Minimizing Autonomous Vehicle Fleet Size for Dynamic Ride-Sharing using the Squirrel Search Algorithm

Sarthak
Department of Computer Science
Indian Institute of Information Technology Guwahati
sarthak22b@iiitg.ac.in

***Abstract*—We explored how to minimize the number of self-driving cars on the road for a ride-sharing service. This work was done using a computer-assisted method known as the Squirrel Search Algorithm (SSA) based on the behavior of squirrels searching for food. We modified this algorithm so that it accommodates the most important rules for a ride-share service, including passenger pickup and dropoff locations, maximum waiting time, and total number of passengers in the vehicle. Our experiments showed that the SSA demonstrated acceptable performance with fewer vehicles while not making the service worse overall for the riders.**

***Index Terms*—autonomous vehicles, ride-sharing, squirrel search algorithm, fleet size optimization**

I. Introduction

With the growing popularity of ride-sharing, we now have new opportunities to make transport systems function better, particularly through autonomous vehicles. A major problem is determining the minimum number of vehicles to cater to all requests for rides without fulfilling different needs. Operating with fewer vehicles reduces operating expenses as well as environmental damage.

Nature-based search algorithms have been found to perform well in solving complex optimization problems. The recently designed Squirrel Search Algorithm performs well by emulating the behavior of flying squirrels when they search for food. However, to use SSA in minimizing vehicle quantities, we must make substantial adjustments to incorporate the special characteristics of ride-sharing systems.

In this work, we introduce a method based on the Squirrel Search Algorithm tailored to the minimization of self-driving vehicle fleets. Our method features transport-specific characteristics and adaptive search mechanisms that balance the exploration of novel strategies and enhancing current ones in order to obtain high-quality solutions using the smallest number of vehicles.

II. Problem Statement

Given a set of ride requests $R = \{r_1, r_2, \ldots, r_n\}$, we need to figure out the minimum number of vehicles that can handle all these requests while meeting time requirements.

Each request r_i is characterized by $(P_i, D_i, T_i, W_i, E_i, Ex_i)$, where P_i is pickup location, D_i is destination, T_i is request time, W_i is maximum waiting time, E_i is direct travel time, and Ex_i is maximum extra travel time.

Constraints: (1) waiting time $\leq W_i$, (2) extra travel time $\leq Ex_i$, (3) vehicle capacity = 4, (4) every single request gets served.

III. Solution/Approach

Our use of SSA is inspired on how flying squirrels seek out for food in winter. When we say they are foraging for food instead of searching cars, we frame the search for the minimum self-driving cars as the flying squirrel searching for the best food sources. Using less self-driving cars signifies the fleet is being effectively utilized.

A. Solution Representation

Each flying squirrel represents a complete solution comprising:

- A set of active autonomous vehicles $AV = \{av_1, av_2, \ldots, av_m\}$
- For each vehicle av_j, a ride-sharing route $RS_j = \{(l_1, a_1, t_1), (l_2, a_2, t_2), \ldots\}$ where l_k is location, a_k is action (pickup/dropoff), and t_k is time

B. Fitness Evaluation

The solutions are evaluated using:

$$f(FS) = |AV| + \lambda \cdot \sum_{i=1}^{n} \mathrm{C}(r_i) \tag{1}$$

where $|AV|$ is the number of autonomous vehicles used and λ is a penalty coefficient for constraint violations.

C. Adapted Movement Strategies for Ride-Sharing

We adapt the three movement cases from the original SSA, reinterpreting them for the ride-sharing context:

1) **Fleet Consolidation Movement**: Solutions with too many cars move toward more efficient setups:

$$|AV_{sub}^{t+1}| = |AV_{sub}^{t}| - d_g \times G_c \times \Delta AV_{opt,sub} \tag{2}$$

$$RS_{sub}^{t+1} = RS_{sub}^{t} + d_g \times G_c \times (RS_{opt}^{t} - RS_{sub}^{t}) \tag{3}$$

Where $|AV_{sub}^{t}|$ represents the number of autonomous vehicles in the suboptimal solution, $\Delta AV_{opt,sub}$ is the fleet size difference between optimal and suboptimal solutions, and RS_{sub}^{t}

represents the ride-sharing route structure. This is implemented through strategic vehicle consolidation and ride reassignment.

2) **Passenger Experience Optimization**: Solutions with inefficient routes move toward better ride quality:

$$RS_{int}^{t+1} = RS_{int}^{t} + d_g \times G_c \times (RS_{sub}^{t} - RS_{int}^{t}) \quad (4)$$

$$DT_{int}^{t+1} = DT_{int}^{t} - d_g \times G_c \times \Delta DT_{sub,int} \quad (5)$$

Where RS_{int}^{t} represents ride-sharing routes in intermediate solutions, DT_{int}^{t} represents passenger detour time, and $\Delta DT_{sub,int}$ is the difference in detour times between solutions. We do this by reassigning which car picks up which person and making route changes to give passengers a better experience.

3) **Global Ride-Sharing Pattern Adoption**: Solutions adopt effective ride-sharing patterns from the best solution:

$$|AV_{int}^{t+1}| = |AV_{int}^{t}| - d_g \times G_c \times \Delta AV_{opt,int} \quad (6)$$

$$RG_{int}^{t+1} = RG_{int}^{t} + d_g \times G_c \times (RG_{opt}^{t} - RG_{int}^{t}) \quad (7)$$

Where $|AV_{int}^{t}|$ is the number of vehicles in intermediate solutions, $\Delta AV_{opt,int}$ is the fleet size difference between optimal and intermediate solutions, and RG_{int}^{t} represents how we group rides together. We do this by completely rearranging the solution, using the way vehicles are combined and rides are grouped from the best solution we've found so far.

D. Transportation-Specific Operators

We introduce specialized operators for ride-sharing optimization within the SSA framework:

1) **Request Clustering**: Grouping ride requests that are close in time and/or locations to fit more people in a car.

2) **Chain Formation**: Linking requests where the drop-off location is close to the next pickup location to minimize empty car driving time.

3) **Time Window Utilization**: Adjusting pickup times, within established time windows, to generate more potential for sharing.

4) **Efficient Request Insertion**: Finding optimal positions to add new requests into existing routes while minimizing detour time.

E. Diversification & Convergence Control

With probability P_{dp} (disruption probability), we simulate service disruptions by simulating service changes (moving passengers between cars, splitting overfull routes, combining cars that aren't carrying many people) to escape local minima in the fleet size search.

We calculate the route similarity metric S_c to assess ride-sharing pattern diversity:

$$S_c = \sqrt{\sum_{k=1}^{d} (RS_{sub,k} - RS_{opt,k})^2} \quad (8)$$

where d represents the number of pattern features we're looking at, $RS_{sub,k}$ and $RS_{opt,k}$ are the k-th elements of suboptimal and optimal ride-sharing patterns respectively.

If $S_c < S_{min}$ (diversity threshold), we trigger a big reshuffling of the whole system, moving cars and reassigning passengers using a special exploration method (called Lévy flight) to discover completely new ride-sharing setups. This prevents premature convergence to suboptimal fleet allocations.

IV. Algorithm

Initialize population of ride-sharing solutions (squirrels)
Evaluate fitness using Eq. (1) and classify solutions (optimal, suboptimal, intermediate)
while termination criteria not met **do**
    Apply adapted movement strategies (Sec. III-C) with transportation operators (Sec. III-D)
    Apply predator presence (diversification) with probability P_{dp}
    Repair constraint violations and evaluate fitness
    Update solution classification
    Calculate S_c; if $S_c < S_{min}$ apply Lévy flights (Sec. III-E)
end while
return best solution found (minimum $|AV|$ satisfying constraints)

V. Results and Analysis

The Squirrel Search Algorithm was evaluated across varying parameters to assess its performance in minimizing autonomous vehicle fleet size for the dynamic ride-sharing problem.

A. Experimental Results

TABLE I
EFFECT OF NETWORK SIZE (FIXED: 30 REQUESTS, 15 TIME UNITS)

Nodes	10	20	30	40	50
Vehicles	3	4	6	6	7
Wait Time (avg)	8.10	8.47	11.13	10.80	8.97

Finding: Each additional 10 nodes requires approximately 1 more vehicle due to increased travel distances.

TABLE II
EFFECT OF REQUEST DENSITY (FIXED: 20 NODES, 20 TIME UNITS)

Requests	20	40	60	80	100
Vehicles	3	8	10	16	20
Wait Time (avg)	9.00	6.00	8.53	6.55	8.22

Finding: Vehicle requirements scale nearly linearly with request volume, maintaining a consistent vehicle-to-request ratio of approximately 1:5.

TABLE III
EFFECT OF TIME PERIOD (FIXED: 20 NODES, 50 REQUESTS)

Time Period	10	20	30	50	100
Vehicles	10	10	10	6	4
Wait Time (avg)	7.26	8.06	8.00	8.90	5.88

B. Key Insights

Our experimental results highlight several important characteristics of the Squirrel Search Algorithm in this application:

1) **Network scaling**: The algorithm maintains efficiency as network size increases, with predictable vehicle requirements.
2) **Request handling**: A consistent 1:5 vehicle-to-request ratio is maintained across scenarios.
3) **Temporal impact**: Doubling the time period from 50 to 100 reduces vehicle requirements by 33%.
4) **Wait times**: Service quality remains relatively consistent (average wait times mostly between 6-9 minutes, with one higher outlier) regardless of parameter changes.

These results demonstrate the algorithm's effectiveness in minimizing fleet size while maintaining reasonable wait times across diverse operating conditions.

VI. CONCLUSION AND FUTURE WORKS

We have applied the Squirrel Search Algorithm to reduce the number of self-driving cars needed for dynamic ride-sharing. Our method incorporates transportation-specific operators and adaptation strategies within the SSA framework, finding a good balance between exploring new solutions and refining existing ones to minimize fleet size effectively.

Future work will explore: (1) adaptive fleet composition with heterogeneous vehicle types; (2) predictive demand modeling to preposition vehicles; (3) transfer point optimization for efficient route connections; and (4) resilient fleet management for handling disruptions. These enhancements would further improve autonomous ride-sharing systems in complex urban environments.

REFERENCES

[1] J. Alonso-Mora, S. Samaranayake, A. Wallar, E. Frazzoli, and D. Rus, "On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment," *Proceedings of the National Academy of Sciences*, vol. 114, no. 3, pp. 462–467, 2017.

[2] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[3] M. Jain, V. Singh, and A. Rani, "A novel nature-inspired algorithm for optimization: Squirrel search algorithm," *Swarm and Evolutionary Computation*, vol. 44, pp. 148–175, 2019.