PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

INTELIGENCIA ARTIFICIAL (INF371)

AGENTES DE RESOLUCIÓN DE PROBLEMAS Y BÚSQUEDA

Dr. Edwin Villanueva Talavera

Contenido

- Agentes de Resolución de Problemas
- Búsqueda de Soluciones

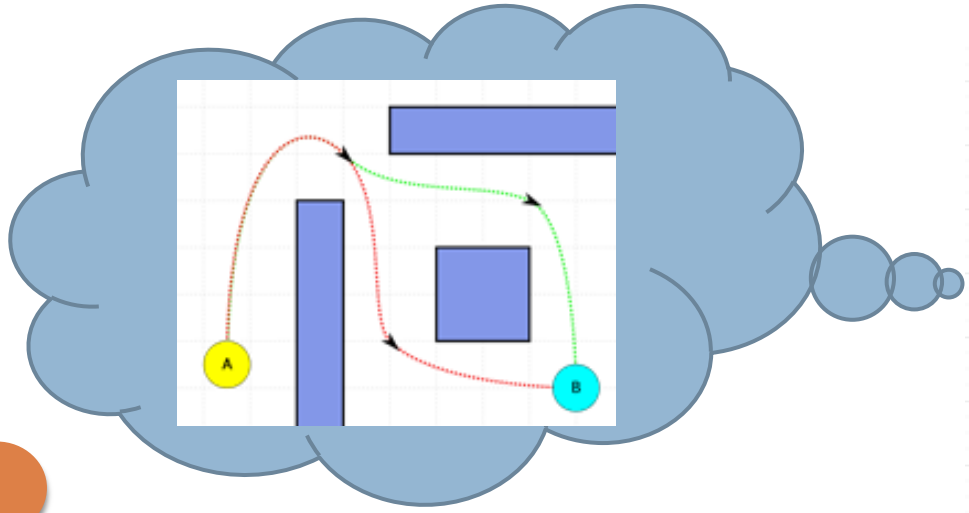
Bibliografía:

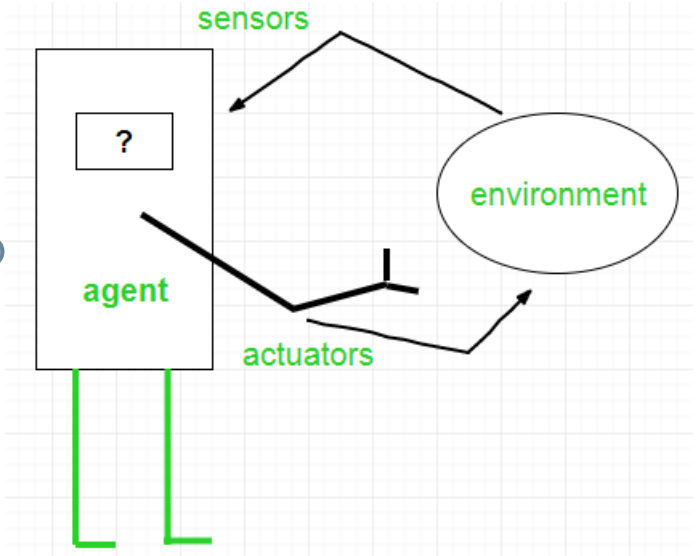
Capitulo 3.1, 3.2, 3.3 del libro:

Stuart Russell & Peter Norvig "Artificial Intelligence: A modern Approach", Prentice Hall, Third Edition, 2010

Agentes de resolución de problemas

- Los agentes reactivos no funcionan en entornos para los cuales el numero de reglas estado-acción es muy grande para almacenar
- En ese caso podemos construir un tipo de agente basado en objetivo llamado de agente de resolución de problemas
- Este tipo de agente usa representación atómica de los estados





Agentes de resolución de problemas

Pasos de un agente básico de resolución de problemas

- Formulación de objetivo
- Formulación de problema:
  - Estado inicial, espacio de estados, acciones, modelo de transición, costo de camino
- Búsqueda de solución:
  - encuentra una secuencia de acciones para llegar a un estado objetivo
- Ejecución de solución

Agentes de resolución de problemas

function SIMPLE-PROBLEM-SOLVING-AGENT(*percept*) **returns** an action
  persistent: *seq*, an action sequence, initially empty
    state, some description of the current world state
    goal, a goal, initially null
    problem, a problem formulation

  state ← UPDATE-STATE(*state*, *percept*)
  if *seq* is empty **then**
    goal ← FORMULATE-GOAL(*state*)
    problem ← FORMULATE-PROBLEM(*state*, *goal*)
    seq ← SEARCH(*problem*)
    if *seq* = *failure* **then return** a null action
  action ← FIRST(*seq*)
  seq ← REST(*seq*)
  return *action*

La suposición es un ambiente conocido, estático, observable, discreto e determinístico.

Agentes de resolución de problemas

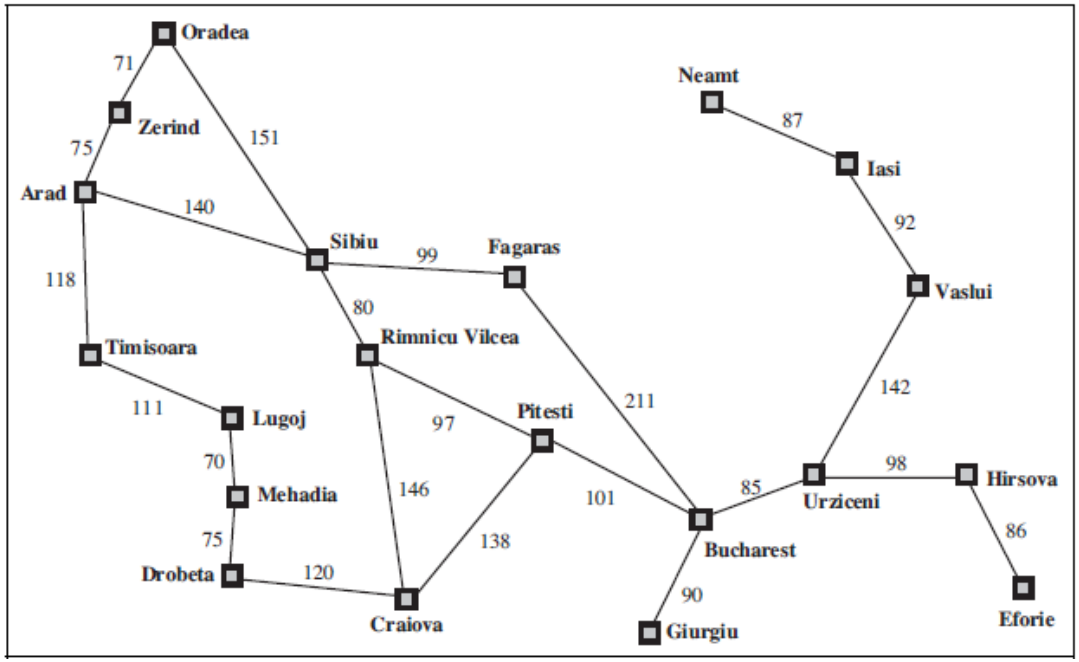
Componentes de la Formulación del Problema:

- Estados: Conjunto de situaciones diferentes que puede estar el problema
- Estado inicial: Situación inicial del problema
- Acciones: Operaciones que pueden ser realizadas desde un determinado estado *s*, denotada comúnmente como: ACTIONS(*s*)
- Modelo de transición: estados alcanzables desde un estado dado *s* con una determinada acción *a*, comúnmente se denota: RESULT(*s*,*a*)
- Función de prueba de objetivo: Determina si un estado es la solución del problema, comúnmente se denota: GOAL-TEST(*s*)
- Costo del camino: Alguna función que mide cuan difícil o costoso es determinado camino para llegar a un nodo *s* desde el estado inicial, *g*(*s*)

Agentes de resolución de problemas

Ejemplo de Formulación de Problema: búsqueda de ruta en mapa

- Estados: Todas las posibles ciudades
- Estado inicial: ciudad inicial
- Acciones: Moverse a alguna ciudad vecina
- Modelo de transición: Mapa



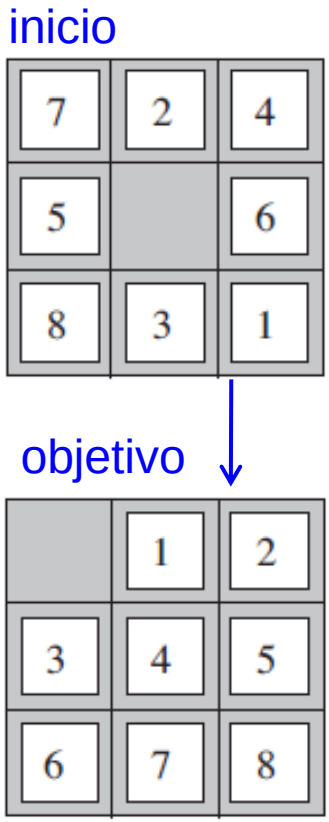
- Prueba de Objetivo: Verificar si se llego a la ciudad deseada
- Costo del camino: Puede ser tiempo, distancia recorrida, contaminación emitida, etc

Agentes de resolución de problemas

Ejemplo de Formulación de Problema: 8-puzzle

- Estados: Todas las configuraciones posibles de 8 números y un blanco
- Estado inicial: Alguna configuración dada del puzzle
- Acciones: Movimientos del casillero blanco: **Derecha, Izquierda, Arriba, Abajo**
- Modelo de transición: resultado de alguna acción, dado un estado:

  Ej. RESULT(inicio, Izquierda) = blanco y 5 intercambiados
- Prueba de Objetivo: Verifica si el estado es el objetivo
- Costo del camino: Cada acción cuesta 1. El costo de la solución seria el costo de todas las acciones



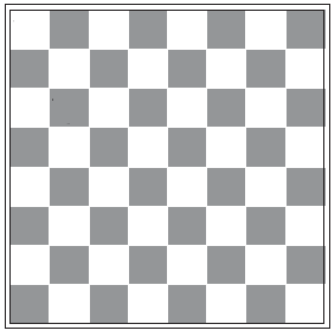
Agentes de resolución de problemas

Ejemplo de Formulación de Problema: 8-queens

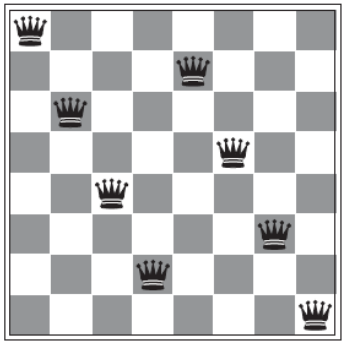
- Estados: configuraciones de 0 a 8 reinas en el tablero
- Estado inicial: 0 reinas en el tablero
- Acciones: **Adicionar** una reina a un casillero vacío
- Modelo de transición: Retorna el tablero con la reina añadida
- Prueba de objetivo: Verificar que el estado tenga 8 reinas no atacadas

Esta formulación tiene
64x63x…x57 ~**1.8x10^{14}** posibles secuencias a investigar!

inicio



Objetivo: estado donde las 8 reinas no se ataquen. Ej:



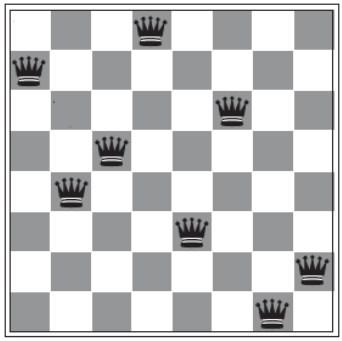
Agentes de resolución de problemas

Ejemplo de otra formulación de Problema: 8-queens

- Estados: Vectores de 8 números no repetidos (permutaciones). Cada elemento indica la fila en que se encuentra la reina en una columna
- Estado inicial: Permutación aleatoria
- Acciones: Intercambiar 2 elementos
- Prueba de objetivo: Verificar si la nueva permutación tiene reinas no atacadas

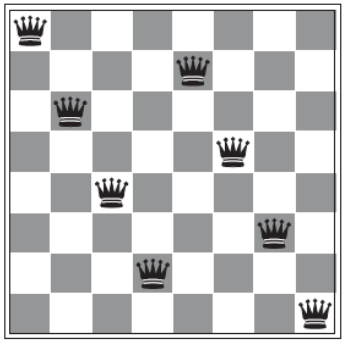
Esta formulación tiene
8x7x…x1 = **40320** posibles secuencias a investigar!

inicio



[2, 5, 4, 1, 6, 3, 8, 7]

Objetivo: estado donde las 8 reinas no se ataquen. Ej:



[1, 3, 5, 7, 2, 4, 6, 8]

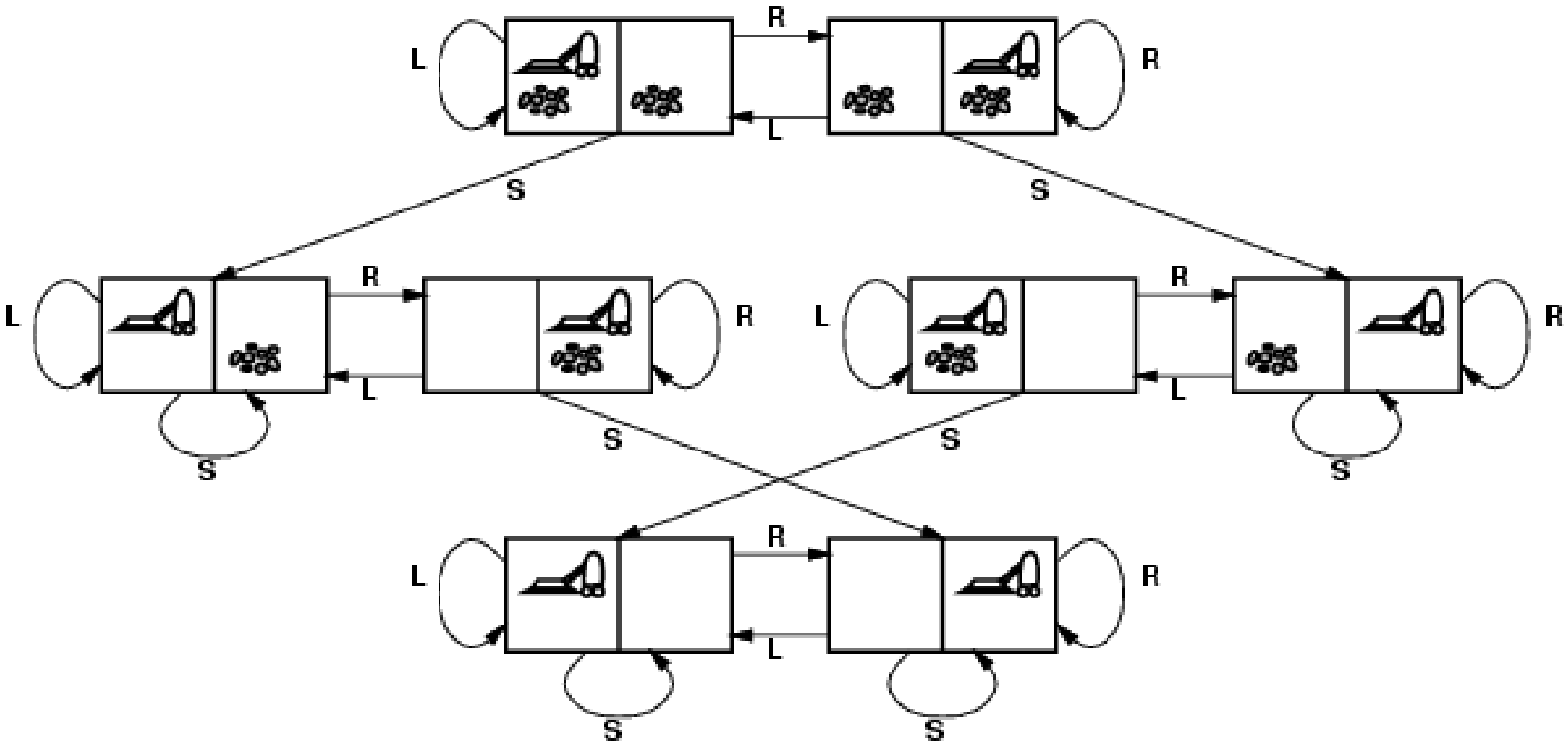
Agentes de resolución de problemas

Espacio de estados

- Conjunto de todos los estados accesibles a partir de un estado inicial
  - El estado inicial, las acciones y el modelo de transición (o **función sucesor**) determinan el espacio de estados
- El espacio de estados puede ser representado con un grafo, donde los nodos representan estados y los arcos acciones

Agentes de resolución de problemas

Ejemplo de espacio de estados del mundo de la aspiradora

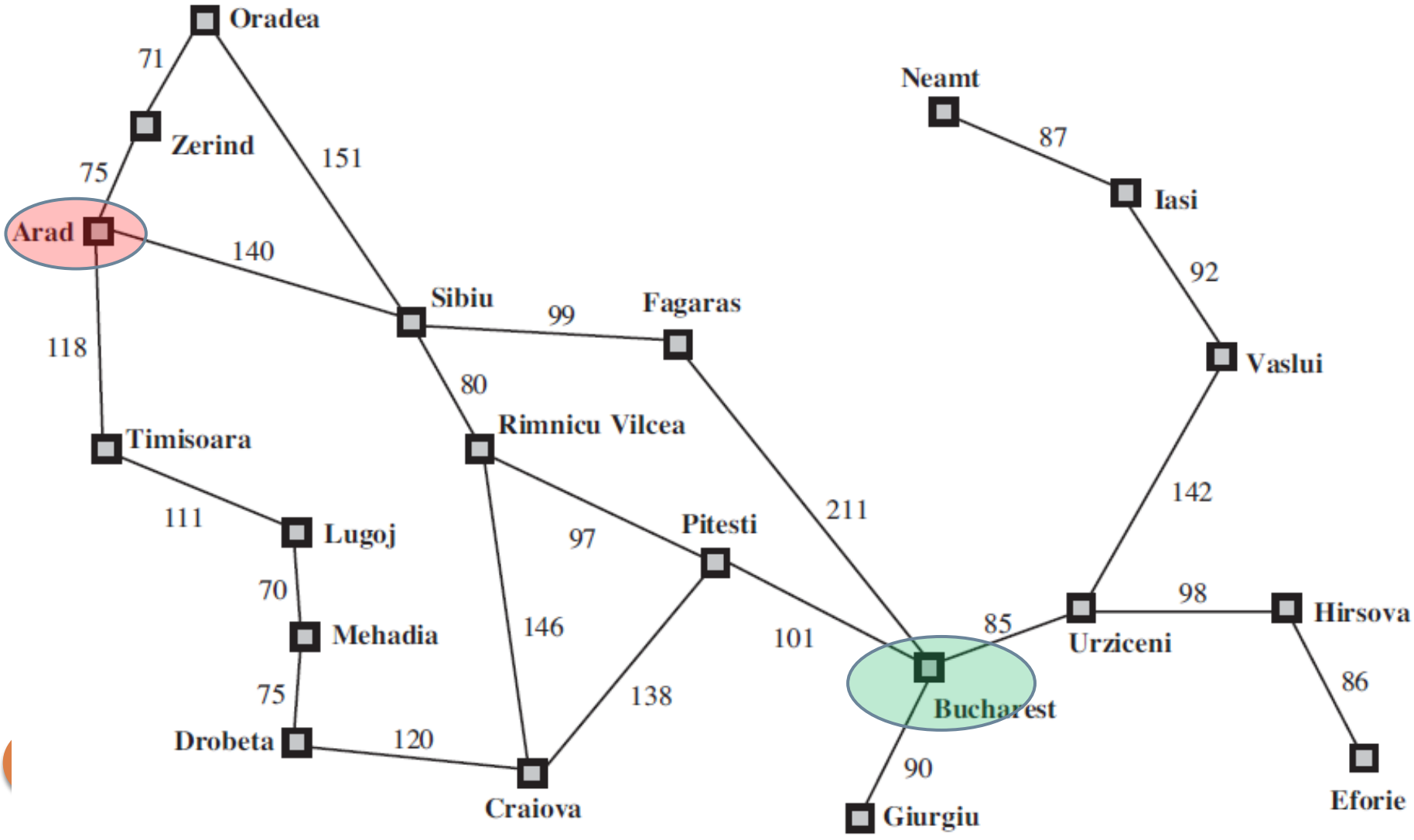


Búsqueda de Soluciones

- La idea es explorar el espacio de estados mediante el recorrido de un árbol de búsqueda
- Expandir el estado actual aplicando la función sucesor, generando nuevos estados
- La estrategia de búsqueda determina el camino a seguir, esto es, que nodos se exploran primero y cuales se dejan para después.

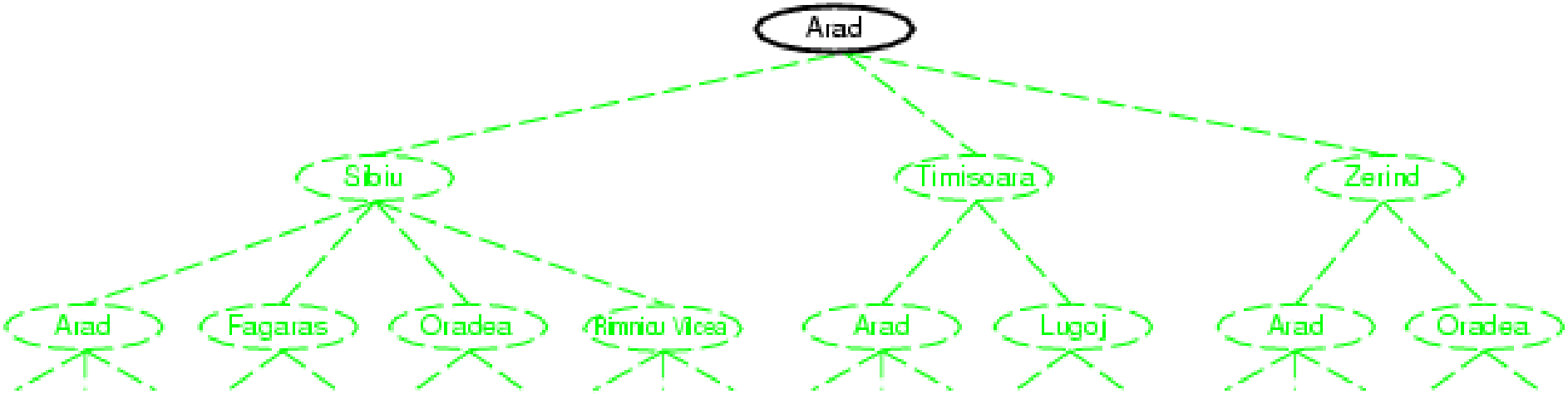
Búsqueda de Soluciones

Ejemplo: Búsqueda en el mapa de Romania



Búsqueda de Soluciones

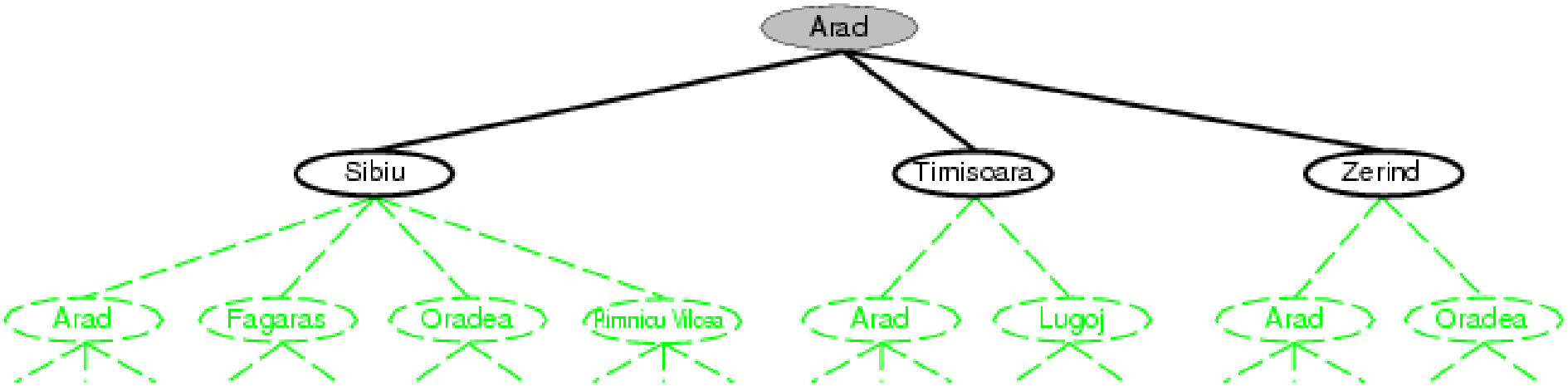
Ejemplo: Búsqueda en el mapa de Romania



Estado Inicial

Búsqueda de Soluciones

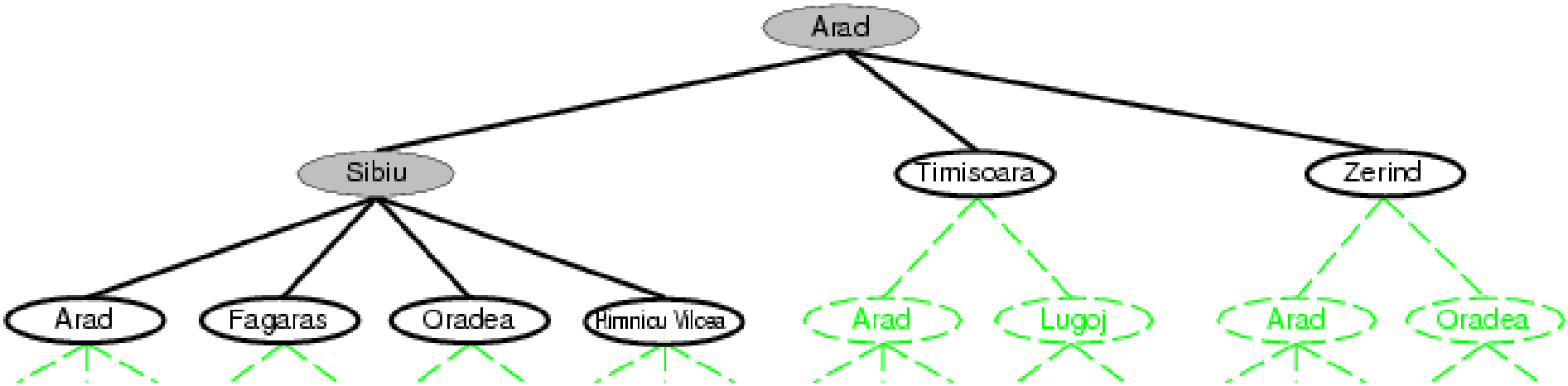
Ejemplo: Búsqueda en el mapa de Romania



Después de expandir Arad

Búsqueda de Soluciones

Ejemplo: Búsqueda en el mapa de Romania



Después de expandir Sibiu

Búsqueda de Soluciones

Árbol de búsqueda ≠ a espacio de estados!

- Un Nodo del árbol es una estructura de datos que implementa el árbol de búsqueda. Un estado es una configuración física.
  - Por ejemplo, el mapa de Romania tiene 20 estados, mientras que el árbol de búsqueda de Romania tiene tamaño infinito, ya que hay infinitos caminos del tipo:

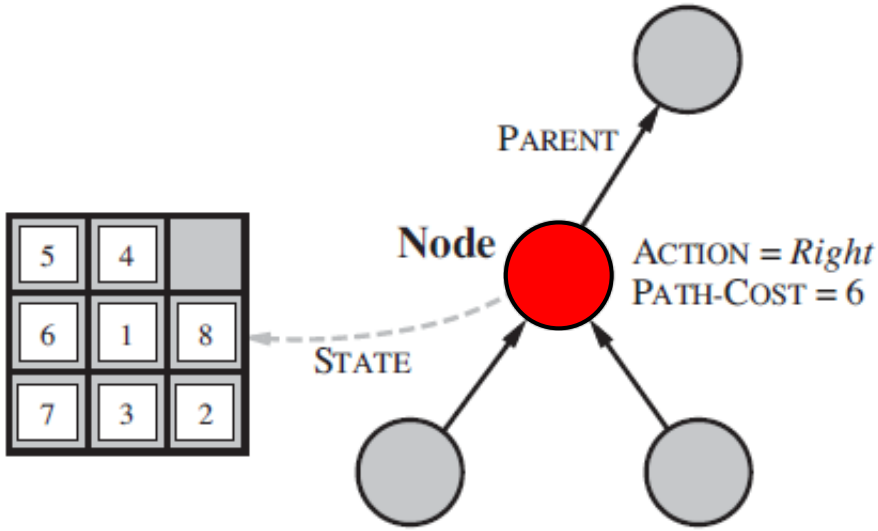
    Arad-Sibiu-Arad-Sibiu-Arad-...

- En el grafo de espacio de estados cada estado es representado por un único nodo.
- En un árbol de búsqueda varios nodos pueden representar un mismo estado (cuando hay varios caminos hacia ese estado).

Búsqueda de Soluciones

Estructura de un Nodo:

- Debe incluir información de: estado, nodo padre, la acción que generó el nodo, costo del camino desde el nodo raíz, y profundidad del nodo



- La colección de nodos que fueron generados pero aún no expandidos es llamada de **frontera**
- La forma como colocar/sacar nodos de la frontera define la **estrategia de búsqueda**

Búsqueda de Soluciones

Generación de nodos hijos:

```
function CHILD-NODE(problem, parent, action) returns a node
    return a node with
        STATE = problem.RESULT(parent.STATE, action),
        PARENT = parent, ACTION = action,
        PATH-COST = parent.PATH-COST + problem.STEP-COST(parent.STATE, action)
```

Búsqueda de Soluciones

Estructuras de datos para implementar la frontera: ***queue***

- First-in First Out (FIFO)
- Last-in First-out (LIFO o Pila)
- Cola de Prioridad

Operaciones en la frontera:

- *EMPTY?*(*queue*): Retorna *true* si la cola esta vacía
- *POP*(*queue*): Remueve y retorna el 1er elemento de la cola
- *INSERT*(*element*, *queue*): Inserta un elemento en la cola y devuelve esta

Preguntas?