ICAISC 2025

The 24th International Conference on Artificial Intelligence and Soft Computing

June 23, 2025

Full paper

Migration Timing in Hybrid Island-Based Metaheuristic Algorithms

Adam Żychowski[1], Xin Yao[2], Jacek Mańdziuk[1,3]

[1]Warsaw University of Technology, Faculty of Mathematics and Information Science
[2]Lingnan University, School of Data Science
[3]AGH University of Krakow, Faculty of Computer Science

Abstract: This paper investigates the critical aspect of migration timing in hybrid island-based metaheuristic algorithms. Migration timing plays a pivotal role in balancing exploration and exploitation, ensuring that the algorithm avoids premature convergence while effectively exploring the search space. We propose and evaluate several migration timing strategies, including periodic migration, fitness-based triggers, and diversity-driven approaches. Our experiments are conducted on a set of benchmark optimization problems, including both discrete (Traveling Salesman Problem) and continuous (Black-box Optimization Benchmarking) tasks. The results demonstrate that adaptive migration strategies, which dynamically adjust based on population diversity and fitness stagnation, outperform static approaches. This study provides insights into the optimal conditions for triggering migration and offers guidelines for designing more effective hybrid metaheuristic frameworks.

Cooperating portfolio of metaheuristics

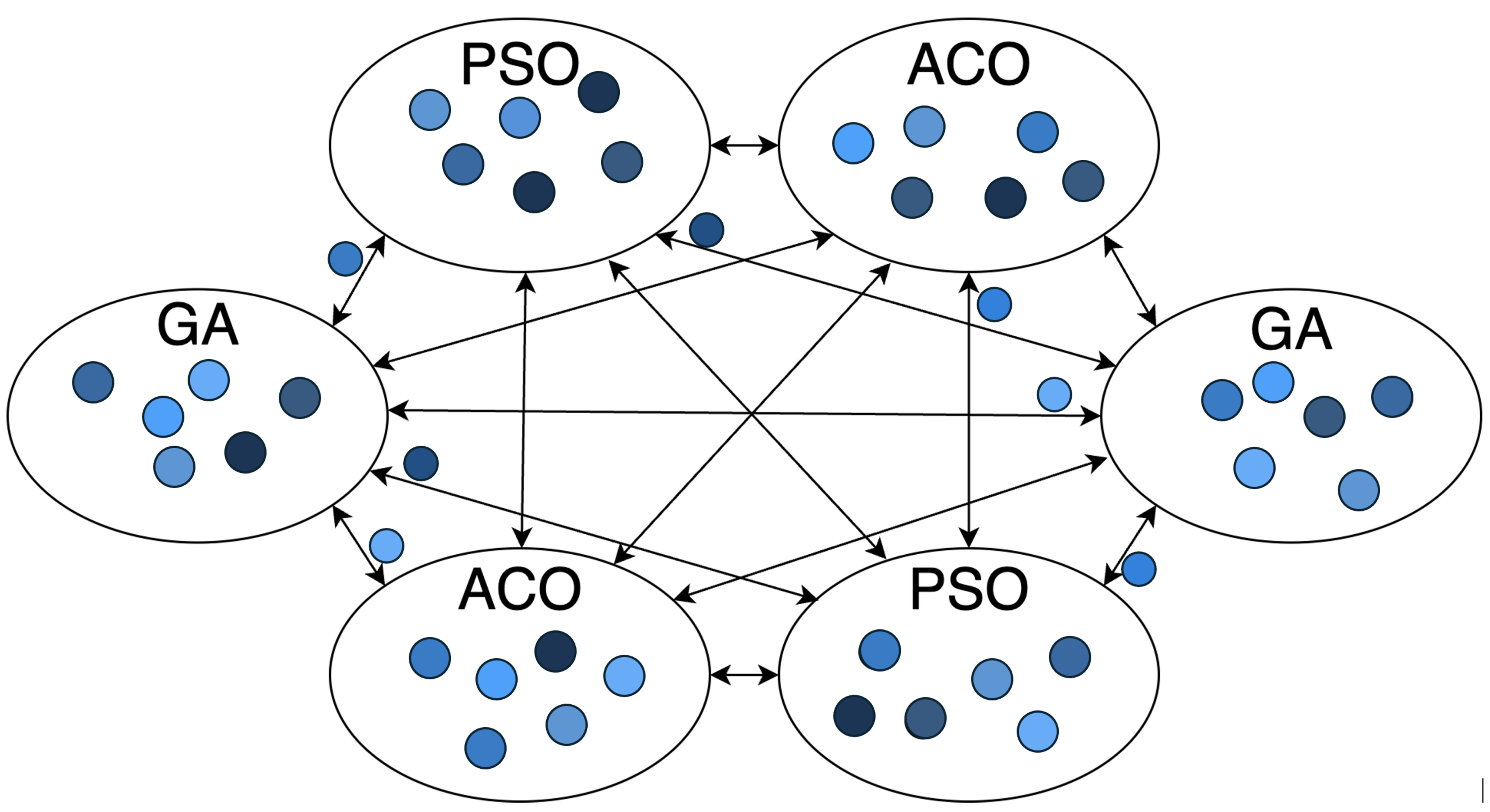

Figure: Overview of the baseline DdCPM framework used for testing migration strategies. Populations in multiple islands are developed by different metaheuristics, with migrations between islands governed by certain migration triggers. A degree of shading indicates an individual's fitness value.

Żychowski, A., Yao, X., Mańdziuk, J.: Diversity-driven cooperating portfolio of metaheuristic algorithms. The Genetic and Evolutionary Computation Conference (GECCO'25) [accepted].

Algorithm 1 Pseudocode of the baseline DdCPM method.

for each island $I \in \mathcal{I}$ **do**
  initialize $I_{population}$ with random individuals
end for
while $evaluation\_budget > 0$ **do**
  for each island $I \in \mathcal{I}$ **do**
    if needs_migration(I) **then**
      $I_{population} = I_{population} \cup$ migrate_from($I_{neighbours}$)
    end if
  end for
  for each island $I \in \mathcal{I}$ **do**
    $I_{population}$ = next_generation($I_{population}$, $I_{metaheuristic}$)
    evaluate($I_{population}$)
    $evaluation\_budget = evaluation\_budget - |I_{population}|$
  end for
end while
return the best individual from $\mathcal{I}$

Key research questions

- How frequently should migration happen?
- What indicators (metrics) trigger migration?
- How does migration timing affect performance?

Migration timing strategies

- **T1: Periodic Migration**: Migration occurs at fixed intervals (every m iterations), regardless of population state.
- **T2: Randomized Migration**: Migration happens each generation with a fixed probability ($m\%$).
- **T3: Average Fitness Stagnation**: Triggered if average fitness doesn't improve for m consecutive iterations.
- **T4: Maximum Fitness Stagnation**: Triggered if the best fitness in the population doesn't improve for m iterations.
- **T5: Combined Fitness Stagnation**: Triggered if both average and maximum fitness fail to improve for m iterations.
- **T6: Diversity Stagnation**: Triggered if diversity (based on standard deviation across dimensions) doesn't improve for m iterations.
- **T7: Combined Fitness and Diversity Stagnation**: Triggered if either both fitness measures stagnate or diversity stagnates.
- **T8: Sum of Normalized Fitness and Diversity Stagnation**: Triggered if the normalized sum of fitness and diversity remains unchanged for m iterations.

Results

TSP	Migration timing strategy							
m	T1	T2	T3	T4	T5	T6	T7	T8
1	36205 ± 47	36536 ± 59	**35832** ± 88	35758 ± 60	35683 ± 55	**35671** ± 70	35646 ± 67	35716 ± 94
2	36193 ± 81	36211 ± 53	35857 ± 67	35745 ± 64	35671 ± 79	35696 ± 58	35634 ± 94	35661 ± 72
5	36168 ± 56	35894 ± 64	35944 ± 92	35721 ± 78	35621 ± 90	35745 ± 52	**35547** ± 79	35644 ± 81
10	36093 ± 76	**35625** ± 47	36019 ± 89	**35621** ± 66	**35559** ± 59	35820 ± 53	**35547** ± 65	**35619** ± 85
15	**35634** ± 70	35866 ± 58	36056 ± 50	35721 ± 71	35708 ± 80	35795 ± 75	35658 ± 90	35715 ± 68
20	35857 ± 91	36170 ± 61	36081 ± 80	35758 ± 93	35708 ± 55	35894 ± 47	35683 ± 88	35765 ± 79
25	36118 ± 67	36182 ± 73	36106 ± 54	35770 ± 68	35708 ± 69	35857 ± 78	35696 ± 60	35714 ± 84
30	36180 ± 60	36248 ± 57	36106 ± 76	35783 ± 91	35721 ± 61	35919 ± 69	35696 ± 65	35816 ± 81
50	36242 ± 81	36286 ± 64	36143 ± 93	35783 ± 54	35733 ± 88	35919 ± 55	35708 ± 80	35778 ± 92

BBOB	Migration timing strategy							
m	T1	T2	T3	T4	T5	T6	T7	T8
1	2.808 ± .12	2.871 ± .09	**2.830** ± .11	1.605 ± .05	1.716 ± .07	**2.585** ± .10	1.515 ± .04	1.580 ± .06
2	2.808 ± .11	2.792 ± .08	2.875 ± .12	1.605 ± .06	1.716 ± .08	2.607 ± .09	1.515 ± .05	1.529 ± .07
5	2.786 ± .10	2.690 ± .07	2.986 ± .13	1.582 ± .05	1.659 ± .06	2.674 ± .11	1.493 ± .04	1.509 ± .05
10	2.697 ± .09	**2.241** ± .06	3.075 ± .14	**1.560** ± .04	**1.649** ± .07	2.674 ± .10	**1.449** ± .03	**1.497** ± .05
15	**2.229** ± .08	2.452 ± .07	3.120 ± .15	1.605 ± .05	1.716 ± .08	2.741 ± .12	1.515 ± .04	1.553 ± .06
20	2.451 ± .10	2.721 ± .09	3.142 ± .16	1.605 ± .06	1.738 ± .07	2.741 ± .11	1.515 ± .05	1.579 ± .07
25	2.719 ± .11	2.808 ± .10	3.165 ± .17	1.627 ± .05	1.738 ± .08	2.786 ± .13	1.519 ± .04	1.563 ± .06
30	2.786 ± .12	2.854 ± .11	3.187 ± .18	1.627 ± .06	1.738 ± .09	2.741 ± .14	1.525 ± .05	1.601 ± .07
50	2.853 ± .13	2.872 ± .12	3.231 ± .19	1.627 ± .07	1.738 ± .10	2.808 ± .15	1.522 ± .06	1.537 ± .08

Table: Averaged results for various m values and different migration timing strategies for BBOB and TSP instances.

Summary

- Adaptive migration outperforms static or random timing.
- Combining fitness and diversity (T7) yields the best results.