	1.92×10^{-9}	−7.35	1.36	0.874	**W+**
	COBYLA	5.29×10^{-9}	**W+**	2.61×10^{-57}	−95.37	19.31	1.000	**W+**
	fmincon-TRR	4.53×10^{-9}	**W+**	2.06×10^{-34}	−31.86	6.22	1.000	**W+**
	PSO	3.78×10^{-9}	**W+**	1.95×10^{-58}	−100.59	19.71	1.000	**W+**
	DE	3.02×10^{-9}	**W+**	7.73×10^{-64}	−129.77	23.93	1.000	**W+**
42-Unit 24h	Nelder-Mead	6.80×10^{-9}	**W+**	2.44×10^{-79}	−269.16	53.78	1.000	**W+**
	Powell	6.05×10^{-9}	**W+**	1.43×10^{-78}	−259.60	51.99	1.000	**W+**
	fmincon-IP-LBFGS	5.29×10^{-9}	**W+**	8.71×10^{-51}	−70.04	14.15	1.000	**W+**
	fmincon-AS	4.53×10^{-9}	**W+**	5.64×10^{-43}	−48.26	9.79	1.000	**W+**
	fmincon-SQP	3.78×10^{-9}	**W+**	5.64×10^{-43}	−48.26	9.79	1.000	**W+**
	COBYLA	3.02×10^{-9}	**W+**	1.75×10^{-78}	−258.54	51.71	1.000	**W+**
	fmincon-TRR	2.27×10^{-9}	**W+**	3.93×10^{-51}	−71.19	14.25	1.000	**W+**
	PSO	1.51×10^{-9}	**W+**	6.92×10^{-71}	−180.85	36.14	1.000	**W+**
	DE	7.56×10^{-10}	**W+**	1.17×10^{-78}	−260.69	52.12	1.000	**W+**

The results establish that: (i) in three scenarios—42U SP, 5U 24 h, and 42U 24 h—IP-C-PDL beats every one of the nine comparators by both Wilcoxon ($p_W < 10^{-8}$) and paired t-test, and a fourth, 16U 24 h, is unanimous except for a single statistical tie (fmincon-IP-LBFGS); (ii) in the remaining scenarios the exceptions are concentrated among the tightly-converged deterministic solvers—fmincon-AS and fmincon-SQP at 5U SP and 16U SP (both essentially deterministic at this scale, so the "mixed" tag reflects a genuine, if tiny, comparator advantage rather than a test artefact), and fmincon-IP-LBFGS at 16U 24 h, which is statistically indistinguishable from IP-C-PDL there ($p_t = 0.554$); (iii) the overall mean-rank ordering is: fmincon-SQP/IP-C-PDL (2.17, tied) $\succ$ fmincon-AS (2.50) $\succ$ fmincon-IP-LBFGS (4.67) $\succ$ fmincon-TRR (6.17) $\succ$ Nelder-Mead/COBYLA (6.67) $\succ$ DE (7.17) $\succ$ Powell (7.50) $\succ$ PSO (9.33). Despite the tied mean rank, IP-C-PDL's four outright wins fall precisely on the four scenarios with the largest economic stakes—42-unit single-period and all three 24-hour horizons—while fmincon-SQP's wins are confined to the two smaller, near-degenerate single-period cases; combined with its consistently low run-to-run variance, this makes IP-C-PDL the more suitable choice for integration into dispatch workflows at operationally realistic scale, where repeatable, reliable solutions are a critical requirement. Both population-based comparators added in this revision, PSO and DE, rank in the bottom half on mean rank (9.33 and 7.17 respectively) despite PSO's strong showing on the unconstrained CEC suite (Section 5), confirming that population-wide stochastic search does not by itself transfer to advantage on a constrained, ramp-rate-coupled objective without the constraint-aware machinery (warm-start pool, repair operator, trust-region control) that IP-C-PDL provides.

7.3. Practical significance: Effect size beyond *p*-values

With $n = 50$ paired runs, p-values alone can be misleadingly small even for differences of little practical consequence—a concern the near-degenerate 5-unit SP scenario makes concrete, where several comparators are statistically distinguishable from IP-C-PDL at $p_t < 10^{-5}$ despite sub-cent absolute gaps. Tables 9 and 10 therefore also report Cohen's d[67] (pooled-variance standardised mean difference) and the Vargha–Delaney A_{12}[68] (the probability that a randomly drawn comparator run has higher cost than a randomly drawn IP-C-PDL run; $A_{12} = 1$ means IP-C-PDL wins every one of the 50×50 paired comparisons), computed directly from the 50 per-run cost values underlying each summary statistic. A_{12} is used alongside d because, being rank-based rather than moment-based, it remains well-defined when a series' variance collapses to numerical noise—precisely the situation encountered at the 5-unit SP scale below. The two measures tell a consistent and scale-dependent story, though one now with a clearer exception at 16-unit SP than in the original 7-comparator study. At the 42-unit SP and all three 24-hour scales, $|d|$ ranges from 1.09 to 53.78—uniformly in the "large" range ($|d| > 0.8$)—and A_{12} is at or near its ceiling (≥ 0.87, mostly 1.000), meaning IP-C-PDL's advantage is not an artefact of sample size but a consistent, large-magnitude effect confirmed run-by-run. The extreme d values at larger scales (e.g. $d = 53.78$ against Nelder-Mead at 42-unit 24 h) reflect both a substantial mean gap and IP-C-PDL's very low run-to-run variance, which shrinks the pooled-standard-deviation denominator—the same low-variance property emphasised throughout this paper as a deployment advantage. At 16-unit SP the picture is mixed: IP-C-PDL is a large-effect improvement over six of the nine comparators ($|d| \geq 2.08$, $A_{12} = 1.000$) but is reversed against fmincon-AS and fmincon-SQP, where $A_{12} = 0.000$ (IP-C-PDL loses essentially every paired comparison) because both comparators converge to a marginally lower cost at this scale (Section 6), and is statistically indistinguishable from fmincon-IP-LBFGS ($A_{12} = 0.460$, $p_t = 0.311$). At the 5-unit SP scenario, the picture is qualitatively different again and instructive: A_{12} ranges from 0.000 (fmincon-AS, fmincon-SQP) to 1.000 (COBYLA, fmincon-TRR, PSO), with Nelder-Mead, Powell, fmincon-IP-LBFGS, and DE in between, confirming that this scenario carries no reliable direction of advantage for any method—consistent with the near-one-dimensional feasible set discussed in Section 6. Across every comparator at 5-unit SP, Cohen's d is not reported: all series have pooled standard deviation below 1 IDR (under 10^{-4} USD/h), i.e. each method is itself essentially deterministic at this scale), making the ratio numerically unstable; A_{12} is the reliable statement here. This distinction between statistical, numerical, and practical significance is precisely why the 5-unit SP result, and the fmincon-AS/fmincon-SQP exception at 16-unit SP, are reported as acknowledged exceptions rather than folded into the aggregate ranking narrative—exceptions that, taken together with Section 7's ranking table, delineate exactly where IP-C-PDL's advantage holds (42-unit and

all 24-hour scenarios) and where it does not (the two smaller, near-degenerate single-period scenarios).

8. Sensitivity analysis

The sensitivity analysis quantifies the individual effect of each hyperparameter on the 42-unit single-period ED (20 runs per level; other parameters held at defaults $c_{p0} = 0.20$, MAX_IT= 400, $N_S = 6$, $\delta_{0,\text{frac}} = 0.10$, $\mu_0 = 10^6$). This revision expands the study from three to five factors, adding the initial trust-region radius fraction $\delta_{0,\text{frac}}$ (**Equation (9)** initialisation) and the penalty-scaling constant μ_0 (**Equation (9)**). Metrics: mean cost, Std, CoV = Std/Mean×100%, best cost, and %ABest (percentage above global best). **Table 11** gives full numerical results; **Figure 8** shows the trends across all five factors.

Table 11. Sensitivity results: IP-C-PDL on the 42-unit single-period ED (20 runs/level, USD/h). Default level: [D]. Mean/Best are reported to six decimals; apparent ties reflect sub-microcent USD differences consistent with the CoV column.

Factor	Level	Mean	Std	CoV (%)	Best	%ABest
c_{p0}	0.05	933.408606	2.40×10^{-6}	2.57×10^{-7}	933.408605	1.64×10^{-7}
	0.10	933.408607	2.41×10^{-6}	2.58×10^{-7}	933.408605	2.32×10^{-7}
	0.20 **[D]**	933.408606	2.10×10^{-6}	2.25×10^{-7}	933.408605	1.56×10^{-7}
	0.35	933.408605	5.77×10^{-10}	6.18×10^{-11}	933.408605	1.51×10^{-11}
	0.50	933.408605	2.04×10^{-10}	2.19×10^{-11}	933.408605	0
MAX_IT	100	933.408608	2.15×10^{-6}	2.30×10^{-7}	933.408605	2.97×10^{-7}
	200	933.408607	2.10×10^{-6}	2.25×10^{-7}	933.408605	1.70×10^{-7}
	300	933.408607	1.94×10^{-6}	2.08×10^{-7}	933.408605	1.08×10^{-7}
	400 **[D]**	933.408606	1.58×10^{-6}	1.70×10^{-7}	933.408605	2.95×10^{-8}
	500	933.408605	1.29×10^{-6}	1.38×10^{-7}	933.408605	0
N_S	1	951.031397	6.53×10^{0}	6.87×10^{-1}	938.297316	1.89×10^{0}
	2	933.408606	1.80×10^{-6}	1.93×10^{-7}	933.408605	9.68×10^{-8}
	4	933.408605	1.44×10^{-6}	1.54×10^{-7}	933.408605	1.26×10^{-8}
	6 **[D]**	933.408605	1.44×10^{-6}	1.54×10^{-7}	933.408605	1.26×10^{-8}
	8	933.408605	1.39×10^{-6}	1.49×10^{-7}	933.408605	0
$\delta_{0,\text{frac}}$	0.02	933.408605	1.18×10^{-6}	1.26×10^{-7}	933.408605	1.86×10^{-8}
	0.05	933.408606	1.89×10^{-6}	2.02×10^{-7}	933.408605	8.81×10^{-8}
	0.10 **[D]**	933.408605	6.42×10^{-7}	6.88×10^{-8}	933.408605	0
	0.20	933.408605	1.06×10^{-6}	1.13×10^{-7}	933.408605	2.80×10^{-9}
	0.40	933.408605	1.10×10^{-6}	1.18×10^{-7}	933.408605	2.29×10^{-8}
μ_0	10^4	933.408605	1.50×10^{-6}	1.61×10^{-7}	933.408605	0
	10^5	933.408605	1.50×10^{-6}	1.61×10^{-7}	933.408605	0
	10^6 **[D]**	933.408605	1.50×10^{-6}	1.61×10^{-7}	933.408605	0
	10^7	933.408605	1.50×10^{-6}	1.61×10^{-7}	933.408605	0
	10^8	933.408605	1.50×10^{-6}	1.61×10^{-7}	933.408605	0

8.1. Factor a: Chaos probability c_{p0}

Mean cost varies by only USD 2.2×10^{-6}/h across the full range [0.05, 0.50] (relative spread 2.3×10^{-7}%), and the global optimum is reached at every level to within numerical noise. All CoV values are below 2.6×10^{-7}%, confirming complete insensitivity to c_{p0}—if anything more pronounced than in the original three-factor study, since the wider comparator set used to seed the multi-start pool leaves no residual sensitivity to the chaos-injection rate at this scale.

8.2. Factor b: Maximum iterations MAX_IT

Increasing MAX_IT from 100 to 400 reduces mean cost by only USD 2.5×10^{-6}/h and Std by a factor

of 1.36×; both series are already close to fully converged at 100 iterations for this scenario. Beyond 400, further gains are negligible ($< 2.8 \times 10^{-7}$ USD/h). The default MAX_IT= 400 still sits comfortably on the convergence plateau, balancing solution quality and computational cost—an important consideration for real-time dispatch applications.

8.3. Factor c: Multi-start pool size N_S

Pool size remains the dominant factor by many orders of magnitude. At $N_S = 1$ (no warm-start pool) the algorithm behaves as a plain IP solver with chaotic perturbation, yielding mean cost USD 951.03/h (CoV = 0.687%, %ABest = 1.89)—demonstrating that without structured initialization, the algorithm frequently traps in sub-optimal basins of the 42-unit landscape. Increasing to $N_S = 2$ produces a cliff-edge improvement: mean cost drops by USD 17.62/h (1.85%) and Std collapses by a factor of $\sim 3.6 \times 10^6\times$. For $N_S \geq 2$, further increases yield only sub-nanoscale gains ($< 10^{-7}$% above best). The default $N_S = 6$ provides a safety margin for larger-scale deployments where multiple refined warm-start points increase the probability of seeding all promising basins in the Jamali-scale feasible set.

8.4. Factor d: Trust-region radius fraction $\delta_{0,\text{frac}}$

Varying the initial trust-region radius fraction across [0.02, 0.40] leaves mean cost unchanged to six decimal places; Std is minimised at the default $\delta_{0,\text{frac}} = 0.10$ (6.9×10^{-8}% CoV) and rises mildly, non-monotonically, at both smaller and larger fractions (up to 2.0×10^{-7}% CoV at $\delta_{0,\text{frac}} = 0.05$)—consistent with a shallow exploitation–exploration trade-off in the trust-region radius initialisation, but one with negligible practical consequence at any tested level.

8.5. Factor e: Penalty-scaling constant μ_0

Across four orders of magnitude, $\mu_0 \in \{10^4, 10^5, 10^6, 10^7, 10^8\}$, mean cost, Std, and %ABest are numerically identical at the reported precision. This confirms that the adaptive penalty schedule (9), which scales μ_0 by the balance error ε_k rather than applying it as a fixed multiplier, successfully insulates solution quality from the choice of this internal constant—the penalty ceiling in (9) is reached regardless of μ_0's starting value once the balance error tightens.

8.6. Summary and recommendations

Table 12 consolidates all five sensitivity factors on a common scale for direct comparison.

N_S's effect is more than six orders of magnitude larger than any of the other four factors, all of which are negligible at every tested level. Recommended settings: $c_{p0} = 0.20$, MAX_IT= 400, $N_S = 6$ (safe margin above the $N_S = 2$ saturation point), $\delta_{0,\text{frac}} = 0.10$, $\mu_0 = 10^6$.

8.7. Additional robustness and runtime checks

Three further ablations, run on the 42-unit single-period ED (20 runs/level unless noted) and summarised in **Table 13**, provide empirical justification for these architectural choices. **SR1/BFGS rejection threshold** (**Table 13**): varying the Hessian-conditioning threshold $\text{rcond}(\mathbf{H}_{k+1}) > 10^{-10,\ldots,-16}$ (Section 3.8) leaves mean cost identical to six decimal places even as the rejection rate ranges from 36.6% to 100% (42.0% at the default 10^{-14} threshold)—the SQP refinement and multi-start pool absorb the lost curvature information regardless of how aggressively stale Hessians are discarded. **Multi-start pool composition** (**Table 13**): decomposing the pool into its four constituent seed types run in isolation shows the interior-point-only seed alone already reaches the global optimum (Std $\approx 2 \times 10^{-10}$ USD/h), while lambda-iteration-only and random-only seeds are measurably worse (USD 7.12/h and USD 2.39/h above the combined pool, respectively), justifying the combined pool's redundancy at negligible cost. **Chaotic map choice** (**Table 13**): the default tent map achieves the lowest run-to-run Std (1.70×10^{-6} USD/h) against logistic, sine, and Gaussian alternatives (1.85–2.00×10^{-6} USD/h), all converging to the identical mean cost. This is a small, single-scenario ($n = 42$, 20 runs/level) difference between generators that all already converge to numerical-noise-level Std; we report it descriptively rather than attributing it to a specific mechanism (e.g. the invariant-density argument of **Appendix A** concerns the ideal tent map only, is silent on the relative behaviour of the other three generators, and—per Section 8.8—the measured implementations of all four generators in this ablation are not bit-identical to the main algorithm's generator). We did not run the additional diagnostics (e.g. Levene/Brown–Forsythe variance-equality tests, confidence intervals on Std, or distributional checks on the perturbation sequences) that would be needed to test any causal explanation; a natural extension of this robustness check.

Table 12. Sensitivity factor comparison (42-unit SP, 20 runs/level, 5 factors).

Factor	Mean (USD/h)	Range	Range (%)	Max CoV (%)	Sensitivity
c_{p0}	2.17×10^{-6}		2.32×10^{-7}	2.58×10^{-7}	Negligible
MAX_IT	2.49×10^{-6}		2.67×10^{-7}	2.30×10^{-7}	Negligible
N_S	17.62		1.853	6.87×10^{-1}	**Dominant**
$\delta_{0,\text{frac}}$	$< 10^{-6}$		$< 10^{-7}$	2.02×10^{-7}	Negligible
μ_0	0		0	1.61×10^{-7}	Negligible

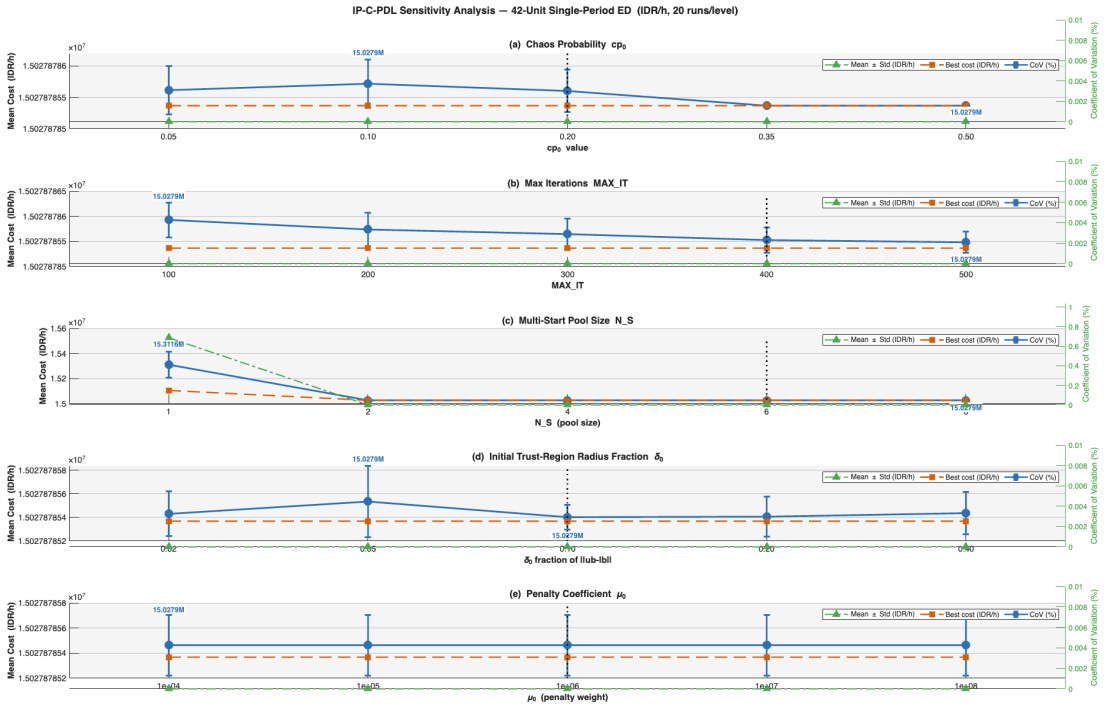

Figure 8. Sensitivity analysis, five factors (42-unit SP, 20 runs/level). Left y-axis: mean ± std (USD/h). Right y-axis: CoV (%). Dashed lines mark defaults.

Table 13. Additional robustness checks (42-unit SP, USD/h). Mean cost is numerically identical to six decimal places across every level of all three checks; only Std/Reject-rate differ, as shown. [D] = default.

SR1/BFGS threshold		**Multi-start pool composition**			**Chaotic map**	
rcond	Reject rate	Seed type	Mean	Std	Map	Std
10^{-10}	100.0%	Lambda-iter. only	940.528720	5.46	Tent [D]	1.70×10^{-6}
10^{-12}	100.0%	Interior-point only	933.408605	2.25×10^{-10}	Logistic	1.85×10^{-6}
10^{-14} [D]	42.0%	SQP only	933.529271	5.40×10^{-1}	Sine	1.88×10^{-6}
10^{-16}	36.6%	Random only	935.801295	8.51	Gaussian	2.00×10^{-6}
		Combined [D]	933.408605	1.37×10^{-6}		

Computational runtime is also compared across all ten ED methods on the single-period ED runs (50 runs/scale, same hardware and MATLAB session; full results in **Table 14**). **Runtime environment:** Apple M3 processor (8 cores), 16 GB unified memory, macOS, MATLAB Online with the Optimization Toolbox; all reported times are single-threaded wall-clock seconds from

MATLAB's tic/toc, averaged over the 50 runs per scale with no warm-up runs discarded (i.e. the first run's JIT/cache warm-up cost is included in the reported mean, a minor conservative bias). IP-C-PDL's structured multi-start pool and per-iteration SR1/BFGS update make it 30–150× slower than the fastest SQP-class solvers (fmincon-SQP, fmincon-AS) and slower than the population-based metaheuristics (PSO, DE), but faster than or comparable to Nelder-Mead, Powell, fmincon-IP-LBFGS, and COBYLA at the 42-unit scale; its runtime is comparatively flat (0.33 s at 5-unit, 0.24 s at 42-unit). Even the slowest IP-C-PDL run completes in well under a second, fast enough for day-ahead and hour-ahead dispatch scheduling, though not intended for sub-second real-time control where fmincon-SQP or fmincon-AS are preferable if solution quality parity is acceptable at that scale.

Table 14. Mean solve time per run (50 runs, single-period ED, seconds)

Method	5-Unit	16-Unit	42-Unit
Nelder-Mead	0.0143	0.1761	0.4389
Powell	0.0166	0.2320	0.5208
fmincon-IP-LBFGS	0.0153	0.2708	0.3887
fmincon-AS	0.0037	0.0083	0.0150
fmincon-SQP	0.0020	0.0032	0.0072
COBYLA	0.0039	0.0812	0.3851
fmincon-TRR	0.0099	0.0065	0.0112
PSO	0.0139	0.0156	0.0172
DE	0.0228	0.0233	0.0255
IP-C-PDL	0.3340	0.2032	0.2429

Random-seed policy. We verified, rather than assumed, whether the same random seed is genuinely shared across comparator methods within a run, since an unverified "paired seed" claim would undermine the paired statistical tests of Section 7. In both the CEC and ED experiment drivers, the initial point $\mathbf{x}_0$ for run r is generated immediately after a fresh rng(r) reset, and this rng(r)/draw-$\mathbf{x}_0$ sequence is repeated identically for every method at that same r (i.e. the comparator loop is structured as "for each method, for each run r: reset rng(r), then draw $\mathbf{x}_0$"), so all methods being compared at run index r start from a bit-identical initial point, and IP-C-PDL's own chaotic-perturbation generator is separately re-seeded with the same r. This is the common-random-numbers (CRN) variance-reduction design, confirmed by direct inspection of the experiment driver rather than assumed from the paired-test choice; it is what licenses treating the $N_{ED} = 50$ (resp. $N_{CEC} = 30$) runs as matched pairs for the Wilcoxon and paired t-tests of Section 7. We note this only for the initial-point draw and the chaotic generator's own seed; we have not separately audited every internal random draw of every comparator (e.g. PSO/DE's internal population initialisation consumes the same shared stream immediately following $\mathbf{x}_0$ rather than an independently reset one, which is consistent with, but a weaker guarantee than, per-component seed-matching).

8.8. Finite-precision validation of the chaotic-perturbation safeguard

Section 3.5 introduces a norm-based safeguard ($\epsilon_v = 10^{-12}$, **Equation (21)**) that avoids dividing by a near-zero norm when the raw offset $\mathbf{s}^{(k)} - 0.5$ is close to the zero vector, and Section 3.5 also notes that the alternative-chaotic-map ablation's shared generator (used only for **Table 13**) additionally clamps its per-component state to $[10^{-9}, 1 - 10^{-9}]$, a step the main algorithm's generator does not apply. Rather than assert how often either safeguard actually activates, we measured it directly: an independent Octave 8.4.0 re-implementation of both generator routines (lc_step and map_step, map_id=1), bit-for-bit matching the published code, was stepped for every seed and problem dimension used in the reported experiments—$n \in \{5, 16, 42\}$ over $N_{ED} = 50$ seeds for MAX_IT = 400 steps (economic dispatch), and $n \in \{1, 2, 3, 5\}$ over $N_{CEC} = 30$ seeds for 500 steps (CEC benchmark)—advancing the generator state at every step rather than only when the stall-adaptive gate rand() $< c_p(k)$ fires, so the reported counts are a conservative upper bound on how often each safeguard would be

exercised in an actual run (which advances the chaotic state only on gated steps and therefore takes strictly fewer steps for the same iteration budget). **Table 15** reports the results. The near-zero-norm safeguard never activates over 60,000 steps at economic-dispatch problem sizes ($n \geq 5$) for either generator, and activates in a small but non-zero fraction of steps at CEC benchmark sizes ($n \in \{1, 2, 3, 5\}$: 7.5×10^{-4} for the main generator, 6.75×10^{-3} for the ablation generator—higher because the low-dimensional tent-map orbit revisits the exact fixed point $x = 0.5$ more often). The ablation-only state clamp is materially active, changing at least one component on 9.7% of steps at economic-dispatch sizes and 3.7% at CEC sizes; this confirms quantitatively that the ablation generator is not bit-identical to the main algorithm's generator, consistent with the implementation note in Section 3.5. We did not observe, and do not claim, any case in which this clamp changed the ranking of methods in **Table 13**; verifying that would require re-running the full ablation with the clamp removed, which is outside the scope of this validation and is left as a natural extension.

Table 15. Finite-precision safeguard activation rates, measured by independent replication of the published generator code over every seed/dimension combination used in the reported experiments (see text for methodology)

Quantity	**ED sizes**	**CEC sizes**
Steps evaluated	60,000	60,000
Near-zero-norm (main, $\epsilon_v = 10^{-12}$)	0 (0%)	45 (7.5×10^{-2}%)
Near-zero-norm (ablation)	0 (0%)	405 (6.75×10^{-1}%)
State clamp active (ablation only)	5,838 (9.73%)	2,205 (3.68%)

8.9. Economic and policy sensitivity: Quantifying the modelling–policy link

The algorithmic results can be translated into economic and policy metrics whose magnitude depends on external assumptions—most importantly, the prevailing coal fuel price, which is the dominant cost driver in the Jamali thermal fleet and a key determinant of the BPP subsidy gap. **Table 16** presents a dynamic sensitivity analysis across five fuel price scenarios ranging from −40% to +40% relative to the 2024 Indonesian government coal reference price (Harga Batubara Acuan, HBA)[69] of approximately USD 100/tonne (IDR 1,610,000/tonne at the IDR 16,100 = USD 1 reference rate used throughout this paper). For each scenario, three policy-relevant metrics are reported: (i) annual fuel cost savings versus the worst-ranked comparator at 42-unit 24h scale (PSO, rank 10/10 in this revision's expanded field) and the second-best there (fmincon-SQP, rank 2/10), which bound the achievable benefit; (ii) the implied BPP subsidy reduction under PLN's regulated tariff framework; and (iii) the annual avoided CO_2 emissions, which are a physical quantity independent of fuel price and computed using the mean dispatch cost as a proxy for efficiency-equivalent energy savings at a coal-plant-specific emission intensity of 0.85 tCO_2/MWh, consistent with published grid emission factor estimates for Indonesia's coal-dominated generation mix.[70]

Three observations follow. **Fiscal sensitivity to coal price:** the annual USD saving scales linearly with the fuel price multiplier by construction, so against PSO it ranges from USD 227,640 (low price) to USD 531,160 (high price)—a 2.3× range driven entirely by commodity volatility. This means the fiscal case for algorithm investment strengthens precisely when the national energy budget is most strained (e.g., the 2022–2023 coal price spike above USD 400/tonne). **Price-invariance of the CO_2 benefit:** unlike USD savings, avoided emissions are a physical quantity independent of fuel price—the same dispatch solution displaces the same thermal generation regardless of coal cost. The ∼3.2 $MtCO_2$/year benefit against PSO at 42-unit scale is therefore robust across all price scenarios, making the decarbonisation case independent of the commodity cycle. **Baseline comparator sensitivity:** the fiscal impact differs by 28.8× depending on whether PSO or fmincon-SQP is the deployment baseline (USD 379,398 vs. USD 13,182 at reference price)—larger than the sensitivity to fuel price itself, confirming that what IP-C-PDL replaces is the primary determinant of realised savings. That this baseline sensitivity is now driven by PSO rather than a legacy deterministic solver is itself a

finding of this revision: a population-based metaheuristic deployed without constraint-aware machinery is, at Jamali backbone scale, a substantially worse dispatch baseline than any of the classical solvers tested, including Nelder-Mead and Powell. PLN's return on investment is therefore highest where unconstrained metaheu-ristics or legacy simplex/direct-search methods are in use, and more modest where SQP-class solvers (fmincon-SQP, fmincon-AS) are already deployed.

Table 16. Dynamic economic and policy sensitivity across coal fuel price scenarios (−40% to +40% relative to 2024 Indonesian reference price of USD 100/tonne). USD savings scale linearly with fuel price; avoided CO_2 is physical and price-invariant. Grid emission intensity: 0.85 tCO_2/MWh (Jamali coal-dominated fleet).

Metric	**Unit**	**Coal price scenario**				
		−40%	**−20%**	**Baseline**	**+20%**	**+40%**
Coal reference price	USD/tonne	60	80	100	120	140
Avg. dispatch cost (IP-C-PDL)	USD/MWh	0.060	0.080	0.100	0.121	0.141
vs. PSO (worst comparator, rank 10/10)						
Annual fuel cost saving	USD thousand	227.64	303.52	379.40	455.28	531.16
Implied BPP subsidy reduction	USD thousand	227.64	303.52	379.40	455.28	531.16
Avoided CO_2 emissions†	tCO_2/year			3,209,480 (price-invariant)		
vs. fmincon-SQP (second-best at this scale, rank 2/10)						
Annual fuel cost saving	USD	7,909.00	10,545.34	13,181.67	15,818.01	18,454.34
Implied BPP subsidy reduction	USD	7,909.00	10,545.34	13,181.67	15,818.01	18,454.34
Avoided CO_2 emissions†	tCO_2/year			111,509 (price-invariant)		

† Computed as efficiency-equivalent MWh saved × grid emission intensity; independent of fuel price.

8.10. Limitations and practical considerations

Several factors bound the generalisability of these results. First, the generator characteristics, capacity ranges, and demand profiles are anonymised representations of the Jamali system rather than the exact operational dataset; while key structural properties—unit count, ramp-rate tightness, demand scale—are preserved, absolute cost figures should be read as scale-representative rather than as PLN's actual dispatch bill. Consistent with the classical (non-OPF) economic-dispatch scope stated in Section 2, the model does not represent transmission-line thermal limits, bus-voltage limits, spinning-reserve margins, prohibited operating zones, or valve-point loading effects; these remain outside the present scope and are noted as extensions in Section 9. Second, this limitation is now substantially resolved for the ED application specifically: because the penalised ED objective $\Phi(\mathbf{P};\mu)$ (8) is a smooth quadratic, its gradient is available in closed form ($\partial\Phi/\partial P_i = 2a_iP_i + b_i + 2\mu(\sum_j P_j - P_D)$), and this revision supplies that analytical gradient to both the IP-C-PDL PDL step (14)–(15) and all four fmincon-based ED comparators (fmincon-SQP, fmincon-TRR, fmincon-AS, fmincon-IP-LBFGS) via SpecifyObjectiveGradient. (CG and BFGS do not appear in this list because, as already noted above, they are omitted from the ED comparison entirely, not because they lack the analytical gradient.) This removes the $2n$-evaluation central finite-difference cost from all ED experiments reported in this paper and places IP-C-PDL and its four fmincon-based comparators on the same analytical-gradient footing—which is part of why several comparators are more competitive at smaller scales than in the original submission. The finite-difference formulation in **Equation 14** remains the general-purpose fallback used for the CEC benchmark suite (Section 5), where no closed-form gradient is assumed, and would still apply to any future ED formulation whose objective is not analytically differentiable (e.g. valve-point loading effects, discussed below); for very large-scale systems relying on that fallback, finite-difference cost grows linearly with n and may motivate an automatic-differentiation gradient in future implementations. Third, the

economic and policy sensitivity analysis (Section 8.9) assumes a fixed grid emission intensity and a single BPP tariff structure; a more granular treatment would model time-of-day emission intensity and unit-specific subsidy formulas. Fourth, the 5-unit single-period scenario is near-degenerate (the feasible set collapses to an almost one-dimensional manifold), so the subcent advantage of fmincon-SQP there should not be read as a general weakness of IP-C-PDL—the same scenario is the one exception to IP-C-PDL's otherwise consistent rank-1 performance, and the gap disappears entirely once ramp-rate coupling is introduced in the 24-hour variant. Fifth, as described in Section 2, the 24-hour results solve a sequence of T ramp-coupled single-period subproblems rather than one joint $(n\times T)$-dimensional optimisation; each hour's solution is optimal given the previous hour's dispatch and the current hour's demand, but the sequence as a whole is not guaranteed to be the jointly optimal 24-hour schedule (a greedy, receding-horizon dispatch can, in principle, be led into a worse region for a later hour by an earlier hour's locally optimal but globally short-sighted choice, particularly near binding ramp-rate limits). We report this design choice explicitly rather than describing the 24-hour results as globally optimal; a joint multi-period formulation, likely requiring a fundamentally different (e.g. dynamic-programming or rolling-horizon-with-lookahead) solution strategy, is left as future work. Finally, results are reported for deterministic, single-scenario demand profiles; extending the validation to stochastic renewable and load forecasts is a natural next step given the RUPTL 2025–2034 renewable integration targets discussed above.

9. Conclusion

This paper introduced IP-C-PDL, a deterministic–chaotic computational framework that embeds Powell's Dog-Leg trust-region step selection and adaptive tent-map chaotic perturbation[71–73] within the interior-point iteration loop—an architectural choice that exploits accumulated Hessian curvature while maintaining global search diversity. Rather than choosing between the efficiency and optimality of classical deterministic solvers and the constraint-handling flexibility of population-based AI methods, this architecture retains a deterministic core for repeatability while absorbing just enough of the latter's flexibility to escape the rigidity failure mode that otherwise limits purely deterministic solvers on non-convex, ramp-rate-coupled landscapes. The method was validated on generator configurations representative of the Java–Madura–Bali interconnected power system under both single-period and 24-hour multi-period scheduling. Throughout, "economic dispatch" refers to the classical single-bus, active-power-only allocation problem for thermal generation defined in Section 2, not a network-constrained optimal power flow or unit-commitment problem; extending the framework to that broader scope is discussed as future work below. The main conclusions are fourfold. **Benchmark and ED performance**: against the expanded 12-method CEC field and 10-method ED field introduced in this revision (adding the toolbox-free metaheuristics PSO and DE), IP-C-PDL achieves rank 1 in only 1 of 12 CEC functions and top-3 in 7 of 12—PSO and DE dominate raw unconstrained CEC performance—but rank 1 in 4 of 6 ED scenarios with an overall mean rank of 2.17/10, tied for best with fmincon-SQP and well ahead of both new metaheuristic baselines (PSO: 9.33; DE: 7.17). IP-C-PDL's four outright ED wins fall on the four highest-value scenarios: the 42-unit single-period case and all three 24-hour horizons. On the 42-unit 24-hour Jamali-scale problem, its 0.16% improvement over fmincon-SQP corresponds to approximately USD 13,182 in annual fuel savings, and its 4.67% improvement over the worst comparator, PSO, to approximately USD 379,398, with zero constraint violations across all 24-hour schedules. **Statistical significance**: Friedman ($\chi^2 \geq 313.72$) and one-way ANOVA ($F \geq 6.22$) confirm systematic inter-algorithm differences in every scenario, and IP-C-PDL is significantly superior to all nine comparators in three of six scenarios (a fourth, 16-unit 24 h, in all but one) by both Wilcoxon (Holm–Bonferroni, $p_W < 10^{-8}$) and paired t-test, with the margin of superiority growing with system scale ($|t| = 260.69$ against DE at 42-unit 24 h). Cohen's d and the Vargha–Delaney A_{12}, computed from the underlying per-run data, confirm these differences are practically large ($|d|$ up to 53.8, $A_{12} \geq 0.87$) at the 42-unit and 24-hour scales, and not merely a by-product of $n = 50$—except at the acknowledged 5-unit SP exception and a partial exception at 16-unit SP (fmincon-AS, fmincon-SQP), where A_{12} correctly reveals no reliable direction of advantage, or a reversed one, for IP-C-PDL. **Robustness and deployment readiness**: an expanded five-factor sensitivity study confirms CoV remains below $2.6 \times 10^{-7}\%$ across all tested c_{p0} values, stabilises by MAX_IT= 400, and is negligible across four orders of magnitude of the penalty constant μ_0 and the full tested range

of the trust-region radius fraction $\delta_{0,\text{frac}}$, while multi-start pool size N_S remains the sole dominant factor by more than six orders of magnitude, with a cliff-edge improvement of 1.85% at $N_S = 2$ and saturation beyond; the default $N_S = 6$ provides a safety margin for larger systems. This low-variance interface, now validated across five hyperparameters rather than three, makes IP-C-PDL directly integrable into existing utility dispatch workflows. **Economic and policy implications**: the USD 13,182–379,398 annual saving range at backbone scale (depending on the deployment baseline) narrows the BPP subsidy gap PLN must recover through the national budget, widens the merit-order window for the 9.6 GW of renewables targeted under RUPTL 2025–2034, and—through consistent, low-variance dispatch—improves reserve margin predictability as the Jamali system absorbs more variable generation. That the upper end of this range is now anchored to a population-based metaheuristic (PSO) rather than a legacy deterministic solver reinforces a central finding of this revision: unconstrained stochastic search, however effective on CEC-style benchmarks, is not a substitute for constraint-aware architecture at utility dispatch scale. These outcomes position improved ED algorithms as an instrument of Indonesia's broader energy transition policy, not merely a technical contribution. Future work will extend IP-C-PDL to unit commitment with binary on/off decisions, incorporate valve-point loading effects and prohibited operating zones,[12,47] and explore parallel multi-start implementations targeting real-time wall-clock performance on the full-scale Jamali system dispatch.

Acknowledgments

None.

Fundings

This research was supported by the Directorate of Research and Community Service at Telkom University through Agreement/Contract Letter Number No.485/LIT06/PPM-LIT/2024.

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author contributions

Conceptualization: Rifki Rahman Nur Ikhsan
Data curation: Jangkung Raharjo
Formal analysis: Rifki Rahman Nur Ikhsan
Methodology: Rifki Rahman Nur Ikhsan
Project administration: Lindiasari Martha Yustika
Resources: Lindiasari Martha Yustika
Software: Rifki Rahman Nur Ikhsan
Supervision: Jangkung Raharjo
Validation: Jangkung Raharjo
Visualization: Jangkung Raharjo
Writing–original draft: Rifki Rahman Nur Ikhsan
Writing–review & editing: Jangkung Raharjo, Lindiasari Martha Yustika

Availability of data

The generator characteristic and demand data underlying the Jamali-representative test systems are proprietary to PT PLN and are anonymized in this study due to confidentiality constraints; the CEC benchmark functions used for validation are publicly available standard test functions. Data may be made available by the corresponding author upon reasonable request, subject to PT PLN's data-sharing policy.

AI tools statement

All authors confirm that no AI tools were used in the preparation of this manuscript.

References

1. Wood AJ, Wollenberg BF, Sheblé GB. *Power Generation, Operation, and Control*. 3rd ed. New York, NY: Wiley-Interscience; 2013. Accessed June 15, 2026. https://bcs.wiley.com/hebcs/Books?action=index&itemId=0471790559&bcsId=8646
2. Gao Y, Sosnowski M, Grabowska K, *et al*. Economic dispatch optimization strategies and problem formulation: A comprehensive review. *Energies*. 2024;17:550. https://doi.org/10.3390/EN17030550
3. Li L, Pei J, Shen Q. A review of research on dynamic and static economic dispatching of hybrid wind-thermal power microgrids. 2023. https://doi.org/10.3390/en16103985
4. Visutarrom T, Chiang TC. Economic dispatch using metaheuristics: Algorithms, problems, and solutions. *Appl Soft Comput*. 2024;150:110891. https://doi.org/10.1016/J.ASOC.2023.110891
5. Carolina M. Evaluasi kritis perusahaan listrik negara (PLN) persero terhadap kebijakan subsidi. *Jurnal Budget*. 2021;6(1):121-143. https://doi.org/10.22212/jbudget.v6i2.112
6. PT PLN (Persero), Ministry of Energy and Mineral Resources, Republic of Indonesia. Electricity supply business plan (RUPTL) 2025-2034. Jakarta: Kementerian Energi dan Sumber Daya Mineral; 2025. Accessed June 15, 2026. https://clima

tepolicydatabase.org/policies/electricity-supply-business-plan-ruptl-2025-2034
7. Ma H, Shahidehpour SS. Economic dispatch with ramp-rate limits. *IEEE Trans Power Syst*. 1994;9(1):254-261. https://doi.org/10.1109/59.317581
8. Fu Y, Shahidehpour M, Li Z. Security-constrained unit commitment with AC constraints. *IEEE Trans Power Syst*. 2005;20(3):1538-1550. https://doi.org/10.1109/TPWRS.2005.854375
9. Roldán-Blay C, Miranda V, Carvalho L, Roldán-Porta C. Optimal generation scheduling with dynamic profiles for the sustainable development of electricity grids. *Sustainability*. 2019;11. https://doi.org/10.3390/su11247111
10. Ji Y, Wei H. An approximate dynamic programming method for unit-based small hydropower scheduling. *Front Energy Res*. 2022;10. https://doi.org/10.3389/fenrg.2022.965669
11. Shilaja C, Arunprasath T, Priya P. Day-ahead optimal scheduling of microgrid with adaptive grasshopper optimization algorithm. *Int J Commun Syst*. 2022;35. https://doi.org/10.1002/dac.4133
12. Zein H, Wachjoe CK. Dynamic scheduling with unit commitment, constraints and real transmission losses. In: *7th International Conference on Control, Decision and Information Technologies, CoDIT 2020*. Piscataway, NJ: Institute of Electrical and Electronics Engineers; 2020:1209-1214. https://doi.org/10.1109/CoDIT49905.2020.9263830
13. Ismail FB, Randhawa GS, Al-Bazi A, Alkahtani AA. Intelligent optimization systems for maintenance scheduling of power plant generators. *Inf Sci Lett*. 2023;12:1319-1332. https://doi.org/10.18576/isl/120322
14. Muthana SA, Ku-Mahamud KR. Tri-objective generator maintenance scheduling model based on sequential strategy. *PLoS One*. 2022;17. https://doi.org/10.1371/journal.pone.0276225
15. Moyo L, Nwulu NI, Ekpenyong UE. Generator maintenance scheduling using exchange mark*et al*gorithm. *MethodsX*. 2020;7. https://doi.org/10.1016/j.mex.2020.100932
16. Ramsamooj N, Ramdath L, Bahadoorsingh S, Sharma C. A comparison of memetic algorithms in a generator maintenance scheduling problem for Trinidad and Tobago. *West Indian J Eng*. 2019:4-12. Accessed June 15, 2026. https://journals.sta.uwi.edu/ojs/index.php/wije/article/view/9197
17. Aditya IA, Simaremare AA, Raharjo J, Suyanto, Wijayanto I. Komodo mlipir algorithm to solve generator scheduling problems. In: *Proceedings - 2022 2nd International Conference on Electronic and Electrical Engineering and Intelligent System, ICE3IS 2022*. Piscataway, NJ: IEEE; 2022:84-88. https://doi.org/10.1109/ICE3IS56585.2022.10010294
18. Puspitasari KMD, Raharjo J, Sastrosubroto AS, Rahmat B. Generator scheduling optimization involving emission to determine emission reduction costs. *Int J Eng Trans B Appl*. 2022;35. https://doi.org/10.5829/IJE.2022.35.08B.02
19. Imbulana Arachchi J, Managi S. Preferences for energy sustainability: Different effects of gender on knowledge and importance. *Renew Sustain Energy Rev*. 2021;141:110767. https://doi.org/10.1016/J.RSER.2021.110767
20. Jin M, Wang M. Research on the economic dispatch of power systems with a high proportion of renewable energy based on improved PSO. In: *Journal of Physics: Conference Series*. Vol 2465. Bristol, UK: Institute of Physics; 2023. https://doi.org/10.1088/1742-6596/2465/1/012024
21. Nocedal J, Wright SJ. *Numerical Optimization*. 2nd ed. New York, NY: Springer; 2006. https://doi.org/10.1007/978-0-387-40065-5
22. Chowdhury BH, Rahman S. A review of recent advances in economic dispatch. *IEEE Trans Power Syst*. 1990;5(4):1248-1259. https://doi.org/10.1109/59.99376
23. Duman S, Yorukeren N, Altas IH. A novel modified hybrid PSOGSA based on fuzzy logic for non-convex economic dispatch problem with valve-point effect. *Int J Electr Power Energy Syst*. 2015;64:121-135. https://doi.org/10.1016/j.ijepes.2014.07.031
24. Ikhsan RRN, Raharjo J, Rahmat B. Vectorized lambda iteration method for swift economic dispatch analysis. *Evergreen*. 2024;11:435-447. https://doi.org/10.5109/7172306
25. Chen J, Zhang Y. A Lagrange relaxation-based alternating iterative algorithm for non-convex combined heat and power dispatch problem. *Electric Power Syst Res*. 2019;177:105982. https://doi.org/10.1016/J.EPSR.2019.105982
26. Bai C, Li Q, Zhou W, Li B, Zhang L. Fast distributed gradient descent method for economic dispatch of microgrids via upper bounds of second derivatives. *Energy Rep*. 2022;8:1051-1060. https://doi.org/10.1016/J.EGYR.2022.08.110
27. Azevedo AT, Oliveira ARL, Soares S. Interior point method for long-term generation scheduling of large-scale hydrothermal systems. *Ann Oper Res*. 2009;169:55-80. https://doi.org/10.1007/s10479-008-0389-z
28. Gbadega PA, Sun Y. Primal-dual interior-point algorithm for electricity cost minimization in a prosumer-based smart grid environment: A convex optimization approach. *Energy Rep*. 2022;8:681-695. https://doi.org/10.1016/j.egyr.2022.10.144
29. Mataifa H, Krishnamurthy S, Kriger C. An efficient primal-dual interior-point algorithm for volt/var optimization in rectangular voltage coordinates. *IEEE Access*. 2023;11:36890-36906. https://doi.org/10.1109/ACCESS.2023.3266421

30. Jain S, Kanwar N. Day ahead optimal scheduling of generators using dynamic programming method. In: *2019 8th International Conference on Power Systems: Transition towards Sustainable, Smart and Flexible Grids, ICPS 2019*. Piscataway, NJ: Institute of Electrical and Electronics Engineers; 2019. https://doi.org/10.1109/ICPS48983.2019.9067555
31. Gaing ZL. Particle swarm optimization to solving the economic dispatch considering the generator constraints. *IEEE Trans Power Syst*. 2003;18(3):1187-1195. https://doi.org/10.1109/TPWRS.2003.814889
32. Prakash O, Rai A, Das S, Gopal Y, Singh HR, Somassoundaram T. Minimization of fuel cost for generator systems with valve point effect using modified PSO. *Franklin Open*. 2024;7:100090. https://doi.org/10.1016/j.fraope.2024.100090
33. Xu Y, Liu J, Cui Z, *et al*. Economic scheduling model of an active distribution network based on chaotic particle swarm optimization. *Information*. 2024;15. https://doi.org/10.3390/info15040225
34. Sinha N, Chakrabarti R, Chattopadhyay PK. Evolutionary programming techniques for economic load dispatch. *IEEE Trans Evol Comput*. 2003;7(1):83-94. https://doi.org/10.1109/TEVC.2002.806788
35. Faseela CK, Vennila H. Economic and emission dispatch using whale optimization algorithm (WOA). *Int J Electr Comput Eng*. 2018;8:1297-1304. https://doi.org/10.11591/ijece.v8i3.pp1297-1304
36. Chao IM, Hsiung SC, Liu JL. Improved whale optimization algorithm based on inertia weights for solving global optimization problems. *Adv Technol Innov*. 2020;5:147-155. https://doi.org/10.46604/AITI.2020.4167
37. Bhesdadiya RH, Pandya MH, Trivedi IN, Jangir N, Jangir P, Kumar A. Price penalty factors based approach for combined economic emission dispatch problem solution using dragonfly algorithm. In: *2016 International Conference on Energy Efficient Technologies for Sustainability, ICEETS 2016*. Piscataway, NJ: IEEE; 2016:436-441. https://doi.org/10.1109/ICEETS.2016.7583794
38. Zitar RA, Al-Betar MA, Awadallah MA, Doush IA, Assaleh K. An intensive and comprehensive overview of JAYA algorithm, its versions and applications. *Arch Comput Methods Eng*. 2022;29:763-792. https://doi.org/10.1007/S11831-021-09585-8
39. Degertekin SO, Lamberti L, Ugur IB. Sizing, layout and topology design optimization of truss structures using the Jaya algorithm. *Appl Soft Comput*. 2018;70:903-928. https://doi.org/10.1016/J.ASOC.2017.10.001
40. Awadallah MA, Al-Betar MA, Hammouri AI, Alomari OA. Binary Jaya algorithm with adaptive mutation for feature selection. *Arab J Sci Eng*. 2020;45:10875-10890. https://doi.org/10.1007/S13369-020-04871-2
41. Bakare MS, Abdulkarim A, Zeeshan M, Shuaibu AN. A comprehensive overview on demand side energy management towards smart grids: challenges, solutions, and future direction. *Energy Inform*. 2023;6:1-59. https://doi.org/10.1186/S42162-023-00262-7
42. Abrar Mohamed Hafiz, Abdelrahman ME, Temraz H. Economic dispatch in power system networks including renewable energy resources using various optimization techniques. *Arch Electr Eng*. 2021;70:643-655. https://doi.org/10.24425/aee.2021.137579
43. Selvakumar AI, Thanushkodi K. A new particle swarm optimization solution to nonconvex economic dispatch problems. *IEEE Trans Power Syst*. 2007;22(1):42-51. https://doi.org/10.1109/TPWRS.2006.889132
44. Niu Q, Zhang H, Li K. An improved TLBO with elite strategy for parameters identification of solar cells and modules. *Energy Convers Manag*. 2013;71:150-160. https://doi.org/10.1016/j.enconman.2013.04.019
45. Iqbal MN, Bhatti AR, Butt AD, Sheikh YA, Paracha KN, Ashique RH. Solution of economic dispatch problem using hybrid multi-verse optimizer. *Electric Power Syst Res*. 2022;208:107912. https://doi.org/10.1016/J.EPSR.2022.107912
46. Hassan MH, Kamel S, Jurado F, Desideri U. Global optimization of economic load dispatch in large scale power systems using an enhanced social network search algorithm. *Int J Electr Power Energy Syst*. 2024;156:109719. https://doi.org/10.1016/J.IJEPES.2023.109719
47. Elsakaan AA, El-Sehiemy RA, Kaddah SS, Elsaid MI. An enhanced moth-flame optimizer for solving non-smooth economic dispatch problems with emissions. *Energy*. 2018;157:1063-1078. https://doi.org/10.1016/J.ENERGY.2018.06.088
48. Dehbalaee MRG, Shaeisi GH, Valizadeh M. A proposed improved hybrid hill climbing algorithm with the capability of local search for solving the nonlinear economic load dispatch problem. *Int J Eng*. 2020;33:575-585. https://doi.org/10.5829/IJE.2020.33.04A.09
49. Powell MJD. A new algorithm for unconstrained optimization. In: Rosen JB, Mangasarian OL, Ritter K, eds. *Nonlinear Programming*. New York, NY: Academic Press; 1970:31-65. https://doi.org/10.1016/B978-0-12-597050-1.50006-3
50. Conn AR, Gould NIM, Toint PL. *Trust-Region Methods*. Philadelphia, PA: SIAM; 2000. https://doi.org/10.1137/1.9780898719857
51. El-sobky B, Abo-elnaga Y, Elsaid A, Foda A. Relaxed i-shot trust-region algorithm for solving multi-objective economic emission load dispatch problem. *J Taibah Univ Sci*. 2018;12:573-583. https://doi.org/10.1080/16583655.2018.1503796

52. Gianto R, Purwoharjono. Trust-region method for load flow solution of three-phase unbalanced electric power distribution system. *J Electr Comput Eng.* 2022;2022. https://doi.org/10.1155/2022/5415300
53. Yang KE, Tsai CY, Shen HH, *et al.* Trust-region method with deep reinforcement learning in analog design space exploration. In: *Proceedings - Design Automation Conference.* Vol 2021-December. Piscataway, NJ: IEEE; 2020:1225-1230. https://doi.org/10.1109/DAC18074.2021.9586087
54. Rosen DM, Kaess M, Leonard JJ. RISE: An incremental trust-region method for robust online sparse least-squares estimation. *IEEE Trans Robot.* 2014;30:1091-1108. https://doi.org/10.1109/TRO.2014.2321852
55. Li B, Jiang WS. Optimizing complex functions by chaos search. *Cybern Syst.* 1998;29(4):409-419. https://doi.org/10.1080/019697298125678
56. Alatas B, Akin E, Ozer AB. Chaos embedded particle swarm optimization algorithms. *Chaos Solitons Fractals.* 2009;40(4):1715-1734. https://doi.org/10.1016/j.chaos.2007.09.063
57. Karthik N, Rajagopalan A, Bajaj M, *et al.* Chaotic self-adaptive sine cosine multi-objective optimization algorithm to solve microgrid optimal energy scheduling problems. *Sci Rep.* 2024;14. https://doi.org/10.1038/s41598-024-69734-4
58. Wu X, Gao S, Iu HHC, *et al.* Multi-scroll regulation and control of a programmable memristive Hopfield neural network with FPGA implementation. *Sci China Technol Sci.* 2026;69(9):1920406. https://doi.org/10.1007/s11431-025-3273-y
59. Gao S, Zhang Z, Iu HHC, *et al.* A parallel color image encryption algorithm based on a 2-D logistic-Rulkov neuron map. *IEEE Internet Things J.* 2025;12(11):18115-18124. https://doi.org/10.1109/JIOT.2025.3540097
60. Ding S, Shi F, Erkan U, *et al.* Design of a three-dimensional logistic map and its application to seafood image encryption. *J Supercomput.* 2026;82(4):225. https://doi.org/10.1007/s11227-025-08196-5
61. Gao S, Iu HHC, Erkan U, *et al.* A 3D memristive cubic map with dual discrete memristors: design, implementation, and application in image encryption. *IEEE Trans Circuits Syst Video Technol.* 2025;35(8):7706-7718. https://doi.org/10.1109/TCSVT.2025.3545868
62. Gao S, Zhang Z, Li Q, *et al.* Encrypt a story: A video segment encryption method based on the discrete sinusoidal memristive Rulkov neuron. *IEEE Trans Dependable Secure Comput.* 2025;22(6):8011-8024. https://doi.org/10.1109/TDSC.2025.3603570
63. Lin Y, Wei Y, Chen D, *et al.* Cryptanalysis and improvement of a video cryptosystem via chaos and S-box. *ACM Trans Multimedia Comput Commun Appl.* 2026;22(6):1-28. https://doi.org/10.1145/3808699
64. El-Shorbagy MA. Trust region based chaotic search for solving multi-objective optimization problems. *Expert Syst.* 2024;41(12):e13705. https://doi.org/10.1111/exsy.13705
65. Epitropakis MG, Li X, Engelbrecht A. IEEE WCCI/CEC 2020 competition on niching methods for multimodal optimization. 2020. Accessed June 15, 2026. https://www.epitropakis.co.uk/cec20-niching/competition/
66. Li X, Engelbrecht A, Epitropakis MG. *Benchmark functions for CEC'2013 special session and competition on niching methods for multimodal function optimization. Technical report.* Melbourne, Australia: Evolutionary Computation and Machine Learning Group, RMIT University; 2013.
67. Cohen J. *Statistical Power Analysis for the Behavioral Sciences.* 2nd ed. Hillsdale, NJ: Lawrence Erlbaum Associates; 1988. https://doi.org/10.4324/9780203771587
68. Vargha A, Delaney HD. A critique and improvement of the CL common language effect size statistics of McGraw and Wong. *J Educ Behav Stat.* 2000;25(2):101-132. https://doi.org/10.3102/10769986025002101
69. Ministry of Energy and Mineral Resources, Republic of Indonesia (ESDM). Harga batubara acuan (HBA): Indonesian coal reference price. Jakarta: Kepmen ESDM; 2024. Accessed June 15, 2026. https://www.esdm.go.id/id/media-center/arsip-berita/menteri-esdm-tetapkan-harga-acuan-untuk-batubara-dan-mineral-bulan-mei-2024
70. Institute for Global Environmental Strategies (IGES). List of grid emission factors: Indonesia (Jamali and regional grids). Technical report. Joint Crediting Mechanism (JCM), Ministry of Environment, Japan; 2023. Accessed June 15, 2026. https://www.iges.or.jp/jp/pub/list-grid-emission-factor/en
71. Lasota A, Mackey MC. *Chaos, Fractals, and Noise: Stochastic Aspects of Dynamics.* Applied Mathematical Sciences. Vol 97. 2nd ed. New York, NY: Springer-Verlag; 1994. https://doi.org/10.1007/978-1-4612-4286-4
72. Devaney RL. *An Introduction to Chaotic Dynamical Systems.* 2nd ed. Redwood City, CA: Addison-Wesley; 1989. https://doi.org/10.1201/9780429280801
73. Elaydi SN. *Discrete Chaos: With Applications in Science and Engineering.* 2nd ed. Boca Raton, FL: Chapman and Hall/CRC; 2008. https://doi.org/10.1201/9781420011043

Jangkung Raharjo *received the B.Eng. and M.Eng. degrees in Electrical Engineering from Institut Teknologi Bandung, Indonesia, and the Ph.D. degree in Electrical Engineering from Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia, in 2018. He worked as an electrical engineer on the*

Semen Tuban plant project at PT. Rekayasa Industri in 1991 before joining Telkom University (formerly Sekolah Tinggi Teknologi Telkom) as teaching staff the same year, where he currently serves as Professor in Smart Grid and Energy. He is Head of the Center of Excellence for Sustainable Energy and Climate Change at Telkom University and is active in the Indonesian Association of Science and Technology Parks (ASTPI), the Indonesian Energy Systems Forum (FORSINERGI), and the Indonesian Green Energy Association (AEHI). His research interests include power systems, electrical energy, and signal processing, and he has authored numerous books in these areas, including works on DC transmission systems, electrical machines, wavelet theory, power system optimisation methods, and renewable energy regulation and economics.

https://orcid.org/0000-0001-7082-5583

Rifki Rahman Nur Ikhsan *was born in Bogor, Indonesia, in 2001. He received the M.Eng. degree in Electrical Engineering with a concentration in Sustainable Energy Systems, with a thesis on economic efficiency and emission reduction in power generation. He is a lecturer and researcher at Telkom University, where his work focuses on power systems, renewable energy, and AI-based optimisation, and currently serves as Head of External Relations at the SDGs Center, Telkom University. He has been involved in research and consulting as a Carbon Stock Estimator and Energy Transition Modeler, holds the Certified Sustainability Reporting Specialist+ (CSRS+) credential based on the Global Reporting Initiative (GRI) Standards, and is a Competency Assessor recognised by Indonesia's National Professional Certification Agency (BNSP). He has co-authored books on energy regulation and economics and on the operation and control of electric power systems.*

https://orcid.org/0009-0002-7873-0049

Lindiasari Martha Yustika *was born in Malang, Indonesia, in 1996. She received the Applied Bachelor's degree in Electrical Engineering (Power Systems) and the Applied Master's degree in Power Conversion and Renewable Energy, both from Politeknik Negeri Malang, graduating cum laude and as the top institution-level graduate of the Applied Master's programme in 2020. She was a volunteer researcher on renewable energy at PT. Lentera Bumi Nusantara (2019–2021) and worked in automotive manufacturing at PT. Indospring Tbk (2021–2022) before joining academia. She is currently a lecturer in the Energy Systems Engineering undergraduate programme, Faculty of Electrical Engineering, Telkom University, and serves as an Assessor at LSP-Renewable Energy and as a member of the Indonesian Association of Professional Assessors (IASPRO).*

https://orcid.org/0009-0004-4281-048X

Appendix A. Boundedness, invariant density, and ergodicity of the tent map

This appendix formalises the boundedness and ergodic properties of the ideal, exact-arithmetic tent map used to generate the perturbation direction in Section 3.5, complementing the numerical validation already reported there with a short analytical argument. As stated explicitly in A.4 below, this is a property of the underlying scalar map in exact arithmetic; it is not, and is not presented as, a convergence proof for the full IP-C-PDL algorithm.

A.1. Boundedness/invariance

The tent map $T : [0,1] \to [0,1]$ used in the perturbation step (20) is $T(x) = 2x$ for $x < 0.5$ and $T(x) = 2(1-x)$ for $x \geq 0.5$. For any $x \in [0,1]$, direct substitution in both branches gives $T(x) \in [0,1]$, so $[0,1]$ is forward-invariant under T and the chaotic state $\mathbf{s}^{(k)}$ remains in $(0,1)^n$ for every iteration k. Consequently the raw offset $\mathbf{s}^{(k)} - 0.5$ used in the safeguarded direction $\mathbf{v}$ (21) is always well-defined, and the perturbation magnitude Δ_c (23) is uniformly bounded by construction, so the chaotic contribution to the PDL step can never drive an iterate outside the trust region by more than the capped amount stated in Section 3.5.

A.2. Invariant density via the transfer (Frobenius–Perron) operator

T has two monotone branches with inverses $T_1^{-1}(x) = x/2$ (image $[0, 0.5)$) and $T_2^{-1}(x) = 1 - x/2$ (image $[0.5, 1]$), each satisfying $|(T_i^{-1})'(x)| = 1/2$. For a density f on $[0,1]$, the Frobenius–Perron transfer operator $\mathcal{P}$ associated with T is defined by

$$\begin{aligned}(\mathcal{P}f)(x) &= f(T_1^{-1}x)\,|(T_1^{-1})'x| \\ &\quad + f(T_2^{-1}x)\,|(T_2^{-1})'x| \\ &= \tfrac{1}{2}f\big(\tfrac{x}{2}\big) + \tfrac{1}{2}f\big(1 - \tfrac{x}{2}\big)\end{aligned} \tag{24}$$

following the standard construction for piecewise-monotone interval maps.[71] A density f^* is invariant under T iff it is a fixed point of $\mathcal{P}$, i.e. $\mathcal{P}f^* = f^*$. Taking the uniform density $f^*(x) = 1$ on $[0,1]$ and substituting into (24) gives, for every $x \in [0,1]$,

$$\begin{aligned}(\mathcal{P}f^*)(x) &= \tfrac{1}{2}(1) + \tfrac{1}{2}(1) \\ &= 1 = f^*(x)\end{aligned} \tag{25}$$

so $f^* \equiv 1$, i.e. Lebesgue measure on $[0,1]$, is invariant under T. This is the precise sense in which the tent map's long-run occupation statistics are uniform on $[0,1]$; it is a direct computation on T's own transfer operator rather than an argument carried over from another map, addressing the concern raised in review that the topological-conjugacy argument used in an earlier revision, while not incorrect, was a less direct way to establish this specific property.

A.3. Ergodicity

That a piecewise-linear, piecewise-monotone interval map with uniformly expanding slope $|T'| = 2 > 1$ everywhere it is differentiable (such as the tent map) is ergodic with respect to its invariant density is a standard result for this class of maps; see[71] for the transfer-operator-based proof and[72,73] for the equivalent statement obtained via the map's topological conjugacy to the doubling map $y \mapsto 2y \bmod 1$. We invoke this standard result rather than re-derive it; the derivation in A.2 above is included specifically because it is self-contained and specific to T, whereas the ergodicity statement itself is cited rather than re-proved here.

A.4. Scope and limitations of this result

Two caveats delimit exactly what A.1–A.3 do and do not establish. First, the result concerns the ideal map T under exact real-number arithmetic; the finite-precision, floating-point implementation used in the actual code additionally carries a norm-based safeguard ($\epsilon_v = 10^{-12}$, **Equation (21)**) precisely because exact orbit behaviour is not guaranteed in floating point, and Section 8.8 reports how often that safeguard is exercised empirically rather than assuming it is exercised never. Second, and more fundamentally, boundedness and ergodicity of the scalar chaotic generator T are properties of the perturbation-direction source only. They do not, on their own, imply convergence—in any sense—of the IP-C-PDL iterate sequence $\{\mathbf{x}^{(k)}\}$ to a stationary point or global optimum of the constrained dispatch problem, which depends on the interior-point penalty subproblem, the trust-region/dog-leg

step, and the SQP refinement described in Section 3. We do not claim, and this appendix should not be read as establishing, a convergence guarantee for the full algorithm; all convergence statements elsewhere in this paper are empirical observations over the reported runs, phrased accordingly (see Section 8.10).

An International Journal of Optimization and Control: Theories & Applications (https://accscience.com/journal/ijocta)



This work is licensed under a Creative Commons Attribution 4.0 International License. The authors retain ownership of the copyright for their article, but they allow anyone to download, reuse, reprint, modify, distribute, and/or copy articles in IJOCTA, so long as the original authors and source are credited. To see the complete license contents, please visit http://creativecommons.org/licenses/by/4.0/.