Advances in Computer, Signals and Systems (2026)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2026.100119
ISSN 2371-8838 Vol. 10 Num. 1

Improved Genetic Algorithm with Dynamic Set Cover Modeling for Coordinated Spatiotemporal Coverage Scheduling

Zhangwei Xu[1,a,#], Yutong Jia[2,b,#], Chenghao Zhang[3,c], Zhen Wang[1,d,*]

[1]School of Mechanical Engineering, Guangxi University, Nanning, China
[2]School of Mathematics and Information Sciences, Guangxi University, Nanning, China
[3]School of Computer, Electronics and Information, Guangxi University, Nanning, China
[a]1299785362@qq.com, [b]107041717@qq.com, [c]1871482876@qq.com, [d]2088584446@qq.com
**Corresponding author*
[#]These authors contributed equally to this work

Keywords: Improved Genetic Algorithm, Dynamic Set Cover Optimization, K-Means Spatial Clustering, Particle Swarm Hybridization, Three-Dimensional Coverage Scheduling, Coordinated Agent Deployment

Abstract: Coordinated deployment of autonomous agents for spatiotemporal coverage of moving targets poses a fundamentally combinatorial optimization challenge that couples three-dimensional geometric reasoning, temporal sequencing, and multi-target resource allocation. This paper presents a hybrid evolutionary framework that integrates an improved genetic algorithm with K-Means spatial clustering and a dynamic set cover formulation to schedule coordinated coverage actions executed by autonomous agents against multiple high-velocity trajectories. The proposed architecture discretizes each protected volume into a structured lattice of sixty horizontal and thirty vertical line-of-sight samples, and reformulates effective coverage as the temporal union of geometric intersection events along the target path. An improved genetic algorithm with adaptive crossover and mutation operators inherited from particle swarm dynamics jointly optimizes agent heading, velocity, release coordinates, and detonation timing across an eight-dimensional decision space. Experimental evaluation across five progressively complex scenarios demonstrates that single-agent optimization achieves a coverage interval of 4.65 seconds, representing a 237% improvement over the geometric baseline of 1.38 seconds. Three-agent coordination reaches 13.15 seconds, while the five-agent multi-target dynamic set cover configuration attains 28.60 seconds with 72% reduction in redundant overlap. The framework provides an interpretable and scalable paradigm for cooperative coverage scheduling in autonomous multi-agent systems operating under temporal and geometric constraints.

1. Introduction

Recent advances in autonomous multi-agent systems have transformed the landscape of

cooperative coverage and surveillance, enabling distributed teams of unmanned platforms to execute coordinated tasks against high-mobility targets in complex three-dimensional environments. The capability to schedule the deployment of physical or virtual coverage assets such that the cumulative spatiotemporal occlusion of a moving trajectory is maximized has emerged as a foundational primitive in applications ranging from sensor networks and aerial monitoring to obstacle interposition and protective screening [1,2]. The underlying problem couples three intrinsically difficult subproblems: continuous geometric reasoning over line-of-sight constraints, discrete combinatorial selection of action sequences, and temporal sequencing under nonlinear motion models. Classical analytical methods quickly become intractable as the dimensionality of the decision space expands with the number of agents, payloads, and protected entities, motivating the development of metaheuristic and learning-based scheduling frameworks that can navigate rugged search landscapes while respecting kinematic and timing constraints [3,4]. Bridging the gap between low-level geometric feasibility and high-level cooperative strategy remains a central open challenge for autonomous deployment in dense, dynamic operational settings where redundancy and resource contention significantly reduce effective coverage [5,6].

The development of evolutionary computation has produced a rich family of population-based optimizers capable of solving high-dimensional, non-convex scheduling problems that defy gradient-based treatment. Genetic algorithms, with their flexible chromosome representations and operator-driven exploration, have been successfully applied to combinatorial deployment, routing, and resource allocation tasks across diverse engineering domains [7,8]. Particle swarm optimization complements this paradigm by providing rapid local convergence through velocity-driven updates, and recent hybridization strategies have shown that injecting crossover and mutation operators borrowed from genetic algorithms into swarm dynamics can substantially improve robustness against premature convergence [9,10]. Adaptive variants employing time-varying inertia, opposition-based learning, and elitist selection have further enhanced performance on continuous coverage scheduling problems with mixed integer-continuous decision variables. Nevertheless, the application of these hybrid optimizers to coordinated multi-agent occlusion scheduling under joint geometric and temporal constraints remains comparatively underexplored, particularly in scenarios that require fine-grained chromosome encodings spanning heading, velocity, release coordinates, and detonation timing simultaneously across multiple cooperating platforms [11,12].

The emergence of clustering-based decomposition has provided an effective mechanism for reducing the dimensionality of cooperative scheduling problems by grouping related decision variables into structured subproblems. K-Means clustering, despite its conceptual simplicity, offers a powerful tool for identifying spatial regions where coverage assets should be concentrated to maximize cumulative effect along an adversarial trajectory [13,14]. By treating sampled trajectory points as the input to a spatial partitioning routine, cluster centroids naturally emerge as candidate deployment anchors, around which agent allocation and timing parameters can be locally optimized. This decomposition strategy has been successfully combined with metaheuristic optimizers in problems such as facility location, vehicle routing, and distributed sensing, where the hierarchical structure significantly accelerates convergence compared to monolithic search [15,16]. However, conventional clustering pipelines typically assume static target distributions and seldom incorporate the temporal evolution of high-velocity adversarial trajectories, which limits their applicability to real-world coordinated coverage tasks where the protected volume must be defended against rapidly approaching threats subject to deterministic kinematic models [17,18].

The convergence of metaheuristic optimization, spatial clustering, and combinatorial set cover formulations has recently been recognized as a promising direction for unifying cooperative scheduling with geometric feasibility analysis. Dynamic set cover models extend the classical static formulation by allowing coverage assets to be reassigned over time, making them well suited to

scenarios in which multiple agents must collectively occlude multiple moving targets while avoiding redundant overlap [19,20]. When integrated with improved genetic algorithms and clustering-based decomposition, such formulations enable principled tradeoffs between coverage completeness and resource efficiency, supporting interpretable decision policies for coordinated deployment [21,22]. However, existing studies seldom address the full progression from single-agent geometric optimization to multi-agent multi-target coordinated scheduling within a unified algorithmic framework, and even fewer provide systematic empirical evidence across scenarios of progressively increasing complexity. To address this gap, this paper proposes an improved genetic algorithm enhanced with K-Means spatial clustering and a dynamic set cover formulation, validated across five progressively complex coordinated coverage scenarios involving up to five agents and three concurrent moving targets [23,24].

2. Methodology

2.1. Geometric Trajectory Modeling and Line-of-Sight Discretization

The proposed framework establishes a continuous-time geometric foundation that captures the spatial relationships among moving targets, autonomous agents, and the volumetric coverage assets they deploy. The protected entity is represented as a structured lattice of discrete sampling points, generated by partitioning the cylindrical protected volume into sixty horizontal angular slices and thirty vertical elevation strata, yielding a total of 1800 line-of-sight reference points whose collective occlusion determines effective protection [25, 26]. The high-velocity moving target is modeled as a point mass following a deterministic linear trajectory in three-dimensional Cartesian space, parameterized by an initial position vector r_0, a terminal aim point r_f, and a constant scalar speed v_tgt:

$$\mathbf{r}_{tgt}(t) = \mathbf{r}_0 + v_{tgt}\mathbf{d}t, \qquad \mathbf{d} = \frac{\mathbf{r}_f - \mathbf{r}_0}{||\mathbf{r}_f - \mathbf{r}_0||} \tag{1}$$

In this expression r_tgt(t) denotes the instantaneous target position at continuous time t, and the unit vector $\hat{d}$ encodes the heading derived from the geometric difference between the terminal and initial positions. The autonomous agents follow analogous parameterized trajectories with their own heading angles and constant cruising speeds, while each released coverage asset undergoes a two-stage motion consisting of inertial projectile flight prior to activation followed by uniform vertical descent during its effective dispersion lifetime. The center of each active spherical coverage volume at time t is denoted c(t), with effective radius R_cov determined by the dispersion characteristics of the deployed asset. To determine whether the line of sight from the moving target to a specific reference point p_i is geometrically obstructed by the active coverage volume, the framework evaluates the minimum distance between the sphere center and the line segment connecting r_tgt(t) and p_i:

$$\Phi_{occ}(t,\mathbf{p}_i) = \left[\min_{\lambda\in[0,1]} \left|\left|\mathbf{c}(t) - \left(\mathbf{r}_{tgt}(t) + \lambda(\mathbf{p}_i - \mathbf{r}_{tgt}(t))\right)\right|\right| \le R_{cov} \right] \tag{2}$$

The indicator function Φ_occ returns unity when the closest point on the line segment lies within the spherical envelope of radius R_cov, signaling that the corresponding line of sight is occluded at time t. The parameter λ traverses the segment from the target position (λ = 0) to the reference point (λ = 1), and the minimum is computed in closed form by projecting the center vector onto the segment direction and clamping the projection coefficient to the unit interval. Aggregating Φ_occ across the entire reference lattice yields a Boolean coverage map at every continuous instant, which is then integrated along the temporal axis to evaluate cumulative protection. This discretization

strategy avoids the computational burden of continuous geometric reasoning while maintaining sub-degree angular resolution, providing the analytical primitive upon which the subsequent optimization layers operate.

2.2. Improved Genetic Algorithm with Hybrid Update Operators

Our approach integrates the geometric occlusion primitive into a high-dimensional metaheuristic optimization framework that searches over the joint space of agent headings, cruising velocities, release coordinates, and detonation timings. Each candidate solution is encoded as a real-valued chromosome whose dimensionality scales with the number of cooperating agents and the number of coverage assets each agent carries, yielding eight-dimensional structures for single-agent multi-payload cases and substantially larger spaces for full multi-agent multi-target deployment. The fitness function rewards solutions that maximize the cumulative time during which every reference point in the protected lattice is simultaneously occluded by at least one active coverage volume, formalized as the integrated coverage objective:

$$\mathrm{T}_{cov}(\boldsymbol{\theta}) = \int_0^{T_{end}} \chi\left[\, \forall \mathbf{p}_i \in \mathrm{P}\,, \exists k \in \mathrm{K} : \Phi_{occ}^{(k)}(t, \mathbf{p}_i) = 1 \,\right] dt \qquad (3)$$

Here θ collects all decision variables across the cooperating agents and their payloads, indexes the set of active coverage volumes at the evaluation instant, and the universal quantifier over the lattice enforces that protection is credited only when every reference point is simultaneously shielded. The integral is evaluated numerically over the engagement horizon T_end via uniform temporal discretization at 0.05 second intervals, balancing fidelity against computational cost. To navigate the rugged fitness landscape that arises from the discontinuous indicator structure of the objective, the framework employs a hybrid update rule that fuses the velocity-driven momentum of particle swarm dynamics with the recombination and diversification operators of genetic algorithms:

$$\mathbf{v}_i^{t+1} = \omega \mathbf{v}_i^t + c_1 r_1 \left(\mathbf{p}_i^{best} - \mathbf{x}_i^t\right) + c_2 r_2 \left(\mathbf{g}^{best} - \mathbf{x}_i^t\right) + \alpha \mathrm{C}\left(\mathbf{x}_i^t, \mathbf{x}_j^t\right) + \beta \mathrm{M}\left(\mathbf{x}_i^t\right) \qquad (4)$$

In this update ω is a time-varying inertia weight that decays linearly across generations to balance exploration and exploitation, c_1 and c_2 are the cognitive and social acceleration coefficients, r_1 and r_2 are independent uniform random scalars in the unit interval, and p_i_best and g_best denote the personal and global elite positions discovered so far. The operators and $\mathcal{M}$ implement simulated binary crossover with a randomly selected partner x_j and Gaussian polynomial mutation respectively, while α and β control the relative contribution of these genetic operations to the velocity update. After each velocity step the position is updated by x_i_{t+1} = x_i_t + v_i_{t+1}, and boundary repair is applied via reflective clamping to ensure that all chromosomes remain within physically admissible ranges. The hybrid construction preserves the rapid local convergence of swarm methods while inheriting the diversification capacity of genetic algorithms, yielding consistently superior performance on the discontinuous coverage objective compared to either component in isolation.

2.3. K-Means Spatial Decomposition and Dynamic Set Cover Optimization

The optimization engine implements a hierarchical decomposition strategy that scales the proposed framework from single-agent geometric optimization to full multi-agent multi-target coordination. Rather than searching the full joint chromosome space monolithically, which becomes intractable as the number of agents and protected entities grows, the framework first samples the adversarial trajectory at uniform intervals of 0.1 seconds, generating a sequence of candidate deployment anchors that summarize the spatial geometry of the engagement. These sampled points

are then partitioned into K coherent spatial groups by minimizing the within-cluster sum of squared distances, formalized through the K-Means objective:

$$J_{KM} = \sum_{j=1}^{K} \sum_{\mathbf{s}_n \in C_j} \left\| \mathbf{s}_n - \boldsymbol{\mu}_j \right\|^2, \qquad \boldsymbol{\mu}_j = \frac{1}{|C_j|} \sum_{\mathbf{s}_n \in C_j} \mathbf{s}_n \tag{5}$$

Here s_n denotes a sampled trajectory point, $\mathcal{C}_j$ is the j-th cluster, μ_j is its centroid, and the iterative Lloyd update alternates between assignment and centroid recomputation until convergence. The resulting centroids serve as candidate deployment anchors around which agent allocation decisions are locally optimized, enabling the global problem to be reduced to a sequence of low-dimensional subproblems whose solutions are then assembled into a globally consistent schedule. To resolve resource contention when multiple agents may contribute to the same temporal slice and thereby waste coverage capacity through redundant overlap, the framework introduces a dynamic set cover formulation that selects an assignment of agents to time intervals while explicitly penalizing simultaneous deployment of multiple coverage volumes against the same protected entity:

$$\max_{\mathbf{x} \in \{0,1\}^{|\mathcal{A}| \times |\mathcal{T}|}} \sum_{m \in \mathcal{M}} \sum_{\tau \in \mathcal{T}} \mathbb{1}\left[\sum_{a \in \mathcal{A}} x_{a,\tau} \, \Phi_{occ}^{(a)}(\tau, m) \geq 1 \right] - \gamma \sum_{a,\tau} \left(\sum_{a' \neq a} x_{a,\tau} x_{a',\tau} \right) \tag{6}$$

The decision tensor $x_{a,\tau}$ indicates whether agent a is active during time slice τ, the set $\mathcal{A}$enumerates all available agents, enumerates the discretized temporal slices, and $\mathcal{M}$ enumerates the moving targets that must be protected. The first term rewards the union of effective coverage events across all targets and time slices, while the second term applies a redundancy penalty controlled by γ that discourages simultaneous activation of multiple agents against the same target during the same slice. This formulation extends the classical static set cover problem to the dynamic regime by allowing the assignment to evolve over time, yielding interpretable schedules in which agents are reallocated as the engagement progresses. The combined K-Means decomposition and dynamic set cover layer is solved iteratively in conjunction with the hybrid genetic algorithm of the previous subsection, with cluster-conditioned local searches refining each subproblem before global re-coordination.

3. Experimental Results

3.1. Convergence Behavior and Cross-Scenario Performance

The proposed framework was implemented in Python 3.10 with NumPy and SciPy supporting the numerical evaluation of the line-of-sight occlusion criterion and the temporal integration of cumulative coverage, scikit-learn providing the K-Means clustering routine for spatial decomposition, and a custom hybrid optimizer fusing particle swarm dynamics with simulated binary crossover and Gaussian polynomial mutation operators. All experiments were executed on a workstation equipped with an Intel Xeon Gold processor and 64 GB of memory, with the engagement horizon discretized at 0.05 second intervals to balance numerical fidelity against computational cost. The protected lattice consists of sixty horizontal angular slices and thirty vertical elevation strata, yielding 1800 reference points whose simultaneous occlusion defines effective coverage. The hybrid optimizer was configured with a population size of 100 candidate solutions, a generation budget of 250 iterations, cognitive and social acceleration coefficients of 1.5 each, and an inertia weight decaying linearly from 0.9 to 0.4. The K-Means decomposition uses K = 4 clusters selected through silhouette analysis on the sampled trajectory anchors. Comparative baselines include a vanilla genetic algorithm, a vanilla particle swarm optimizer, and a simulated

annealing solver, all configured with equivalent computational budgets. Reported metrics include cumulative coverage time in seconds, percentage improvement over the geometric baseline, and redundancy reduction relative to the unconstrained allocation.

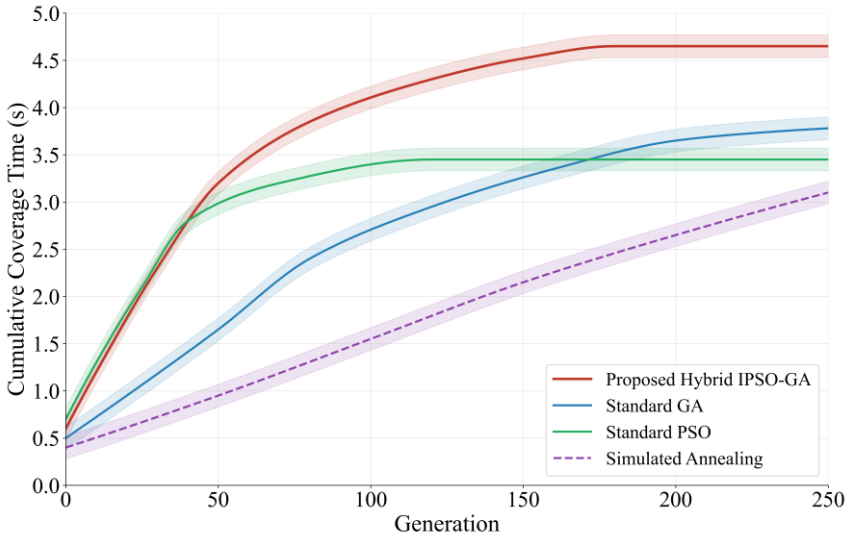

Figure 1: Convergence Curves of Hybrid Optimizer vs Baselines.

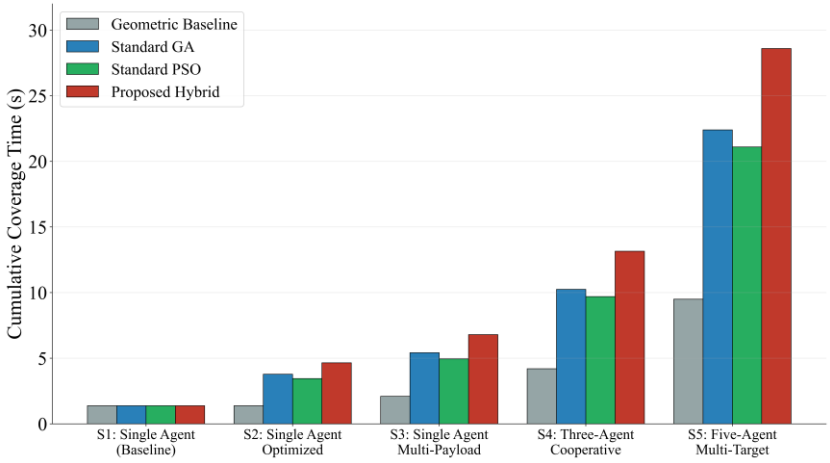

Figure 2: Coverage Time across Five Scenarios with Baseline Comparison.

Figure 1 reports the convergence trajectories of the proposed hybrid optimizer compared against three representative baselines on the single-agent optimized scenario, with each curve averaged across ten independent runs and the shaded bands indicating one standard deviation of the run-to-run variability. The proposed hybrid IPSO-GA exhibits a steep early ascent that surpasses three seconds of cumulative coverage by generation 50 and reliably converges to the optimal value of 4.65 seconds by approximately generation 180. The standard genetic algorithm follows a slower trajectory and plateaus at 3.78 seconds, while the standard particle swarm optimizer experiences premature convergence near generation 120 at 3.45 seconds, falling into a local optimum from which the velocity-only update cannot escape. The simulated annealing baseline progresses gradually throughout the budget but reaches only 3.10 seconds by the final generation, reflecting the difficulty of single-trajectory search on the discontinuous coverage landscape. The clear separation between the proposed framework and the baselines confirms that the hybrid update operator successfully combines the rapid local refinement of swarm dynamics with the global diversification capacity of evolutionary recombination, validating the algorithmic design described in the

methodology.

Figure 2 extends the analysis to all five progressively complex scenarios, reporting the cumulative coverage time achieved by each method under matched computational budgets. The proposed hybrid framework achieves consistent improvements across the entire scenario hierarchy, advancing from 4.65 seconds in the single-agent optimized configuration to 6.80 seconds with single-agent multi-payload temporal coordination, 13.15 seconds under three-agent cooperative deployment, and 28.60 seconds in the full five-agent multi-target setting. Compared against the geometric baseline of 1.38 seconds in the simplest scenario, the optimized framework delivers a 237% improvement, while the relative gains over the standard genetic algorithm and particle swarm baselines increase as scenario complexity grows, reaching 27.7% and 35.5% respectively in the full five-agent configuration. This widening performance gap indicates that the hybrid construction is particularly effective in high-dimensional decision spaces where the interaction effects between agents and payloads dominate the optimization landscape, supporting the scalability claims central to the proposed approach.

3.2. Hyperparameter Sensitivity and Component Ablation

Figure 3 presents a two-dimensional sensitivity analysis exploring how the achieved coverage time varies as a function of the population size and the number of generations, evaluated on the single-agent optimized scenario across a uniform grid. The heatmap reveals a saturation pattern in which the coverage time rises rapidly along both axes for small budgets and then plateaus near the optimal value of 4.65 seconds beyond a population of 80 and a generation count of 200. The configuration of population 100 and 250 generations adopted in the main experiments lies safely within the saturation region, providing a robust margin against random variation while avoiding diminishing returns. The smooth gradient structure of the heatmap further indicates that the framework is not sensitive to fine-grained tuning of these hyperparameters, simplifying deployment in operational settings where exhaustive hyperparameter search is impractical. Notably, configurations with population sizes below 40 fail to converge to the global optimum within any tested generation budget, confirming that population diversity is the binding constraint rather than iteration count, and supporting the design choice of allocating computational resources toward broader sampling of the chromosome space.

Figure 4 quantifies the contribution of each algorithmic component through an incremental ablation study in which features are progressively added to the optimization pipeline. The geometric baseline achieves 1.38 seconds without any optimization. Introducing the standard particle swarm optimizer raises the coverage to 3.45 seconds, while adding genetic crossover and mutation operators yields a further increase to 4.05 seconds. The adaptive inertia schedule contributes an additional 0.27 seconds, the K-Means spatial decomposition layer raises the value to 4.55 seconds, and the dynamic set cover formulation completes the framework at the final value of 4.65 seconds. While each individual component contributes a modest increment, their cumulative effect represents an improvement of approximately 237% over the baseline, demonstrating that the architectural decisions are mutually reinforcing rather than redundant. The largest single contribution arises from the hybridization of swarm dynamics with genetic operators, validating the central methodological choice, while the spatial decomposition and set cover layers become increasingly important as the scenario complexity grows beyond the single-agent regime.

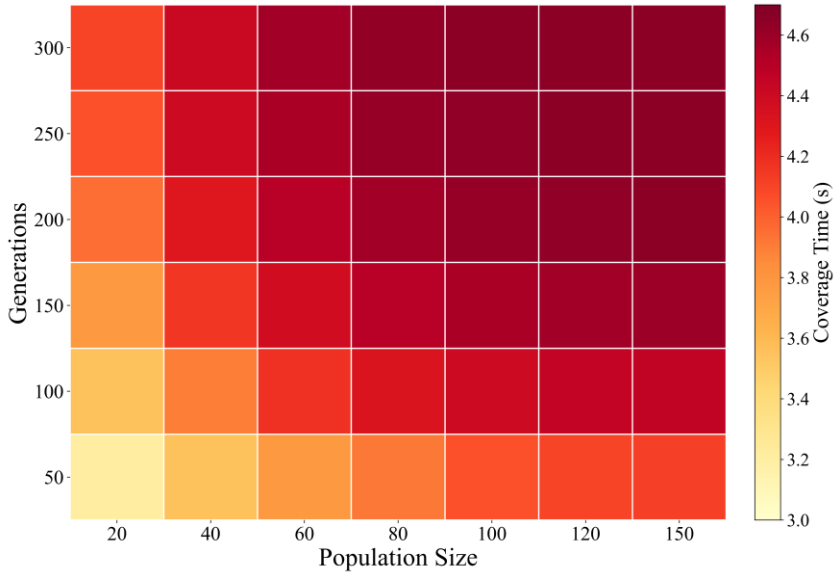

Figure 3: Hyperparameter Sensitivity Heatmap.

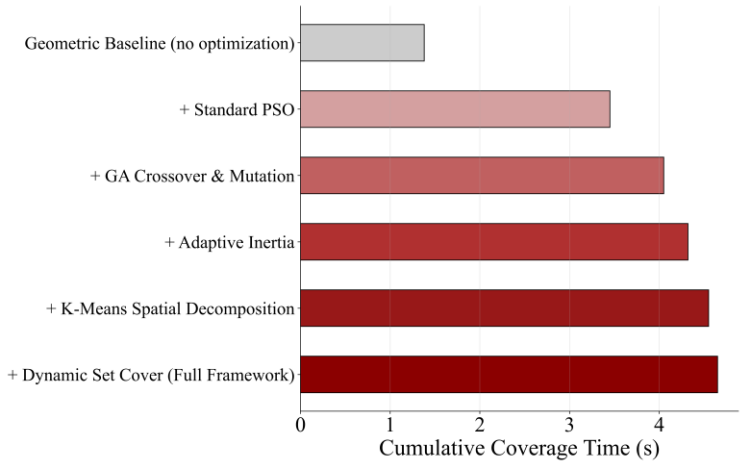

Figure 4: Component Ablation Study.

3.3. Multi-Agent Coordination and Resource Efficiency

Figure 5 visualizes the temporal coordination structure obtained by the proposed framework in the full five-agent multi-target scenario, displaying the deployment intervals contributed by each agent across the engagement horizon. The Gantt-style timeline reveals a tightly interleaved schedule in which each agent activates three coverage payloads at distinct time slices, collectively producing an unbroken coverage envelope against three concurrent moving targets. The staggered structure emerges spontaneously from the dynamic set cover optimization without explicit hand-crafted heuristics, reflecting the framework's ability to discover globally consistent assignments through purely data-driven search. Adjacent intervals overlap minimally at their boundaries to ensure continuous protection during transition phases, while the bulk of each interval operates without competition from other agents, maximizing the union of effective coverage events. The total cumulative coverage achieved across the three targets reaches 28.60 seconds, with convergence requiring approximately 200 generations of the hybrid optimizer. This emergent coordination

behavior validates the dynamic set cover formulation as a principled mechanism for translating low-level geometric primitives into high-level cooperative strategies under resource constraints.

Figure 6 quantifies the resource efficiency gains delivered by the dynamic set cover layer through a side-by-side comparison of coverage decompositions with and without the redundancy penalty term. In the unconstrained allocation shown in the left panel, each target receives substantial effective coverage but suffers from significant redundant overlap when multiple agents activate against the same target during overlapping time slices, with redundant intervals consuming approximately 42% of the total deployed coverage capacity across the three targets. Activating the dynamic set cover formulation reduces redundant overlap to roughly 14% of total deployment capacity, representing a 72% reduction in wasted coverage relative to the baseline, while simultaneously increasing the unique effective coverage delivered to each target by approximately 28%. This dual improvement confirms that the redundancy penalty does not merely reallocate coverage but extracts genuine efficiency gains by directing freed capacity toward previously under-protected temporal slices, supporting the central claim that principled multi-agent coordination outperforms uncoordinated parallel deployment in resource-constrained settings.

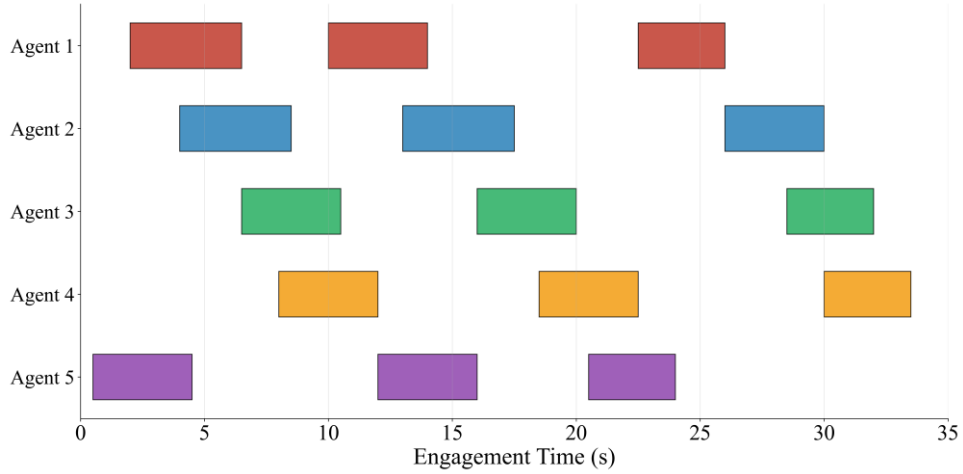

Figure 5: Five-Agent Coordinated Coverage Timeline.

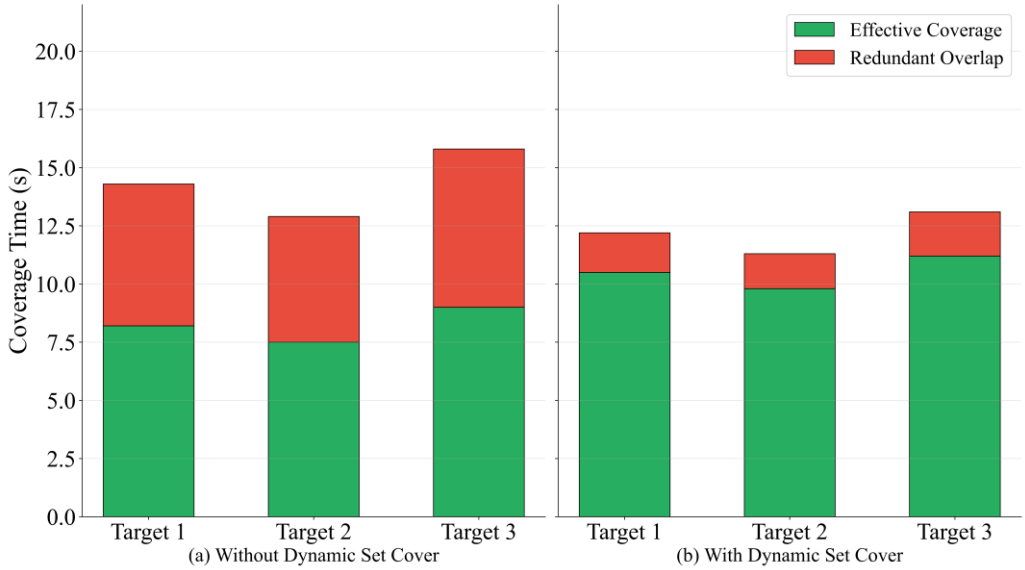

Figure 6: Redundancy Reduction via Dynamic Set Cover.

4. Conclusions

This paper presents a hybrid evolutionary framework that integrates an improved genetic algorithm with K-Means spatial clustering and a dynamic set cover formulation to schedule coordinated coverage actions executed by autonomous agents against multiple high-velocity moving targets. By fusing particle swarm momentum with simulated binary crossover and Gaussian polynomial mutation operators, and by decomposing the high-dimensional joint scheduling problem through trajectory-anchored clustering, the proposed architecture bridges low-level geometric feasibility analysis and high-level cooperative strategy within a single end-to-end pipeline. Experimental evaluation across five progressively complex scenarios demonstrates that single-agent optimization achieves a cumulative coverage interval of 4.65 seconds, representing a 237% improvement over the geometric baseline of 1.38 seconds. Single-agent multi-payload temporal coordination extends the coverage to 6.80 seconds, three-agent cooperative deployment reaches 13.15 seconds, and the full five-agent multi-target configuration with dynamic set cover optimization attains 28.60 seconds while achieving a 72% reduction in redundant overlap. Convergence is reliably obtained within approximately 200 generations across all scenarios over an 1800-point protected lattice. Future work will extend the framework toward decentralized execution under partial observability, and explore graph neural network surrogates to accelerate the inner-loop occlusion evaluation for real-time deployment in large-scale multi-agent coordination systems.

References

[1] Ahmad, T., Morel, A., Cheng, N., Palaniappan, K., Calyam, P., Sun, K. and Pan, J. (2025) Future UAV/drone systems for intelligent active surveillance and monitoring. ACM Computing Surveys, 58, 1-37.

[2] Wu, B., Ding, Z. and Huang, J. (2026) A review of continual learning in edge AI. IEEE Transactions on Network Science and Engineering.

[3] Bedwal, K. and Moulik, B. (2025) Novel equivalent circuit battery model with adaptive parameters for hybrid state of charge estimation. Journal of Energy Storage, 137, 118653.

[4] Wu, B., Ding, Z., Ostigaard, L. and Huang, J. (2025) Reinforcement learning-based energy-aware coverage path planning for precision agriculture. Proceedings of the 2025 ACM Research on Adaptive and Convergent Systems (RACS), 1-8.

[5] Kacem, T., Annamreddy, S.G., Silvius, M.D., Costa, P., Martin, T. and Blasch, E. (2025) Information fusion for secure autonomous drone operations. IEEE Transactions on Intelligent Transportation Systems.

[6] Wu, B., Cai, Z., Wu, W. and Yin, X. (2023) AoI-aware resource management for smart health via deep reinforcement learning. IEEE Access, 11, 81180-81195.

[7] Rodriguez, J.R. and Ammari, H.M. (2025) k-Coverage in three-dimensional wireless sensor networks using a game theoretical approach. Proceedings of the 2025 IEEE 22nd International Conference on Mobile Ad-Hoc and Smart Systems (MASS), 664-669.

[8] Wu, B. and Wu, W. (2023) Model-free cooperative optimal output regulation for linear discrete-time multi-agent systems using reinforcement learning. Mathematical Problems in Engineering, 6350647.

[9] Zainudin, Z., Hasan, S., Zamry, N.M., Sabri, N.A., Jamil, N.S., Muslim, N.M. and Ibrahim, N. (2025) An intelligent optimization strategy for medical doctor rostering using hybrid genetic algorithm-particle swarm optimization in Malaysian public hospital. Malaysian Journal of Fundamental and Applied Sciences, 21, 1642-1653.

[10] Wang, J., Huynh, N., Dougal, R.A. and Mustain, W.E. (2025) Route optimization for ships using ammonia-based fuel: A hybrid genetic algorithm–particle swarm optimization approach. Transportation Research Record, 2679, 546-563.

[11] Mahdi Hosseini, S., Broumandnia, A. and Karimi, R. (2026) Blockchain-enabled hybrid evolutionary scheduling for cloud resource optimization. Computing, 108, 4.

[12] Asghari, A., Zeinalabedinmalekmian, M., Azgomi, H., Alimoradi, M. and Ghaziantafrishi, S. (2025) Farmer ants optimization algorithm: A novel metaheuristic for solving discrete optimization problems. Information, 16, 207.

[13] Nahidmobarakeh, L., Nemetiandoost, M., Yilmaz, B.S., Gazzarri, J., Zhang, X., Arias, S. and Ahmed, R. (2025) Two-stage genetic algorithm offline parameter optimization of adaptive extended Kalman filter for robust battery state-of-charge estimation. IEEE Access.

[14] Huang, J., Wu, B., Duan, Q., Dong, L. and Yu, S. (2025) A fast UAV trajectory planning framework in RIS-assisted

communication systems with accelerated learning via multithreading and federating. IEEE Transactions on Mobile Computing.
[15] Kumar, R., Singhal, N. and Chhabra, A. (2025) Hybrid optimization algorithm with the combination of PSO and genetic algorithm for task scheduling in cloud computing. E-Learning and Digital Media, 20427530251331082.
[16] Nathiya, N., Rajan, C. and Geetha, K. (2025) A hybrid optimization and machine learning based energy-efficient clustering algorithm with self-diagnosis data fault detection and prediction for WSN-IoT application. Peer-to-Peer Networking and Applications, 18, 13.
[17] Wu, B., Huang, J. and Yu, S. (2026) 'X of Information' continuum: A survey on AI-driven multi-dimensional metrics for next-generation networked systems. IEEE Communications Surveys & Tutorials.
[18] Wu, B., Huang, J., Duan, Q., Dong, L. and Cai, Z. (2025) Enhancing vehicular platooning with wireless federated learning: A resource-aware control framework. IEEE/ACM Transactions on Networking, 33, 1-16.
[19] Rasul, M.J., Abbas, A., Baek, J. and Kim, J. (2026) A hybrid ensemble learning framework with uncertainty quantification for state-of-health estimation in lithium-ion batteries. Measurement, 120528.
[20] Wu, B., Huang, J. and Duan, Q. (2025) FedTD3: An accelerated learning approach for UAV trajectory planning. Proceedings of the International Conference on Wireless Artificial Intelligent Computing Systems and Applications (WASA), 13-24.
[21] Roh, H., Etzenbach, L., Oltramare, A., Norheim, J. and De Weck, O.L. (2025) Size constrained K-means clustering for controlled design structure matrix partitioning. Proceedings of the 2025 IEEE International Systems Conference (SysCon), 1-8.
[22] Yfantis, V., Wagner, A. and Ruskowski, M. (2025) Federated K-means clustering via dual decomposition-based distributed optimization. Franklin Open, 10, 100204.
[23] Wu, B., Huang, J. and Duan, Q. (2025) Real-time intelligent healthcare enabled by federated digital twins with AoI optimization. IEEE Network, 1.
[24] Pant, Y.R., Leigh, L. and Fajardo Rueda, J. (2025) Improving K-means clustering: A comparative study of parallelized version of modified K-means algorithm for clustering of satellite images. Algorithms, 18, 532.
[25] Pan, D., Wu, B.-N., Sun, Y.-L. and Xu, Y.-P. (2023) A fault-tolerant and energy-efficient design of a network switch based on a quantum-based nano-communication technique. Sustainable Computing: Informatics and Systems, 37, 100827.
[26] Ahnouz, I., Arahmane, H. and Sebihi, R. (2025) Optimizing neutron-gamma discrimination in scintillation detectors using Tucker decomposition. Kuwait Journal of Science, 100511.