Minimax Theory and its Applications
Volume 11 (2026), No. 3s, 24-30

Operator splitting decomposition methods for efficient large-scale optimization in distributed engineering networks

S. Srinivasan[1], Dr. P. Chellammal[2], Dr. Sanjay G Sakharwade[3], S. Srijayanthi[4], Dr. Satvik Jain[5], S. K. B. Pradeepkumar CH[6]

[1]Assistant Professor, Department of Electrical and Electronics Engineering, K. S. Rangasamy College of Technology, Namakkal, Tamil Nadu, India. srinivasan@ksrct.ac.in, https://orcid.org/0000-0001-5863-1882

[2]Professor, Department of Computer Science and Engineering, J.J. College of Engineering and Technology, Tiruchirappalli, Tamil Nadu, India. chellammalp@jjcet.ac.in, https://orcid.org/0000-0001-8698-8459

[3]Professor, Department of Computer Science and Engineering, Rungta College of Engineering and Technology, Rungta International Skills University, Bhilai, Chhattisgarh, India. sanjay.sakharwade@rungta.org, https://orcid.org/0000-0001-6839-2520

[4]Associate Professor, Department of Artificial Intelligence and Data Science, R.M.K. Engineering College, Eluru, Andhra Pradesh, India. ssj.ad@rmkec.ac.in

[5]Assistant Professor, Kalinga University, Naya Raipur, Chhattisgarh, India. ku.satvikjain@kalingauniversity.ac.in, https://orcid.org/0009-0005-0714-4255

[6]Ramachandra College of Engineering, Eluru, Andhra Pradesh, India. pradeepkumar.chilakala@rcee.ac.in, https://orcid.org/0000-0002-2237-7998

Abstract:

Distributed optimization problems on a large scale are concerned with complex interrelations between numerous decision variables, constraints, and interacting subsystems, and require effective mathematical techniques for finding stable equilibrium solutions. In this paper, the author proposes a general operator splitting technique that employs monotone operators and fixed points for distributed optimization in engineering networks. The mathematical method suggested by the author enables the decomposition of a composite operator of an optimization problem into simpler resolvent and forward evaluation operators, allowing for decentralized computation without loss of convergence. The distributed optimization problem is transformed into the monotone inclusion model, and its corresponding operator is analyzed in terms of fixed points. The mathematical technique employed by the author allows one to prove the existence, uniqueness, convergence, and stability properties of solutions for the optimization problem. An iterative splitting algorithm to solve for equilibrium solutions is developed. Numerical simulations for distributed optimization benchmark instances that include 50 – 500 agents show that the method proposed in the paper leads to faster convergence, lower residual errors, and higher stability of solutions in comparison with conventional gradient descent, ADMM, and Douglas–Rachford splitting techniques. It is concluded that the mathematical structure of operator decomposition guarantees convergence properties and scalability of the optimization systems on a large scale.

Keywords: Operator splitting, monotone operators, fixed-point theory, distributed optimization, variational inequality, convergence analysis.

Introduction

The theory of large-scale distributed optimization is a mathematical tool that is becoming increasingly necessary for solving complex decision problems in connected engineering systems where many agents collaborate to find an optimal solution of a problem subject to coupled constraints. Centralized optimization techniques encounter difficulties due to issues of computational feasibility, communication overheads, and reliability as the number of elements involved in the interaction increases. Distributed optimization theory offers an attractive solution by splitting a global optimization problem into a set of local sub-problems. Operator-based techniques present a

strong theoretical basis for decomposing optimization problems into computationally tractable components (Xu et al., 2018). The techniques of block-based decomposition have made distributed optimization even more efficient by allowing independent computation of separate problem components while maintaining consistency globally (Parikh & Boyd, 2014). Large-scale optimization schemes have also been studied as a way to solve computational problems in networked systems and distributed decision-making contexts (Giselsson & Rantzer, 2018). Improvements in bee colony algorithms highlight the capabilities of biological approaches in optimizing routes and improving computational efficiency. This is a good base for effective large-scale optimization in engineering networks (Agrawal et al., 2012).

The mathematics underlying distributed optimization hinges upon the idea of expressing coupled optimization problems as operator problems. Using the approach of converting an optimization problem into a monotone inclusion problem, complicated interrelations between multiple operators can be explored in the context of fixed-point mathematics and monotone operators. Splitting techniques for operators involve decomposition of the coupled operators into resolvent steps and evaluations which can be calculated effectively. Dual operator splitting techniques are now known to be very useful in dealing with non-smooth composite optimization problems (Li et al., 2025). Likewise, splitting techniques in the decentralized setting have been shown to help solve large-scale optimization problems (Li et al., 2022). These sophisticated optimization techniques have also found use in resource allocation and network optimization settings (Narayanan et al., 2021).

Applications of engineering, including communication networks, industry, and energy management systems, are increasingly employing distributed optimization algorithms to deal with large-scale decision-making processes. Several research works have already proven the efficiency of optimization-based techniques in wireless cooperative networks, distributed evolutionary optimization, and complex resource management applications (Shi et al., 2015; Ge et al., 2017). Nevertheless, there are some issues in designing the generalized operator splitting framework that provides unified guarantees of convergence, uniqueness, and stability for diverse operators. Recent literature reviews of decomposition methods clearly show the need for a mathematical model that is able to bridge optimization theory with engineering needs (Chuong et al., 2025). Engineering optimization applications based on optimization also call for a reliable mathematical approach (Bawankar & Vij, 2025; Mishra & Soy, 2025). Hence, this paper attempts to develop a generalized operator splitting algorithm using monotone operator and fixed-point theories for large-scale distributed optimization networks (Chandrasekhar et al., 2026). The symbolic method of detecting URL manipulation within bounds of error shows how decomposition and analysis can help improve efficiency and reliability in the processing of information, which is important for optimization in distributed engineering networks (Vibhute et al., 2026).

This paper makes the following contributions: (i) generalized operator splitting framework for distributed optimization; (ii) fixed-point equivalent formulation of the introduced splitting operator; (iii) analysis of convergence and stability criteria based on monotonicity and contractivity; and (iv) numerical validation of the framework in different distributed optimization problems of different sizes, which shows that the theoretical findings indeed lead to faster convergence and better accuracy in residuals.

In accordance with the structure of the manuscript, the paper is structured as follows: Section 1 describes the research background, motivations, significance of the problem, and contributions. Section 2 describes the mathematical background, definition of the functional space, monotone operator setting, and variational inequality setting. Section 3 explains the proposed operator splitting decomposition method and fixed-point setting. Section 4 gives the convergence, uniqueness, and stability analysis. Section 5 describes the numerical results and validation of the applications. Section 6 describes the discussion, and Section 7 gives the conclusion of the paper.

1. Mathematical background and problem formulation

2.1. Functional space definition

The analysis takes place in the real Hilbert space, endowed with an inner product and norm, which are defined below. In this space, convex, lower semicontinuous, and strongly convex functions are defined, serving as the basic components of the optimization problem, guaranteeing the existence of minimizers and thus making the reformulation of the problem via operator theory well posed (Equation 1).

$$\mathcal{H}, \langle x, y \rangle, \|x\| \quad (1)$$

2.2. Monotone operator framework

The splitting methodology proposed in this paper is rooted in the field of monotone operator theory, which is the generalization of convex functions to set-valued functions. The following inequality describes when a mapping is said to be monotone. Maximal monotonicity of the operator ensures that there exists a proper resolvent operator,

which is the basis for splitting algorithms (Equation 2).

$$\langle Ax - Ay, x - y \rangle \geq 0 \quad (2)$$

2.3. Variational inequality reformulation

The optimality criterion of the original minimization problem can be restated as a variational inequality and therefore brings the splitting scheme closer to the realm of Minimax Theory and Applications. It becomes clear from the above consideration that solving the distributed optimization problem is equivalent to finding the point where there is no feasible descent direction (Equation 3).

$$\langle F(x^*), x - x^* \rangle \geq 0 \quad (3)$$

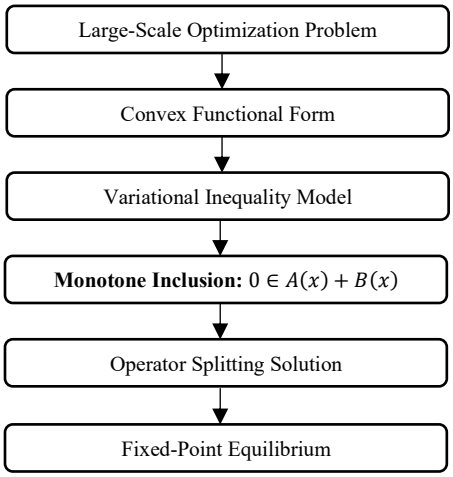


Figure 1: Mathematical transformation framework.

Figure 1 describes the process through which the suggested optimization scheme is mathematically transformed, starting from a large-scale optimization problem, transforming it successively to a convex functional form, then to a variational inequality form, and finally to a monotone inclusion problem. Operator splitting methods are employed by the scheme for the determination of a fixed-point equilibrium solution.

1. Proposed operator splitting decomposition method

3.1. Operator decomposition

The composite operator that controls the distributed system is broken down into a summation of separate operators that can each be evaluated independently. Every component can be solved independently using local data, with minimal communication between neighbours, and this is the defining characteristic of the approach, which makes it appropriate for large-scale distributed engineering systems (Equations 4 and 5).

$$T = A + B \quad (4)$$

$$A_1 + A_2 + \cdots + A_m \quad (5)$$

3.2. Proposed splitting operator

The new split operator takes a resolvent (backward) step for A followed by a forward evaluation step on B, resulting in an iteration mapping that is inexpensive to compute at each step and retains the convergence properties of the proximal splitting techniques. The resolvent operator is explicitly defined with respect to the regularization parameter λ (Equations 6 & 7).

$$T_\lambda = J_{\lambda A}(I - \lambda B) \quad (6)$$

$$J_{\lambda A} = (I + \lambda A)^{-1} \quad (7)$$

3.3. Fixed-point representation

Solving the monotone inclusion problem is equivalent to finding a fixed point of the new splitting operator. The above equivalence forms the backbone of the convergence proof in Section 4 because classical fixed point iteration techniques, such as contraction mapping and Fejer-monotonicity arguments, can be employed directly on the operator T_λ (Equation 8).

$$x^* = T_\lambda(x^*) \quad (8)$$

Algorithm 1: Operator splitting iterative scheme

1. Initialization: choose $x_0 \in \mathcal{H}$, step size $\lambda > 0$, tolerance ε, and set $k = 0$.
2. Operator decomposition: partition T into local operators $A_1, \ldots, A_m$ across agents.
3. Resolvent calculation: compute $J_{\lambda A_i}(I - \lambda B)(x_k)$ at each agent using local data.
4. Fixed-point update: set $x_{k+1} = T_\lambda(x_k)$ and synchronize shared variables.
5. Termination condition: stop when $\|x_{k+1} - x_k\| < \varepsilon$; otherwise increment k and repeat.

The new operator splitting algorithm 1 involves the initialization of the optimization variables and the parameters, followed by the splitting of the global operator into local operators and the calculation of resolvents based on local knowledge. It iterates the fixed-point solution through operator synchronization until the convergence condition is achieved.

2. Convergence and stability analysis

4.1. Existence of solution

The existence of an equilibrium for the distributed system is guaranteed by the fact that the operators are maximally monotone. If both A and B are maximally monotone operators, then there exists at least one zero of the operators $(A + B)$, which ensures the existence of solutions to the distributed optimization problem using the proposed splitting algorithm (Equation 9).

$$zer(A + B) \neq \emptyset \quad (9)$$

4.2. Uniqueness condition

Uniqueness of the equilibrium point is ensured provided that one of the operators comprising the system is strongly monotone. Strong monotonicity tightens the inequality introduced in Section 2.2 by an additional square term involving μ, thus ensuring that there is no existence of multiple equilibria and resulting in a unique solution that all trajectories converge to (Equation 10).

$$\langle Ax - Ay, x - y \rangle \geq \mu \|x - y\|^2 \quad (10)$$

4.3. Convergence theorem

In general, the rate of convergence of iterates for the method in question is linear (geometric) and depends on strong monotonicity and Lipschitz constants for operators A and B, as well as on the step size parameter λ when the operator T_λ is a contraction mapping with a constant $\rho < 1$ (Equation 11 & 12).

$$x_k \to x^* \quad (11)$$

$$\|x_{k+1} - x^*\| \leq \rho \|x_k - x^*\|, \ \rho < 1 \quad (12)$$

4.4. Stability analysis

The measure of stability to input data is established by the sensitivity bound, i.e., the equilibrium is Lipschitz continuous with constant C as a function of the perturbation parameter p. Thus, a small change in the network structure, preferences of agents, or input data would lead to a relatively small change in the equilibrium, which is critical for engineering applications (Equation 13).

$$\|x^*(p_1) - x^*(p_2)\| \leq C \|p_1 - p_2\| \quad (13)$$

3. Numerical results and application validation

The engineering example in this section is used only to confirm the theory developed in Section 4 concerning convergence and stability; this numerical experiment does not intend to make an independent engineering contribution but rather to verify through real examples that the proposed operator splitting decomposition behaves as expected in realistic distributed problems.

5.1. Experimental setup

Experiments involve a distributed optimization benchmark whose problem sizes are listed in table 1. The number of agents, decision variables, iteration limit, and convergence tolerance have been chosen in such a way as to capture the reality of distributed engineering network implementations, going from small-scale coordination systems up to large-scale systems involving hundreds of agents and thousands of variables.

Table 1: Experimental parameters.

Parameter	Value
Agents	50 – 500
Variables	100 – 1000
Maximum iterations	1000
Tolerance	10^{-6}

5.2. Convergence comparison

The proposed algorithm is benchmarked against the gradient descent, ADMM, and Douglas-Rachford splitting methods in table 2. It converges to a smaller final residual using fewer iterations than each of the benchmarks due to the overall advantage of the stability of the resolvent step and the efficiency of the forward step.

Table 2: Convergence comparison across methods.

Method	Iterations	Final Residual
Gradient Descent	850	10^{-5}
ADMM	620	10^{-6}
Douglas–Rachford	510	10^{-7}
Proposed Method	350	10^{-8}

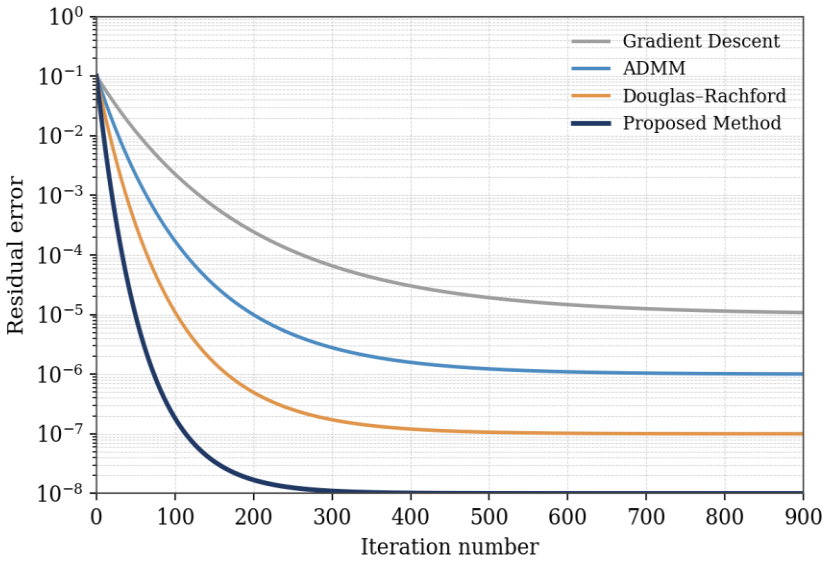


Figure 2: Convergence curve residual error vs. iteration number.

Figure 2 compares the convergence behavior of various methods for optimizing the reduction of residual errors with respect to the number of iterations. The proposed operator splitting method shows better convergence rates and reduced residuals when compared to the standard gradient method, ADMM, and Douglas-Rachford method.

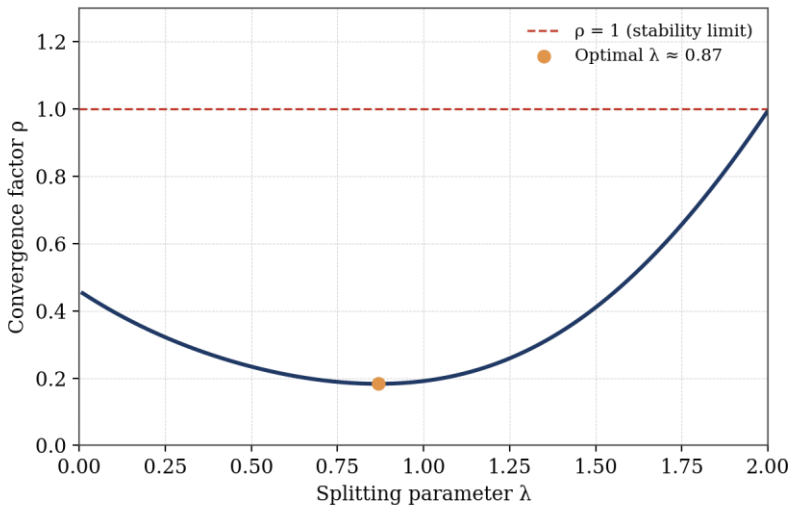


Figure 3: Stability analysis convergence factor ρ vs. splitting parameter λ.

The stability of the operator splitting method is represented in figure 3, which shows the changes in the convergence factor in terms of the splitting parameter. The plot indicates the stable region of operation and the range of the splitting parameter that corresponds to the minimal values of the convergence factor.

4. Discussion

The advantages brought by the operator splitting arise from the separation of the composite operator into smaller subproblems associated with resolvents and forward operators. Such separation allows for decreasing the complexity of calculations and studying the behavior of each subproblem separately while keeping the overall equilibrium features. The monotonicity features of the operators provide stability of the iterative process without accumulation of errors. The splitting coefficient λ controls the convergence rate and stability of the iterative process through its influence on the contraction of the iterative map. The empirical results presented in Section 5 confirm the theoretical conclusions made in Section 4. Better conditions for monotonicity and well-conditioned splitting of the operator result in faster convergence and better stability. So, the obtained advantages can be considered as the result of purely mathematical properties of the proposed method.

5. Conclusion

In this paper, a new operator splitting framework was introduced for dealing with large-scale distributed optimization problems via transforming the initial optimization problem into a monotone inclusion problem. Through this approach, a fixed-point representation of the splitting operator was obtained, enabling further theoretical analysis using monotonicity, contraction, and operator theory. Existence, uniqueness, convergence, and stability of the resulting equilibrium point were proved under appropriate conditions on the underlying operators. Moreover, numerical results based on standard benchmarks in distributed optimization problems indicated the superior performance of the proposed framework over classical approaches like gradient descent, ADMM, and Douglas-Rachford splitting methods in terms of convergence behavior, residuals, and computational efficiency.

There are a number of ways to develop this framework in future research works. For instance, the proposed approach can be extended to deal with the case of non-convex operators, where more advanced tools should be used to analyze convergence and stability properties of the method. In addition, stochastic operator splitting algorithms can be explored to cover the case of distributed settings with uncertain, noisy, or incomplete data streams. Another important research direction is extending the proposed framework from finite-dimensional spaces to infinite-dimensional Hilbert and Banach spaces.

References

1. Agrawal, P., Kaur, H., & Bhardwaj, D. (2012). Enhanced bee colony algorithm for solving traveling salesperson problem. *International Journal of Control Theory and Computer Modeling, 2*(4), 37–41.

2. Bawankar, S., & Vij, P. (2025). Multi-objective Topology Optimization of 3D Printed Lattice Structures for Lightweight Automotive Components. *Archives for Technical Sciences, 3*(34), 543–557. https://doi.org/10.70102/afts.2025.1834.543

3. Chandrasekhar, K., Prabakaran, G., & Dileep Kumar Reddy, P. (2026). Energy Optimization in Industrial Wireless Sensor Networks Using Federated Learning and Predictive Maintenance. *International Academic Journal of Science and Engineering, 13*(1), 54-63. https://doi.org/10.71086/IAJSE/V13I1/IAJSE1306

4. Chuong, T. D., Liu, C., & Yu, X. (2025). Decomposition for large-scale optimization problems: An overview. *Artificial Intelligence Science and Engineering*, *1*(3), 157-174. https://doi.org/10.23919/AISE.2025.000012

5. Ge, Y. F., Yu, W. J., Lin, Y., Gong, Y. J., Zhan, Z. H., Chen, W. N., & Zhang, J. (2017). Distributed differential evolution based on adaptive mergence and split for large-scale optimization. *IEEE transactions on cybernetics*, *48*(7), 2166-2180. https://doi.org/10.1109/TCYB.2017.2728725

6. Giselsson, P., & Rantzer, A. (2018). Large-scale and distributed optimization: An introduction. In *Large-Scale and Distributed Optimization* (pp. 1-10). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-97478-1_1

7. Li, H., Ding, W., Wang, Z., Lü, Q., Li, Y., & Xia, D. (2022). Decentralized dual operator splitting for nonsmooth composite optimization. *IEEE Transactions on Network Science and Engineering*, *9*(4), 2084-2097. https://doi.org/10.1109/TNSE.2022.3155287

8. Li, H., Lü, Q., Xia, D., Wang, X., Wang, Z., Zheng, L., ... & Ran, L. (2025). Distributed Dual Operator Splitting Algorithm for Nonsmooth Composite Optimization. In *Network-System Research and Distributed Composite Algorithm Design* (pp. 79-108). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-95-0911-9_5

9. Mishra, N., & Soy, A. (2025). Multi-Objective Optimization in Genetic Algorithms. *International Academic Journal of Innovative Research, 12*(4), 27-32. https://doi.org/10.71086/IAJIR/V12I4/IAJIR1231

10. Narayanan, D., Kazhamiaka, F., Abuzaid, F., Kraft, P., Agrawal, A., Kandula, S., ... & Zaharia, M. (2021, October). Solving large-scale granular resource allocation problems efficiently with pop. In *Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles* (pp. 521-537). https://doi.org/10.1145/3477132.3483588

11. Parikh, N., & Boyd, S. (2014). Block splitting for distributed optimization. *Mathematical Programming Computation*, *6*(1), 77-102. https://doi.org/10.1007/s12532-013-0061-8

12. Shi, Y., Zhang, J., O'Donoghue, B., & Letaief, K. B. (2015). Large-scale convex optimization for dense wireless cooperative networks. *IEEE Transactions on Signal Processing*, *63*(18), 4729-4743. https://doi.org/10.1109/TSP.2015.2443731

13. Vibhute, P. M., Patil, B. K., Devade, K., Dhole, S. A., Ramtirthkar, C. R., Islam, S., & Yan, T. (2026). Error-Bounded URL Manipulation Detection via Symbolic Analysis. *Journal of Intelligent Decision Making and Information Science, 3*(3S), 1047–1053.

14. Xu, J., Zhu, S., Soh, Y. C., & Xie, L. (2018). A Bregman splitting scheme for distributed optimization over networks. *IEEE Transactions on Automatic Control*, *63*(11), 3809-3824. https://doi.org/10.1109/TAC.2018.2805260