

# **REKAYASA SISTEM CERDAS**



Tugas algoritma pencarian dan perencanaan otomatis pertemuan ke-3

Di Susun Oleh

Hendra

NPM. 2024130007

**PROGRAM STUDI SISTEM KOMPUTER**

**UNIVERSITAS HANDAYANI**

**MAKASSAR**

**TAHUN 2025**

# Laporan STRIPS-Based Planning

---

## 1. Pendahuluan

Perencanaan otomatis adalah salah satu bidang penting dalam kecerdasan buatan yang berfokus pada pencarian urutan aksi yang diperlukan untuk mencapai tujuan tertentu dari kondisi awal. Salah satu algoritma klasik dalam perencanaan adalah STRIPS (Stanford Research Institute Problem Solver) yang digunakan untuk merepresentasikan aksi dan kondisi dalam bentuk logika deklaratif.

## 2. Tujuan

Tujuan dari laporan ini adalah untuk menjelaskan algoritma STRIPS-Based Planning secara teoritis, menerapkannya pada studi kasus sederhana, menyelesaikannya menggunakan persamaan logika, dan mengimplementasikan solusi dalam bahasa pemrograman PHP.

## 3. Landasan Teori

STRIPS (Stanford Research Institute Problem Solver) adalah pendekatan perencanaan berbasis logika yang dikembangkan oleh Richard Fikes dan Nils Nilsson. STRIPS bekerja dengan merepresentasikan keadaan dunia dalam bentuk fakta, dan aksi dalam bentuk operator yang memiliki prekondisi dan efek (positif dan negatif).

## 4. Studi Kasus

Deskripsi: Robot berada di ruang A dan ingin mencapai ruang C. Koneksi: A terhubung ke B, B terhubung ke C.

Representasi STRIPS:

Kondisi Awal:  $At(Robot, A)$ ,  $Connected(A, B)$ ,  $Connected(B, C)$

Tujuan:  $At(Robot, C)$

Operator  $Move(X, Y)$ :

- Prekondisi:  $At(Robot, X)$ ,  $Connected(X, Y)$
- Efek Positif:  $At(Robot, Y)$
- Efek Negatif:  $At(Robot, X)$

## 5. Penyelesaian

Langkah-langkah:

1.  $\text{At}(\text{Robot}, A), \text{Connected}(A, B) \rightarrow \text{Move}(A, B) \rightarrow \text{At}(\text{Robot}, B)$
2.  $\text{At}(\text{Robot}, B), \text{Connected}(B, C) \rightarrow \text{Move}(B, C) \rightarrow \text{At}(\text{Robot}, C)$

Tujuan tercapai.

## 6. Implementasi dalam PHP

Kode Program:

```
<?php
class STRIPSPanner {
    private $state;
    private $connections;

    public function __construct($initialState, $connections) {
        $this->state = $initialState;
        $this->connections = $connections;
    }

    public function move($from, $to) {
        if ($this->state['location'] === $from && in_array($to, $this->connections[$from])) {
            echo "Moving from $from to $to\n";
            $this->state['location'] = $to;
        } else {
            echo "Cannot move from $from to $to\n";
        }
    }

    public function plan($goal) {
        $current = $this->state['location'];
        $path = $this->findPath($current, $goal);
        if ($path) {
            foreach ($path as $step) {
                $this->move($current, $step);
                $current = $step;
            }
        }
    }
}
```

```

        echo "Goal reached: Robot is at $goal\n";
    } else {
        echo "No path found to $goal\n";
    }
}

private function findPath($start, $goal, $visited = []) {
    if ($start === $goal) {
        return [];
    }
    $visited[] = $start;
    foreach ($this->connections[$start] as $neighbor) {
        if (!in_array($neighbor, $visited)) {
            $path = $this->findPath($neighbor, $goal, $visited);
            if ($path !== null) {
                array_unshift($path, $neighbor);
                return $path;
            }
        }
    }
    return null;
}

}

$initialState = ['location' => 'A'];
$connections = [
    'A' => ['B'],
    'B' => ['C'],
    'C' => []
];

$planner = new STRIPSPanner($initialState, $connections);
$planner->plan('C');
?>

```

## 7. Kesimpulan

STRIPS-Based Planning adalah algoritma perencanaan logis yang sangat berguna dalam domain kecerdasan buatan. Melalui studi kasus sederhana dan implementasi dalam bahasa

PHP, kita dapat memahami cara kerja dan struktur dari algoritma ini dalam menyelesaikan masalah perencanaan.