

Smart Farming Agent Menggunakan Metode Generate and Test untuk Penyiraman Tanaman Otomatis

Septian Dwiky Nugroho*¹, Nonot Wisnu Karyanto*², Anang Kukuh Adisusilo*³

^{1,2,3} Program Studi Informatika, Universitas Wijaya Kusuma Surabaya

Email: ¹sdn035@mhs.uwks.ac.id, ²nonot@uwks.ac.id, ³anang65@uwks.ac.id

Abstrak

Perkembangan teknologi kecerdasan buatan mendorong penerapan sistem otomatis pada sektor pertanian modern. Salah satu permasalahan yang sering terjadi pada pertanian adalah proses pemantauan kelembapan tanah dan penyiraman tanaman yang masih dilakukan secara manual sehingga membutuhkan waktu dan tenaga yang besar. Penelitian ini membahas implementasi *Smart Farming Agent* berbasis kecerdasan buatan menggunakan metode *Generate and Test* untuk membantu proses penyiraman tanaman secara otomatis.

Sistem dirancang menggunakan simulasi berbasis *Python* dan *Tkinter* dengan memanfaatkan sensor kelembapan tanah, sensor posisi, dan sensor jarak. Agent bekerja dengan menghasilkan kemungkinan jalur pergerakan robot menuju tanaman yang membutuhkan air, kemudian melakukan pengujian terhadap setiap state untuk menentukan apakah tujuan telah tercapai atau belum.

Hasil penelitian menunjukkan bahwa metode *Generate and Test* mampu membantu robot menentukan urutan tindakan secara sistematis hingga seluruh tanaman memiliki tingkat kelembapan yang cukup. Sistem berhasil melakukan navigasi dari posisi awal menuju tanaman yang membutuhkan air, melakukan penyiraman, kembali ke posisi awal, serta mencapai *goal state* secara optimal.

Kata kunci: generate and test, kecerdasan buatan, robot penyiram, smart farming, state space

Abstract

The development of artificial intelligence technology encourages the implementation of automated systems in modern agriculture. One of the common problems in agriculture is monitoring soil moisture and watering plants manually, which requires considerable time and effort. This study discusses the implementation of a *Smart Farming Agent* using the *Generate and Test* method to support automatic plant watering.

The system was designed using a *Python* and *Tkinter*-based simulation supported by soil moisture sensors, position sensors, and distance sensors. The agent works by generating possible movement paths toward plants requiring water and testing each state to determine whether the goal has been achieved.

The results show that the *Generate and Test* method can assist the robot in determining action sequences systematically until all plants reach sufficient moisture levels. The system successfully navigates from the starting position to the target plants, performs watering actions, returns to the base position, and achieves the goal state optimally.

Keywords: artificial intelligence, generate and test, smart farming, state space, watering robot

I. PENDAHULUAN

Perkembangan teknologi kecerdasan buatan telah membawa perubahan besar dalam berbagai bidang, termasuk bidang pertanian. Konsep *smart farming* mulai banyak diterapkan untuk meningkatkan efisiensi pemantauan dan perawatan tanaman secara otomatis [1]. Sistem pertanian modern tidak hanya berfokus pada peningkatan hasil panen, tetapi juga bagaimana proses pemantauan dan penyiraman dapat dilakukan secara lebih cepat, tepat, dan efisien.

Salah satu masalah utama pada proses pertanian adalah pemantauan kelembapan tanah yang masih dilakukan secara manual. Petani harus mengecek kondisi tanaman satu per satu untuk mengetahui apakah tanaman membutuhkan air atau tidak. Proses tersebut memerlukan waktu yang lama dan kurang efisien apabila jumlah tanaman sangat banyak. Pemanfaatan sensor dan teknologi Internet of Things (*IoT*) menjadi solusi untuk meningkatkan efisiensi pada bidang pertanian modern [2].

Untuk mengatasi masalah tersebut, penelitian ini mengembangkan *Smart Farming Agent* berbasis kecerdasan buatan yang mampu melakukan penyiraman tanaman secara otomatis. Sistem menggunakan metode *Generate and Test* sebagai strategi pencarian solusi [3]. Metode ini bekerja dengan menghasilkan kemungkinan tindakan atau jalur pergerakan robot, kemudian menguji apakah *state* yang dihasilkan telah memenuhi kondisi tujuan.

Metode *Generate and Test* dipilih karena memiliki konsep sederhana namun efektif dalam menyelesaikan permasalahan berbasis *state space*. Pada sistem ini, robot akan menghasilkan kemungkinan perpindahan menuju tanaman yang membutuhkan air, kemudian menguji apakah seluruh tanaman sudah

memiliki kelembapan yang cukup. Jika belum tercapai, maka sistem akan menghasilkan *state* baru hingga *goal state* ditemukan. Penggunaan algoritma pencarian pada bidang kecerdasan buatan sangat penting untuk membantu sistem menentukan solusi terbaik berdasarkan kondisi lingkungan yang dinamis [3].

II. METODE PENELITIAN

Berikut adalah metode yang digunakan dalam penelitian ini yaitu:

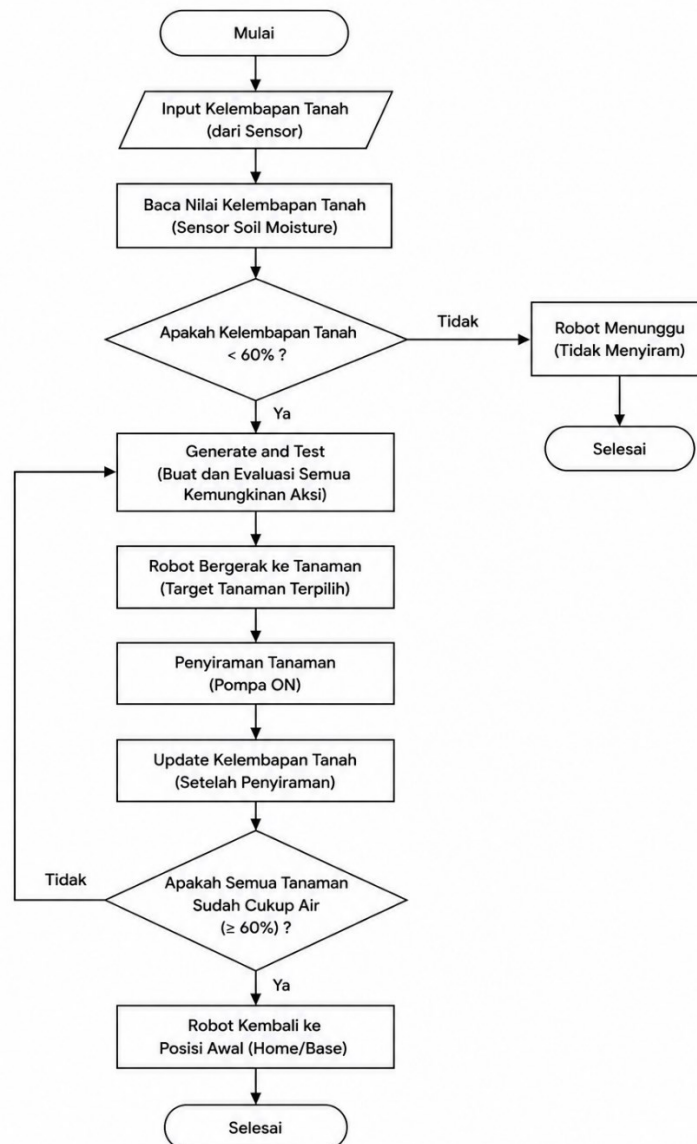
2.1 Desain Sistem

A. Model Page

- a. Percept: Agent menerima input berupa posisi robot, posisi tanaman, dan status kelembapan tanah.
- b. Action: Bergerak menuju tanaman, melakukan penyiraman, berhenti ketika semua tanaman sudah cukup air, dan kembali ke base.
- c. Goal: Seluruh tanaman memiliki moisture ≥ 60 dan robot kembali ke posisi awal.
- d. Enviroment: Lingkungan sistem berupa area pertanian simulasi yang berisi robot, beberapa tanaman, dan kondisi kelembapan tanah masing-masing tanaman.

2.2 Alur Kerja Sistem.

A. Flowchart



Gambar 1. (Gambar Menunjukkan Flowchart dari Smart Farming Agent)

B. Pseudocode

START

Input data kelembapan tanah

Baca nilai kelembapan setiap tanaman

WHILE masih ada tanaman dengan kelembapan < 60 DO

Generate kemungkinan aksi

Test setiap aksi yang memungkinkan

Pilih tanaman target

Robot bergerak menuju tanaman target

IF robot sudah sampai di target THEN

Aktifkan pompa air

Siram tanaman

Update nilai kelembapan tanah

ENDIF

ENDWHILE

Robot kembali ke posisi awal (home/base)

STOP

2.3 Formulasi Problem

A. State

State merupakan representasi kondisi sistem pada setiap proses yang dijalankan. Setiap state menggambarkan posisi robot, kondisi tanaman, dan nilai kelembapan tanah

B. Possible Action

Move (Robot bergerak menuju tanaman yang membutuhkan air), Water (Robot melakukan penyiraman tanaman), Check (Sistem mengecek nilai kelembapan tanah setelah penyiraman dilakukan), Return (Robot kembali ke posisi awal), Repeat (Sistem mengulang proses Generate and Test apabila masih terdapat tanaman dengan kelembapan di bawah batas minimum).

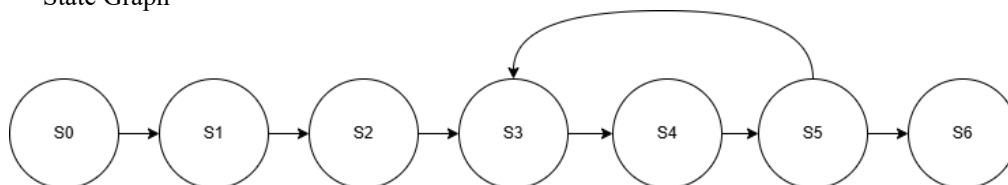
C. Goal Test

seluruh tanaman memiliki nilai kelembapan ≥ 60 , tidak ada tanaman yang membutuhkan air, robot kembali ke posisi awal.

D. Path Cost

Setiap perpindahan robot dihitung sebagai satu langkah, Semakin sedikit langkah yang digunakan maka jalur yang dipilih semakin optimal.

2.4 State Graph



Gambar 2. (Gambar Menunjukkan State Graph dari Smart Farming Agent)

Keterangan:

S0: Robot berada di posisi awal.

S1: Input kelembapan tanah.

S2: Membaca nilai kelembapan.

S3: Proses Generate and Test.

S4: Robot bergerak dan menyiram tanaman.

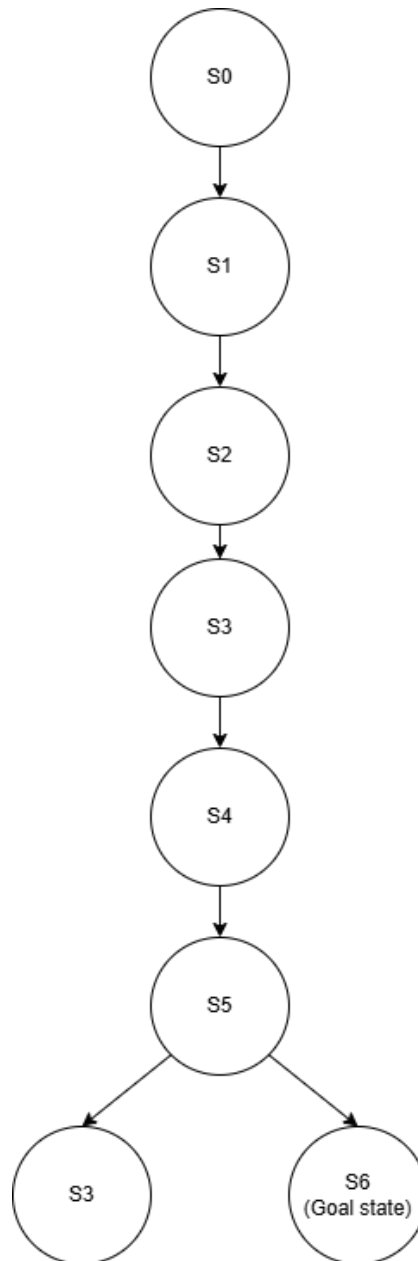
S5: Update kelembapan dan pengecekan goal.

Jika tanaman masih membutuhkan air, maka sistem kembali ke S3.

Jika semua tanaman sudah cukup air, maka lanjut ke S6.

S6: Robot kembali ke posisi awal dan sistem selesai.

2.5 Tree Terminology



Gambar 3. (Gambar Menunjukkan Tree Terminology dari Smart Farming Agent)

Penjelasan Setiap State

S0: kondisi awal robot berada di posisi awal (base).

S1: sistem menerima input kelembapan tanah.

S2: sistem membaca nilai kelembapan tanaman.

S3: proses Generate and Test dilakukan untuk menentukan tindakan robot.

S4: robot bergerak menuju tanaman dan melakukan penyiraman.

S5: sistem melakukan pengecekan apakah seluruh tanaman sudah memiliki kelembapan yang cukup.

S6: robot kembali ke posisi awal dan proses selesai (goal state).

2.6 Penerapan Generate and Test

Metode Generate and Test digunakan pada sistem Smart Farming Agent untuk menentukan proses penyiraman tanaman secara otomatis berdasarkan nilai kelembapan tanah. Metode ini bekerja dengan menghasilkan kemungkinan aksi (generate), kemudian menguji hasil aksi tersebut (test) sampai kondisi tujuan tercapai.

A. Input Sensor

Sistem menerima data kelembapan tanah dari setiap tanaman melalui sensor atau input manual. Data tersebut digunakan sebagai dasar untuk menentukan kondisi tanaman yang membutuhkan air.

Contoh:

- Tanaman A = 35
- Tanaman B = 80
- Tanaman C = 40

B. Membaca Kondisi Tanaman

Sistem membaca nilai kelembapan setiap tanaman dan membandingkannya dengan batas minimum kelembapan yang telah ditentukan.

a. Aturan:

- Jika nilai kelembapan kurang dari 60, maka tanaman membutuhkan air.
- Jika nilai kelembapan lebih dari atau sama dengan 60, maka tanaman dianggap cukup air.

b. Hasil:

- Tanaman A membutuhkan air.
- Tanaman C membutuhkan air.
- Tanaman B tidak membutuhkan air.

C. Generate

Pada tahap generate, sistem menghasilkan kemungkinan target tanaman yang perlu disiram berdasarkan hasil pembacaan sensor.

Kemungkinan aksi yang dihasilkan:

- Bergerak menuju Tanaman A.
- Bergerak menuju Tanaman C.

Menunggu apabila tidak ada tanaman yang membutuhkan air.

Tahap ini digunakan untuk menentukan kandidat solusi yang dapat dipilih oleh sistem.

D. Test

Setelah kemungkinan aksi dihasilkan, sistem melakukan pengujian (test) terhadap setiap kandidat.

Pengujian dilakukan dengan mengecek:

- apakah tanaman benar-benar membutuhkan air,
- apakah robot sudah mencapai target,
- apakah proses penyiraman berhasil dilakukan.

Jika target memenuhi kondisi penyiraman, maka sistem melanjutkan proses ke tahap berikutnya.

E. Pergerakan Robot

Robot bergerak menuju tanaman yang dipilih sebagai target penyiraman. Pergerakan dilakukan secara bertahap sampai robot mencapai posisi tanaman.

Contoh perpindahan:

- $(0,0) \rightarrow (1,0)$
- $(1,0) \rightarrow (2,0)$
- $(2,0) \rightarrow (2,1)$
- $(2,1) \rightarrow (2,2)$

F. Penyiraman Tanaman

Setelah robot mencapai target, sistem melakukan penyiraman tanaman sehingga nilai kelembapan meningkat.

Contoh:

- Moisture awal = 35
- Setelah disiram = 100

G. Test Goal State

Sistem melakukan pengecekan ulang untuk mengetahui apakah seluruh tanaman sudah memiliki kelembapan yang cukup.

- Jika masih terdapat tanaman dengan kelembapan kurang dari 60, maka sistem kembali ke tahap generate.
- Jika seluruh tanaman sudah memiliki kelembapan lebih dari atau sama dengan 60, maka kondisi goal state tercapai.

Tahap ini menunjukkan adanya proses pengulangan (looping) pada metode Generate and Test.

H. Return to Base

Setelah seluruh tanaman memiliki kelembapan yang cukup, robot kembali ke posisi awal (base). Kondisi ini menandakan bahwa proses penyiraman telah selesai dilakukan.

III. HASIL DAN PEMBAHASAN

3.1 Hasil Implementasi Sistem

Hasil implementasi penelitian ini berupa aplikasi Smart Farming Agent berbasis Python dan Tkinter yang mampu melakukan monitoring kelembapan tanah dan penyiraman tanaman secara otomatis menggunakan metode Generate and Test.

- Sistem terdiri dari beberapa komponen utama, yaitu:
- Tampilan sensor kelembapan tanaman.
- Input manual nilai kelembapan.
- Simulasi pergerakan robot.
- Proses pencarian target menggunakan metode Generate and Test.
- Penyiraman otomatis.
- Robot kembali ke posisi awal (base).

Pada simulasi, lingkungan pertanian direpresentasikan dalam bentuk grid berukuran 10×10. Robot ditampilkan dengan warna biru, sedangkan tanaman ditampilkan dengan warna:

- merah: membutuhkan air,
- hijau: kondisi kelembapan cukup.

3.2 Hasil Pengujian Sistem

Pengujian dilakukan menggunakan tiga tanaman dengan nilai kelembapan yang berbeda.

Data Awal Pengujian:

Tanaman	Posisi	Moisture Awal
Tanaman A	(2,2)	35
Tanaman B	(6,3)	80
Tanaman C	(4,7)	40

Tabel 1. (Data Awal Pengujian)

Threshold kelembapan minimum:

60

Berdasarkan data tersebut, sistem mendeteksi bahwa Tanaman A dan Tanaman C membutuhkan penyiraman.

3.3 Proses Generate and Test

Iterasi 1

Generate

Sistem menghasilkan kandidat tanaman yang membutuhkan air:

- Tanaman A
- Tanaman C

Test

Sistem menguji kandidat berdasarkan nilai kelembapan.

Hasil:

- Tanaman A = $35 < 60 \rightarrow$ valid
- Tanaman C = $40 < 60 \rightarrow$ valid

Sistem memilih Tanaman A sebagai target pertama.

Pergerakan Robot

Robot bergerak dari posisi:

- $(0,0) \rightarrow (1,0) \rightarrow (2,0) \rightarrow (2,1) \rightarrow (2,2)$

Penyiraman

Setelah robot mencapai target, sistem melakukan penyiraman.

Hasil:

- $35 \rightarrow 100$

Iterasi 2

Generate

Sistem kembali menghasilkan kandidat tanaman:

- Tanaman C

Test

Hasil pengujian:

- $40 < 60 \rightarrow$ valid

Pergerakan Robot

Robot bergerak menuju Tanaman C.

Penyiraman

Nilai kelembapan berubah:

- $40 \rightarrow 100$

3.4 Hasil Akhir Pengujian

Tanaman	Moisture Awal	Moisture Akhir	Status Akhir
Tanaman A	35	100	Cukup
Tanaman B	80	80	Cukup
Tanaman C	40	100	Cukup

Tabel 2. (Hasil Akhir Pengujian)

Berdasarkan hasil pengujian, seluruh tanaman berhasil mencapai nilai kelembapan di atas threshold minimum.

3.5 Pembahasan

Hasil implementasi menunjukkan bahwa metode Generate and Test dapat digunakan untuk menentukan proses penyiraman tanaman secara otomatis berdasarkan kondisi kelembapan tanah.

Pada tahap generate, sistem menghasilkan daftar tanaman yang membutuhkan air. Selanjutnya pada tahap test, sistem melakukan pengecekan terhadap kandidat tanaman sebelum robot bergerak menuju target.

Sistem bekerja secara berulang (looping) hingga seluruh tanaman memiliki kelembapan yang cukup. Setelah kondisi goal state tercapai, robot kembali ke posisi awal sebagai tanda bahwa proses penyiraman selesai dilakukan.

Berdasarkan pengujian yang dilakukan, metode Generate and Test mampu:

- menentukan target penyiraman secara otomatis,
- melakukan proses pencarian solusi secara bertahap,
- mengurangi penyiraman yang tidak diperlukan,
- membantu monitoring kondisi tanaman secara efisien.

Namun, metode ini masih memiliki keterbatasan karena pemilihan target dilakukan berdasarkan urutan hasil generate dan belum menggunakan optimasi jalur terpendek. Oleh karena itu, pada pengembangan selanjutnya sistem dapat dikombinasikan dengan algoritma pencarian lain seperti BFS, DFS, atau A* agar proses pergerakan robot menjadi lebih optimal.

IV. KESIMPULAN

Berdasarkan hasil penelitian dan implementasi sistem Smart Farming Agent, dapat disimpulkan bahwa metode Generate and Test berhasil diterapkan untuk proses penyiraman tanaman secara otomatis berdasarkan kondisi kelembapan tanah.

Sistem mampu:

- membaca nilai kelembapan tanaman melalui sensor atau input manual,
- menentukan tanaman yang membutuhkan air berdasarkan nilai threshold,
- menghasilkan kemungkinan aksi penyiraman menggunakan metode Generate and Test,
- menggerakkan robot menuju target tanaman,
- melakukan penyiraman secara otomatis,
- kembali ke posisi awal setelah seluruh tanaman memiliki kelembapan yang cukup.

Hasil pengujian menunjukkan bahwa sistem dapat mendeteksi tanaman dengan kelembapan di bawah batas minimum dan melakukan penyiraman hingga nilai kelembapan mencapai kondisi yang diinginkan. Proses looping pada metode Generate and Test juga berjalan dengan baik sampai goal state tercapai.

Dengan demikian, metode Generate and Test dapat digunakan sebagai solusi sederhana dalam penerapan kecerdasan buatan pada sistem smart farming, khususnya untuk monitoring dan penyiraman tanaman otomatis.

REFERENSI

1. Artikel Jurnal

Format:

- [1] S. Wolfert, L. Ge, C. Verdouw, and M. J. Bogaardt, "Big Data in Smart Farming – A review," *Agric. Syst.*, vol. 153, pp. 69–80, 2017, doi: 10.1016/j.agry.2017.01.023.
- [2] K. Kour *et al.*, "Monitoring Ambient Parameters in the IoT Precision Agriculture Scenario: An Approach to Sensor Selection and Hydroponic Saffron Cultivation," *Sensors*, vol. 22, no. 22, 2022, doi: 10.3390/s22228905.
- [3] P. Russel, S. J. & Norvig, "Artificial Intelligence: A Modern Approach," *An Autom. Irrig. Syst. Using Arduino Microcontroller*, vol. 1908, no. January, pp. 2–6, 2011.