

Agen Otonom Untuk Ketepatan Penembakan Pada Game Roket Berbasis Ant Colony Optimization

Septian Enggar Sukmana^{1*}, Ely Nur Rahayu²

¹Program Studi D4 Teknik Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang

²Program Studi D3 Manajemen Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Malang

*Corresponding Author

E-mail: enggar@polinema.ac.id*

Abstrak

Ant Colony Optimization (ACO) berbasis klasifikasi diterapkan pada game Rocket Tank untuk membantu karakter roket dalam menembak target (tower) secara tepat. Pada paper ini, roket berfungsi sebagai sarang semut dan tower sebagai sumber makanan. Agen semut dijalankan secara acak dan berjalan menemukan sumber makanan. Sumber makanan yang ditemukan didefinisikan sebagai target yang harus ditembak, dasar dari penentuan sumber makanan adalah jumlah bobot feromon yang terkumpul tanpa nilai jarak. Implementasi yang dilakukan pada paper ini adalah membandingkan penerapan ACO berbasis klasifikasi dalam melibatkan bobot feromon pada sumber makanan dengan ACO state-of-the art dengan hasil pada 100 kali iterasi adalah ACO berbasis klasifikasi mampu menembak secara tepat pada sasaran dibandingkan dengan ACO state-of-the art dengan peningkatan jumlah tembakan yang lebih banyak pada ACO berbasis klasifikasi dibandingkan ACO state-of-the art.

Kata Kunci: ant colony optimization, game, simulasi game, agen otomatis

Abstract

Classification-based Ant Colony Optimization (ACO) is applied to the Rocket Tank game to help rocket characters shoot targets (towers) accurately. In this paper, rockets function as ant nests and towers as a food source. The ant agent is run randomly and runs to find a food source. The food source found is defined as the target that must be shot, the basis for determining the source of food is the amount of pheromone weight collected regardless of distance value. The implementation carried out in this paper is to compare the application of classification-based ACO in involving pheromone weights at food sources with state-of-the-art ACO with the result that 100 iterations are classification-based ACOs being able to shoot precisely at the target compared to state-of-the-art ACO. the art with an increased number of shots fired on a classification-based ACO than a state-of-the art ACO.

Keywords: ant colony optimization, game, game simulation, auto agent

1. PENDAHULUAN

Game menjadi salah satu tren yang tinggi penelitian pada era ini. Pemanfaatan game tidak hanya untuk tujuan permainan semata, namun juga sebagai pemodelan, media bantu atau media pembelajaran, dan juga sebagai media untuk simulasi suatu kondisi. Realisasi dari pengembangan supaya lebih implementatif biasanya menerapkan kecerdasan buatan sebagai komponen internal game tersebut. Salah satu penerapan kecerdasan buatan di dalam game adalah metode *ant colony optimization* (ACO) dengan beberapa penerapan seperti *denoising*[1], sistem interaktif[2],

optimalisasi daratan pada sistem GIS[3]. Penerapan metode ini juga diterapkan di dalam game seperti tetris[4].

Penggunaan teknologi AI dan pembelajaran mesin dalam *game* semakin banyak digunakan, namun untuk game yang serius seperti game yang bergantung pada kemampuan *player* dalam pengembangan NPC cerdas seperti pada penelitian darmawan dkk [5] yang menitikberatkan pada pengembangan NPC cerdas yang dapat beradaptasi dengan gaya pemain dalam bermain game dimana NPC bisa berevolusi dengan perilaku adaptif seperti mengubah kecepatan dan tingkat kesehatan dengan melihat perilaku pemain. Namun, penggunaan yang menambah kekuatan teknologi AI dengan intuisi manusia dan kognisi manusia misalkan untuk memecahkan permasalahan yang kompleks misalkan masalah optimasi untuk mencari jalur yang terinspirasi koloni semut seperti penggunaan algoritma *Ant Colony Optimization* seperti pada *Improved modeling of intelligent tutoring systems using ant colony optimization*[6] dimana menggunakan perilaku semut dalam koloni semut buatan untuk menyelesaikan masalah pengoptimalan diskrit contohnya untuk pembelajaran *adaptive tutoring systems* menggunakan ACO. Penelitian ACO pada game lainnya misalkan pada penelitian Romero dan Ventura dkk yang menggunakan *artificial intelligent* dalam pencarian jalan misalkan untuk mengontrol NPC dalam menemukan jalur yang optimal melalui elemen *game*, pembelajaran *player* dan penilaian[7]. Namun penelitian lainnya, yang khusus menggunakan *ant colony optimization* dalam permainan game diantaranya michael d. kickmeier-Rust dkk dimana penggunaan algoritma ACO dipakai untuk mendukung adaptasi dalam *game* yang serius (kompleks)[8], penelitian G.Recio dkk[9] melakukan pengembangan kecerdasan swarm dengan pengoptimalkan *ant-colony-based framework* pada *video game* dimana kerangka kerja game diadaptasi untuk memungkinkan penerapan agen cerdas misalkan dalam permainan *Pac-Man*. Contoh penggunaan *Ant Colony Optimization* di game yang diaplikasikan untuk pencarian jarak pada games dapat ditemukan pada penelitian [10] dan [11].

Paper ini menjelaskan penerapan ACO pada game rocket tank, hal yang menjadi khas pada paper ini adalah penerapan metode ACO berbasis klasifikasi yang berbeda model dengan ACO *state-of-the art*. Fokus ACO berbasis klasifikasi adalah pada sumber makanan selain bobot perjalanan[1], hal ini berbeda dengan ACO *state-of-the-art* dimana ACO *state-of-the art* hanya berfokus pada bobot jalur perjalanan semut semata. Secara singkat, ACO berbasis klasifikasi menempatkan sumber makanan lebih dari satu lokasi dalam satu bidang sehingga setiap sumber makanan memiliki bobot feromon yang saling berbeda. Paper ini adalah catatan dari kegiatan penelitian dasar dari kegiatan pengembangan simulasi sistem geolokasi berbasis kecerdasan buatan. Pemilihan *game rocket tank* didasarkan pada kecocokan jenis data yang dipakai pada pengembangan sistem yang dikembangkan, pada paper ini seluruh karakter yaitu *tank* (penembak) dan *tower* (target) akan saling melakukan penyerangan secara otomatis tanpa control dari manusia.

2. METODOLOGI

Objek yang menjadi penerapan pada penelitian ini adalah *Game Rocket Tank* (Gambar 1). *Game Rocket tank* ini memiliki 1 pemain yang berupa objek *Tank*. *Tank* tersebut membawa roket di dalamnya. Pemain tersebut dapat bergerak ke segala arah (kanan kiri depan dan belakang). Selain itu terdapat juga beberapa objek yang menjadi target dalam permainan ini, ketika pemain menembakkan roket, maka objek yang menjadi target tersebut akan hilang ketika ditembak. Namun ketika pemain tidak menembakkan roket sama sekali, maka pemain tersebut akan hancur ditembak oleh objek yang menjadi target tersebut.



Gambar 1. Game Rocket Tank terdiri dari penembak (roket) dan sasaran (tower) yang saling menyerang

Skenario penelitian adalah penembakan otomatis oleh roket ke target yang dituju secara acak tujuan yang hendak ditembak. Supaya target yang disasar berhasil ditembak oleh roket, ACO diterapkan sebagai fitur pembantu roket untuk menentukan posisi target yang ditembak dan target yang menjadi sasaran pada saat itu. Sebagai ilustrasi, posisi roket dianalogikan sebagai sarang semut dan target yang dituju dianalogikan sebagai sumber makanan. Bobot paling optimal akan menjadi jarak sesuai untuk ditembak ke target.

Pada paper ini, ACO berbasis klasifikasi diterapkan pada *game Rocket Tank*. ACO berbasis klasifikasi merupakan metode ACO dengan pendekatan yang berbeda dengan ACO *state-of-the-art*, perbedaan ini terletak pada penilaian bobot tidak hanya terletak pada jalur yang dilewati semut, namun juga berfokus pada jumlah semut yang berada di lokasi sumber makanan. Untuk setiap iterasi n th, seekor agen semut berpindah dari satu *node* (koordinat sumbu x sebagai m dan y sebagai l matriks) ke *node* lain (koordinat sumbu y sebagai i dan z sebagai j matriks) secara acak.

$$p_{(l,m),(i,j)}^{(n)} = \frac{(\tau_{i,j}^{(n-1)})^\alpha (\varphi_{i,j})^\beta}{\sum_{i,j \in \Omega_{(l,m)}} (\tau_{i,j}^{(n-1)})^\alpha (\varphi_{i,j})^\beta} \quad (1)$$

$\tau_{i,j}^{(n-1)}$ adalah nilai feromon yang tersebar di setiap *node*, $\Omega_{(l,m)}$ merupakan *node* tetangga dari *node* (l,m) , α dan β didefinisikan sebagai pengaruh dari matriks feromon dan matriks *heuristic*. *Update* feromon terdiri dari dua kondisi, yaitu: (a) *update* yang terjadi setelah semua agen semut bergerak

$$\tau_{i,j}^{(n-1)} = \begin{cases} (1-p) \cdot \tau_{i,j}^{(n-1)} + p \cdot \Delta_{i,j}^{(k)} \\ \tau_{i,j}^{(n-1)} \end{cases} \quad (2)$$

Kriteria pertama ini adalah kondisi ketika seekor semut k th mengunjungi (i,j) , $\Delta_{i,j}^{(k)}$ ditentukan oleh matrik heuristik; dan (b) *update* yang terjadi setelah semua agen semut melakukan perpindahan

$$\tau^{(n)} = (1 - \psi) \cdot \tau^{(n-1)} + \psi \cdot \tau^{(0)} \quad (3)$$

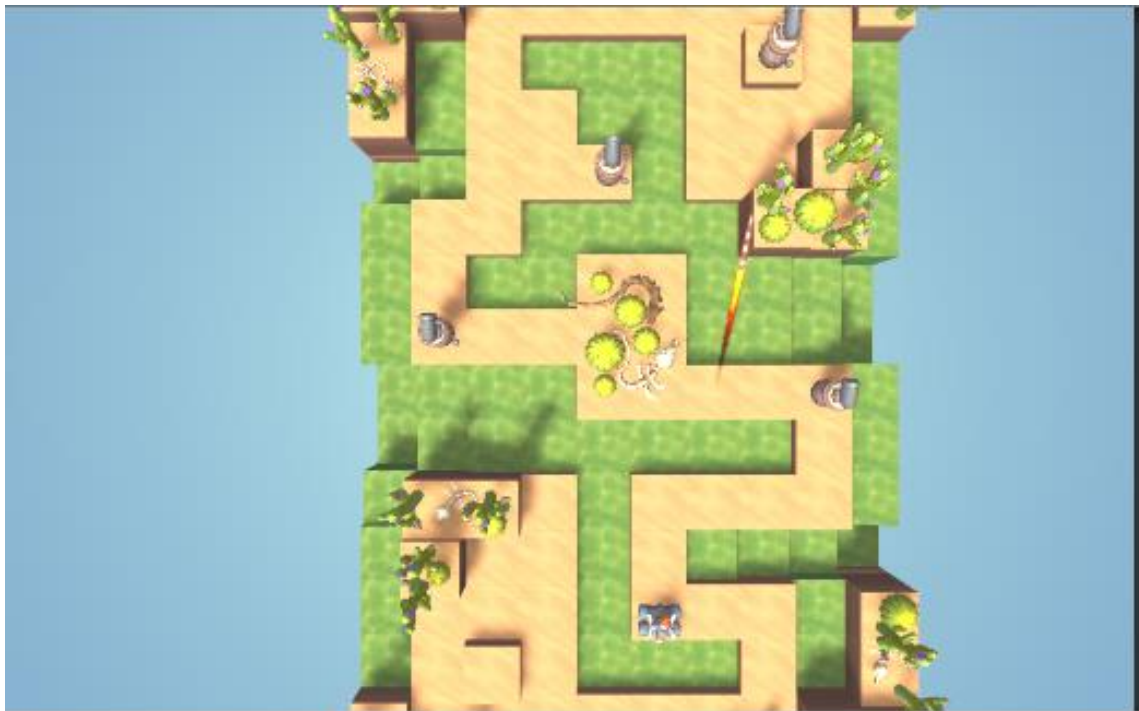
ψ adalah koefisien feromon yang menyimpang. Klasifikasi pada penelitian ini menggunakan *k-means* dengan $K = 2$.

3. HASIL DAN PEMBAHASAN

Pada kegiatan ini, jumlah agen semut yang digunakan sejumlah 100 agen. Faktor feromon diatur sebesar $\alpha = 1,5$. Secara umum, setiap target berhasil terdeteksi atau dengan kata lain agen-agen semut telah menemukan sumber-sumber makan (*tower* sebagai target) (Gambar 2). Gambar 3 menunjukkan serangan roket ke arah target sebagai hasil lokasi sumber makanan semut yang telah terdeteksi, serangan roket ternyata tidak selalu mengarah pada tower (sumber makanan) terdekat, beberapa percobaan menunjukkan tower secara acak terpilih untuk ditembak pertama kali sehingga dapat dijelaskan bahwa bobot sumber makanan yang diketahui pertama kali menjadi target serangan pertama.

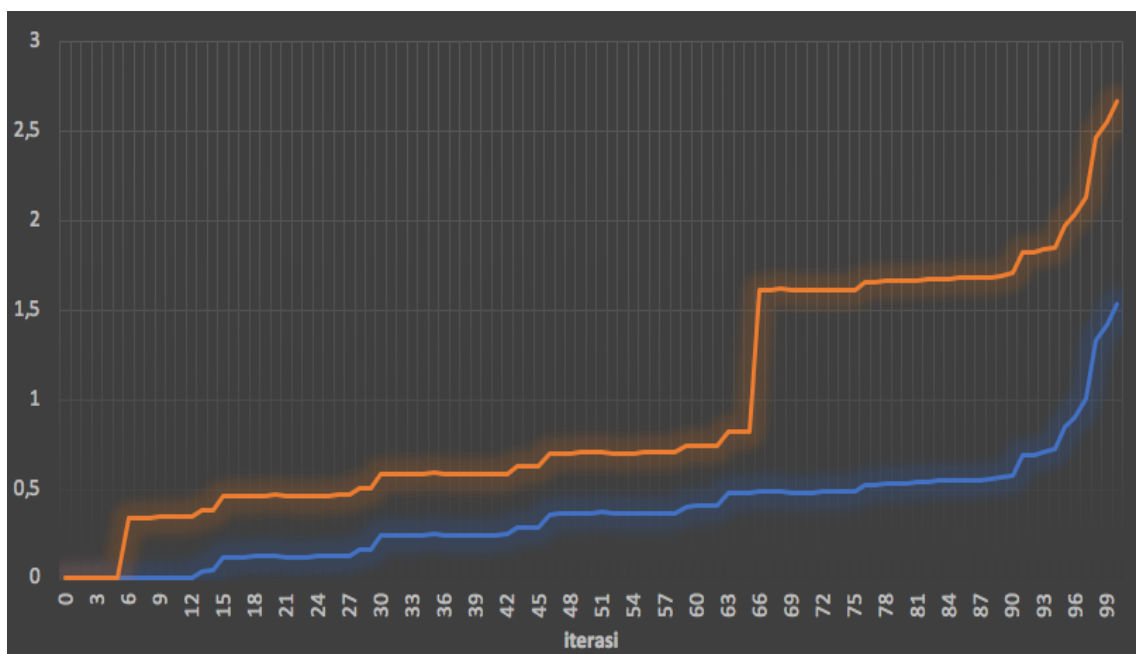


Gambar 2. Sasaran yang terdeteksi



Gambar 3. Tembakkan roket dari tank menuju target

Berdasarkan Gambar 4, penerapan *ACO state-of-the-art* dibandingkan dengan penerapan ACO berbasis klasifikasi, hasil dari penerapan ini menunjukkan bahwa dalam 100 iterasi, sebanyak 70 target berhasil tertembak dari penerapan *ACO state-of-the-art* dan 80 target berhasil tertembak dari penerapan ACO berbasis klasifikasi.



Gambar 4. Rata-rata dari akumulasi jumlah peluru roket yang jatuh ke posisi target dalam setiap iterasi (n x 10)

4. KESIMPULAN

Penerapan ACO berbasis klasifikasi pada *game Rocket Tank* memberikan hasil penembakan target yang lebih unggul dibandingkan penerapan ACO *state-of-the-art* melalui jumlah tembakan yang tepat sasaran ke lokasi target. Jumlah target juga berhasil terdeteksi melalui penerapan ACO berbasis klasifikasi ini. Berdasarkan kondisi ini, ACO berbasis klasifikasi mampu mendapatkan sumber makanan yang lebih tepat dibandingkan dengan ACO *state-of-the-art*. Namun, penerapan metode ini masih terbatas pada gerak *tank* yang masih diam sehingga perlu pengembangan teknis khusus untuk gerak dinamis pada tank. Selain itu, juga perlu pengembangan untuk penentuan target yang ditembak pertama kali dan berurutan target berikutnya.

5. REFERENCES

- [1] J. Tian, W. Yu, and L. Ma, "AntShrink: Ant colony optimization for image shrinkage," *Pattern Recognit. Lett.*, vol. 31, no. 13, pp. 1751–1758, 2010.
- [2] X. Yan, Z. Hao, H. Huang, and H. Wu, "Ant colony optimization with human-computer cooperative strategy for two-echelon vehicle routing problem," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2017, pp. 1443–1446.
- [3] X. Gao, "An improved Ant Colony Algorithm Used In Land Optimized Allocation," in *Proceedings of the 2017 International Conference on Artificial Intelligence, Automation and Control Technologies*, 2017, pp. 1–6.
- [4] X. Chen, H. Wang, W. Wang, Y. Shi, and Y. Gao, "Apply ant colony optimization to tetris," in *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, 2009, pp. 1741–1742.
- [5] D. Aditama, R. P. N. Budiarti, S. N. S. Mardi, and M. Hariadi, "Dynamic Evolution Behavior for Non-Player," vol. 5, no. 1, pp. 185–198, 2017.
- [6] M. Rastegarmoghadam and K. Ziarati, "Improved modeling of intelligent tutoring systems using ant colony optimization," *Educ. Inf. Technol.*, vol. 22, no. 3, pp. 1067–1087, 2017.
- [7] C. Romero and S. Ventura, *Data mining in e-learning*, vol. 4. WIT press, 2006.
- [8] M. Kickmeier-Rust and A. Holzinger, "Interactive Ant Colony Optimization to Support Adaptation in Serious Games," *Int. J. Serious Games*, vol. 6, no. 3, pp. 37–50, 2019.
- [9] G. Recio, E. Martin, C. Estébanez, and Y. Saez, "AntBot: Ant colonies for video games," *IEEE Trans. Comput. Intell. AI Games*, vol. 4, no. 4, pp. 295–308, 2012.
- [10] R. Du, X. Zhang, C. Chen, and X. Guan, "Path planning with obstacle avoidance in PEGs: Ant colony optimization method," in *2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, 2010, pp. 768–773.
- [11] J. A. Mocholi, J. Jaen, A. Catala, and E. Navarro, "An emotionally biased ant colony algorithm for pathfinding in games," *Expert Syst. Appl.*, vol. 37, no. 7, pp. 4921–4927, 2010.