

Introdução à Inteligência Computacional

Fundamentos, Aplicações e Perspectivas do Curso

Prof. Reginaldo Fernandes

Curso: Análise e Desenvolvimento de Sistemas
Disciplina: Inteligência Computacional

Aula #1 - October 23, 2025

Sumário e Agenda do Encontro

- 1 Resolução de Problemas
- 2 Agentes Baseados em Objetivos
- 3 Agentes Baseados em Objetivos
- 4 Busca e Resolução de Problemas
- 5 Algoritmos de Busca
- 6 Escopo da Resolução de Problemas
- 7 Agentes de Resolução de Problemas
- 8 Agentes de Resolução de Problemas
- 9 Agentes de Resolução de Problemas
- 10 A Simplificação pelo Objetivo
- 11 Foco e Formulação de Objetivos
- 12 Definição Formal de Objetivo
- 13 A Abstração na Formulação de Problemas
- 14 Formulação do Problema
- 15 Exemplo de Formulação de Problemas
- 16 Busca Cega em Ambientes Desconhecidos

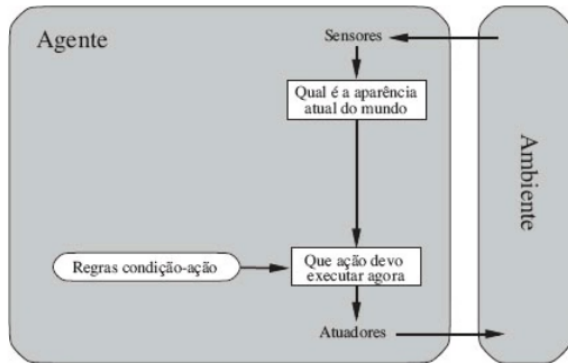
Resolução de Problemas

Os agentes mais simples que discutimos até agora são os agentes reflexivos

- Ações baseadas em um mapeamento direto de estados para ações.

Não conseguem operar bem em todos os ambientes

- 1 Este mapeamento pode ser muito grande para armazenar.
- 2 Este mapeamento pode levar muito tempo para aprender.



Agentes Baseados em Objetivos

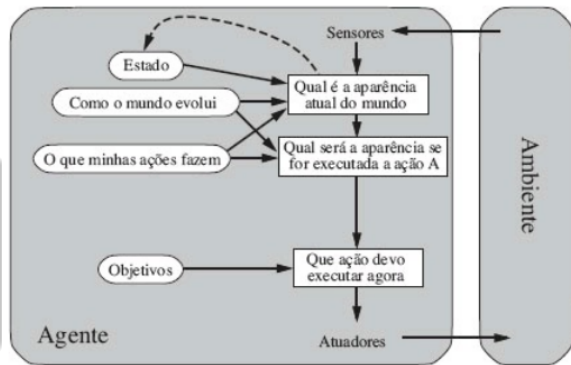
- Usam ações futuras e a desejabilidade dos resultados.
- Consideram o que o mundo **será** após uma sequência de ações.

O Papel do Objetivo

- O agente precisa de uma descrição do seu **Goal (Objetivo)**.
- Objetivo define situações desejáveis que o agente deve alcançar.

Estratégias Chave

- 1 **Busca:** Encontrar a sequência de ações (caminho) mais eficiente.
- 2 **Planejamento:** Criar um plano de execução de longo prazo.



Resolução de Problemas (Cont.)

Estudamos um tipo de agente baseado em objetivos: **Agente de Resolução de Problemas**

1. Agentes de Resolução de Problemas

- Usam **Representações Atômicas**.
- Estados são vistos como **totalidades** (wholes).
- **Não há estrutura interna** visível para os algoritmos de busca.
- **Exemplo:** Encontrar o caminho mais curto entre $A \rightarrow B$ em um mapa simples.

Distinção e Nível de Abstração Superior:

2. Agentes de Planejamento (Planning Agents)

- Usam **Representações Fatoradas ou Estruturadas**.
- O estado possui **estrutura interna** (objetos, propriedades, relações).
- A estrutura interna é usada para planejar ações complexas.
- **Exemplo:** Planejar ações baseado em "Posição do robô: (x, y)" e "Bateria: 80%".

Resolução de Problemas: Métodos de Busca

Transição: De Agentes para Algoritmos de Busca

Definições Fundamentais

Iniciaremos com as definições de:

- **Problemas:** Modelagem do ambiente e do objetivo (Estado Inicial, Ações, Custo).
- **Soluções:** O que caracteriza um caminho válido e ideal.

Algoritmos de Busca:

Algoritmos de Propósito Geral

Descreveremos métodos usados para resolver estes problemas:

- 1 **Busca Não Informada** (Cega): Explora o espaço sem conhecimento prévio (ex: BFS, DFS).
- 2 **Busca Informada** (Heurística): Usa estimativas para guiar a busca ao objetivo (ex: A*).

Resolução de Problemas: Busca Não Informada vs. Informada

1. Algoritmos de Busca Não Informada (Uninformed Search)

Busca Cega (Blind Search)

- **Não usam informação** sobre o problema além de sua definição (Estado Inicial, Ações, Teste de Objetivo).
- **Completude:** Podem resolver **qualquer problema solucionável**.
- **Eficiência:** Geralmente não são eficientes, pois exploram cegamente o espaço de estados.

2. Algoritmos de Busca Informada (Informed Search)

Busca Heurística

- Usam **informação extra** (heurísticas) para guiar a busca, indicando onde procurar as soluções.
- **Desempenho:** Podem ser muito mais rápidos e eficazes, dependendo da qualidade da heurística.

Resolução de Problemas: Definindo o Escopo

Escopo dos Problemas Abordados

Ambiente de Tarefas Simplificado

Nesta seção, consideraremos apenas o tipo mais simples de ambiente de tarefa:

- A solução é sempre uma **sequência de ações fixa** (sem contingências).

Limitação do Estudo:

Caso Geral (Contingente)

- O caso geral, onde as futuras ações dependem das futuras percepções (percepts), **não será discutido** neste módulo.

Avaliação dos Algoritmos

- Usaremos os conceitos de **Complexidade Assintótica**.
- Isso inclui a notação O (**Big-O notation**) e a introdução ao conceito de **NP-completeness** (complexidade intratável).

Agentes de Resolução de Problemas

Objetivo Primário da IA:

- Espera-se que os agentes maximizem sua **Medida de Desempenho**.
- O processo de atingir essa meta complexa pode ser **simplificado**.

Adoção de um Objetivo (Goal):

Para simplificar a maximização de desempenho, o agente precisa:

- 1 **Adotar um Objetivo:** Definir uma situação futura desejável.
- 2 **Visar Satisfazê-lo:** Focar em alcançar esse estado.

Foco da Análise:

- Analisaremos **por que** e **como** um agente decide adotar e perseguir um objetivo.

Exemplo: Por Que Precisamos de Objetivos?

Cenário:

- Imagine um agente turista na cidade de **Arad, Romênia**.

Medida de Desempenho (Performance Measure):

A medida de desempenho do agente engloba diversos fatores simultâneos:

- Aprimorar o bronzeadado.
- Melhorar o Romeno.
- Apreciar as vistas turísticas.
- Aproveitar a vida noturna.
- Evitar ressacas.
- ... (muitos outros).

O Problema de Decisão é Complexo:

- **Muitos Objetivos:** Há múltiplas metas que devem ser avaliadas.
- **Muitos Trade-offs:** É necessário equilibrar objetivos conflitantes (ex: vida noturna vs. evitar ressacas; bronzeadado vs. aprender).

Conclusão: O agente precisa de um **Objetivo Simplificado** para começar a agir.

Exemplo: Adoção do Objetivo Explícito

A Simplificação da Decisão

- Suponha que o agente tenha um **bilhete não reembolsável**.
- O voo de partida de **Bucareste** é no dia seguinte.

Adoção do Objetivo:

Neste cenário, faz sentido o agente adotar um objetivo explícito e primário:

- 1 **Meta: Chegar a Bucareste, até amanhã.**

Benefício Imediato:

- Qualquer curso de ação que **não chegue a Bucareste a tempo** pode ser **rejeitado** imediatamente.
- Essas ações não precisam de nenhuma consideração adicional.

Resultado: O problema de decisão do agente é **grandemente simplificado**.

Objetivos: Organização do Comportamento

A Função Principal dos Objetivos (Goals)

Os objetivos ajudam o agente a **organizar o comportamento** (ou seja, a tomar decisões de forma mais eficiente):

- **Limitação de Objetivos:** Os objetivos a serem alcançados pelo agente são **limitados** pelo **Goal** escolhido.
- **Limitação de Ações:** As ações que o agente precisa considerar também são **limitadas** (exclui ações que não levam ao objetivo).

Próxima Etapa: Formulação do Problema

- 1 A **Formulação do Objetivo (Goal Formulation)** é o primeiro passo na resolução de problemas.
- 2 A formulação do objetivo é baseada na **situação atual** do agente e na sua **Medida de Desempenho** (Performance Measure).

Foco: Uma vez que o objetivo é definido, precisamos saber como **alcançá-lo** – isto nos leva à **Formulação do Problema**.

Definição de Goal e Tarefa do Agente

Definição de Goal (Objetivo)

- O **Objetivo** será considerado um **conjunto de estados do mundo**.
- São todos os estados nos quais a condição do objetivo é **satisfeita**.
- **Exemplo:** O objetivo "Chegar a Bucareste" é o conjunto de todos os estados onde o agente está na localização "Bucareste".

A Tarefa do Agente

A principal tarefa do agente é descobrir **como agir** para que um estado objetivo seja alcançado.

- Isso envolve considerar **ações atuais e futuras**.

Próxima Etapa: Precisamos de uma forma de **modelar** esses estados e ações para que o agente possa planejar.

Definindo Ações e Estados: O Nível de Detalhe

Tarefa do Agente:

- O agente precisa decidir **quais tipos de ações e estados** deve considerar.

O Nível de Detalhe é Crítico:

- Considere ações excessivamente detalhadas, como:
 - 1 "Mover o pé esquerdo uma polegada para frente"
 - 2 "Girar o volante um grau para a esquerda"

Consequências do Excesso de Detalhe:

- O agente provavelmente **jamais encontraria a saída** do estacionamento (devido ao número de passos).
- Há **muita incerteza** no mundo (ruído) naquele nível de detalhe.
- Haveria **muitos passos** em uma única solução (ineficiência).

Conclusão: A formulação exige um nível de **abstração** adequado.

Definição: Formulação do Problema

Definição Formal

- A **Formulação do Problema** é o processo de **decidir quais ações e estados considerar**.

Propósito Central

- É o processo de **abstração** que transforma uma tarefa complexa de maximização de desempenho em uma **meta de busca**.
- A formulação é realizada com base em um **Goal (Objetivo)** já definido.

Foco na Abstração

- Ao escolher o nível de detalhe dos estados e ações, a formulação garante que o problema seja **solucionável** e **eficiente** para o agente.

Exemplo Prático: Viajando para Bucareste

Nível de Abstração Adotado:

- As **Ações** são definidas no nível de **dirigir de uma cidade principal para outra**.
- Cada **Estado** corresponde a **estar em uma determinada cidade**.

Ponto de Partida e Objetivo:

- **Estado Inicial:** O agente está em **Arad**.
- **Objetivo:** Chegar a **Bucareste**.
- O agente está considerando **para onde ir a partir de Arad**.

Ações Possíveis (A partir de Arad):

- Três estradas levam para fora de Arad, correspondendo às ações:
 - 1 Dirigir para **Sibiu**.
 - 2 Dirigir para **Timisoara**.
 - 3 Dirigir para **Zerind**.

Busca por Solução: Nenhuma destas ações imediatas **satisfaz o objetivo**. O agente precisa de uma **sequência de ações**.

Exemplo: Ausência de Informação Prévia

A Situação de Incerteza

- A menos que o agente "conheça a Romênia" (ou seja, tenha um modelo do mundo), ele não saberá **qual estrada seguir** a partir de Arad.

Consequências da Busca Cega

- O agente não sabe **qual de suas ações possíveis é a melhor** para chegar a Bucareste.
- O agente **não conhece o estado** que resultará de suas ações (além do nome da próxima cidade).

A Necessidade de Exploração

- Sem informação extra (heurística), o ambiente é considerado **desconhecido**.
- O agente **não tem escolha** a não ser tentar uma ação **aleatoriamente** (ou de forma sistemática cega, como BFS/DFS).

Transição: Precisamos de um modelo formal do problema para que o agente possa **planejar antes de agir**.

Exemplo: O Poder da Informação (O Mapa)

A Importância do Modelo (O Mapa)

- Suponha que o agente possua um **mapa da Romênia**.
- O mapa fornece ao agente a informação necessária para o planejamento:
 - ① Os **estados** em que ele pode se encontrar (cidades).
 - ② As **ações** que pode realizar (estradas entre cidades).

O Processo de Planejamento (Busca)

- O agente pode usar essa informação para considerar estágios subsequentes: simular uma jornada hipotética através de cada uma das três cidades iniciais.
- O objetivo é **encontrar um caminho no mapa** que, eventualmente, chegue a Bucareste.

Execução da Solução

- Assim que o agente encontra um **caminho (solução)** no mapa, ele deve **alcançar o objetivo** agindo.
- Isso é feito **executando as ações** na sequência planejada (dirigindo os trechos individuais da viagem).

Avaliação de Ações com Valor Desconhecido

O Dilema da Decisão Imediata

- Um agente com opções imediatas de **valor desconhecido** (como "ir para Sibiu ou Timisoara?") ainda pode decidir o que fazer.

Como a Decisão é Tomada:

O agente decide o que fazer ao:

- 1 **Examinar ações futuras** (Planejamento).
- 2 Encontrar uma sequência de ações que leve, eventualmente, a **estados de valor conhecido**.

O Valor Conhecido (Goal):

- O estado de **Bucareste** tem um valor conhecido (satisfaz o objetivo).
- A busca é o processo de **transferir o valor** desse estado final para a primeira ação imediata.

Suposições do Ambiente para a Busca

Para usar Algoritmos de Busca, assumimos que o ambiente é:

1. **Observável:** O agente **sempre sabe o estado atual**.
 - *Exemplo:* Cada cidade no mapa tem uma placa indicando sua presença.
2. **Discreto:** Em qualquer estado, há apenas um **número finito de ações** para escolher.
 - *Exemplo:* Cada cidade está ligada a um pequeno número de outras cidades.
3. **Conhecido:** O agente sabe **quais estados são alcançados** por cada ação.
 - *Exemplo:* Um mapa preciso atende a essa condição de navegação.
4. **Determinístico:** Cada ação tem **apenas um resultado**.
 - *Exemplo Ideal:* Dirigir de Arad para Sibiu leva idealmente apenas a Sibiu (sem falhas ou resultados múltiplos).

Consequência das Suposições do Modelo

A Solução em um Ambiente Idealizado

- Devido às suposições de **ambiente observável, conhecido e determinístico**, a solução do problema é sempre uma **sequência fixa de ações**.

Por que a Sequência Fixa Funciona?

- Se o mundo é conhecido e previsível (determinístico), o agente pode **simular** a sequência inteira do início ao fim **antes de agir**.

O Caso Geral (Mais Complexo):

- **Em geral**, a solução de um problema de IA seria uma **estratégia de ramificação** (*branching strategy*).
- Isso significa que as **ações futuras dependeriam das percepções futuras** do agente (e.g., "Se a porta estiver fechada, gire a chave; se aberta, siga em frente").

Exemplo: Condições Sub-Ideais e Planos de Contingência

Cenário Sub-Ideal (Falha de Execução)

- O agente planeja dirigir: **Arad** → **Sibiu** → **Rimnicu Vilcea**.
- Mas, por acidente, o agente chega a **Zerind** em vez de Sibiu.

A Necessidade de um Plano de Contingência

- Em condições sub-ideais, um **plano de contingência** é necessário para lidar com desvios ou falhas de execução.
- Assumimos **ambiente observável, conhecido e determinístico**.
- O agente **sabe exatamente** onde estará (e o que perceberá) após a primeira ação.

Consequência no Nosso Modelo:

- Apenas **um percepto** é possível após a primeira ação.
- Portanto, a solução pode especificar **apenas uma segunda ação possível**.

Foco: A busca é por uma **sequência de ações fixa** (e não uma estratégia de ramificação).

Definição: O Ciclo de Vida do Agente

Definição: Busca (Search)

- É o processo de **procurar uma sequência de ações** que alcança o objetivo (**goal**).
- O algoritmo de busca recebe um **Problema como entrada** e retorna uma **Solução**.
- A solução tem a forma de uma **sequência de ações**.

Definição: Execução (Execution)

- Uma vez que uma solução é encontrada, as ações recomendadas podem ser **executadas**.

Design Simples para o Agente

O processo de tomada de decisão é resumido no ciclo:

Formular \implies Buscar \implies Executar

Pseudocódigo: Variáveis e Lógica Base

Função Principal:

- `function SIMPLE-PROBLEM-SOLVING-AGENT(percept) returns an action`

Variáveis Persistentes (Memória do Agente):

O agente mantém as seguintes variáveis na memória:

- `seq`: A sequência de ações planejada, inicialmente vazia.
- `state`: Descrição do estado atual do mundo.
- `goal`: O objetivo atual, inicialmente nulo.
- `problem`: A formulação do problema.

Lógica de Recálculo:

- **Funcionamento:** O agente recalcula o plano (`seq`) somente quando a sequência anterior está **vazia** ou **falhou**.

Pseudocódigo: Agente de Resolução de Problemas Simples (Algoritmo)

```
função AGENTE-DE RESOLUÇÃO-DE-PROBLEMAS-SIMPLES(percepção) retorna uma ação  
  persistente: seq, uma sequência de ações, inicialmente vazia  
                estado, alguma descrição do estado atual do mundo  
                objetivo, um objetivo, inicialmente nulo  
                problema, uma formulação de problema  
  estado  $\leftarrow$  ATUALIZAR-ESTADO(estado, percepção)  
  se seq está vazia então faça  
    objetivo  $\leftarrow$  FORMULAR-OBJETIVO(estado)  
    problema  $\leftarrow$  FORMULAR-PROBLEMA(estado, objetivo)  
    seq  $\leftarrow$  BUSCA(problema)  
    se seq = falhar então retorne uma ação nula  
  ação  $\leftarrow$  PRIMEIRO(seq)  
  seq  $\leftarrow$  RESTO(seq)  
  retornar ação
```

Figure: AGENTE DE RESOLUÇÃO DE PROBLEMAS SIMPLES

Ignorando Percepções (Open-Loop)

Comportamento Durante a Execução

- Enquanto o agente está **executando a sequência de ações** (solução), ele **ignora suas percepções** (*percepts*).
- Para a escolha da ação, as percepções são **irrelevantes** (não são usadas na decisão).

Por Que Ignorar?

- O agente sabe **antecipadamente** (através da fase de Busca) qual será o resultado de cada ação.
- Isso só é possível porque assumimos um ambiente **observável, conhecido e determinístico**.

Terminologia (Controle)

- Teóricos de Controle chamam isso de **Sistema de Laço Aberto (Open-Loop System)**.
- Ignorar as percepções **quebra o laço de feedback** entre o agente e o ambiente.

Uso da Solução e Reinício do Ciclo

Fluxo de Processamento

Após formular um **Objetivo (Goal)** e um **Problema** a ser resolvido:

- 1 O agente chama um **procedimento de busca** para resolver o problema.
- 2 Ele utiliza a solução encontrada (**sequência de ações**) para guiar suas ações.

Execução da Sequência

- O agente executa o que a solução recomenda como **próxima coisa a fazer**.
- Tipicamente, esta é a **primeira ação da sequência**.
- Após a execução, essa ação é **removida** da sequência.

Recomeço

- Uma vez que toda a solução tenha sido executada (ou seja, a sequência está vazia), o agente **formulará um novo objetivo** (e um novo problema).

Definição Formal de Problemas para Agentes de Resolução

Definindo Formalmente um Problema de Busca

Um problema de busca pode ser definido formalmente por cinco componentes:

① **Estado Inicial (Initial State)**

- O estado a partir do qual o agente começa (e.g., estar na cidade de Arad).

② **Ações (Actions)**

- Uma função que retorna o conjunto de ações possíveis em um dado estado.

③ **Modelo de Transição (Transition Model)**

- Uma função que descreve o resultado de uma ação; retorna o próximo estado após a execução de uma ação.

④ **Teste de Objetivo (Goal Test)**

- Uma função que verifica se um dado estado satisfaz o objetivo do agente (e.g., verificar se a cidade é Bucareste).

⑤ **Custo do Caminho (Path Cost)**

- Uma função que atribui um custo numérico a um caminho percorrido (e.g., distância total percorrida ou tempo gasto).

1. Estado Inicial (*Initial State*)

Definição:

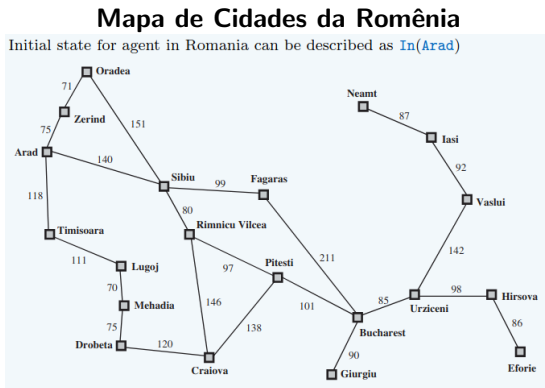
- É o estado a partir do qual o agente começa sua jornada de busca.
- Na modelagem atômica, é uma descrição completa, mas sem detalhes internos.

Exemplo:

- O Estado Inicial para o agente na Romênia pode ser descrito como:
- **In(Arad)**

Propriedade:

- É o único componente do problema que não requer uma função, pois é um valor fixo.



2. Modelo de Transição (*Transition Model*)

Definição:

- É a descrição formal do **que cada ação faz**.
- Essencialmente, define as "regras" do mundo (o mapa).

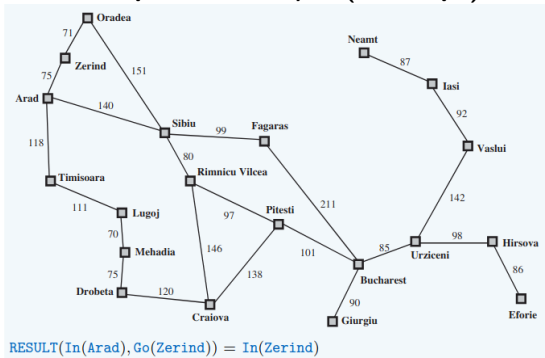
Função Formal:

- **RESULT**(s, a) retorna o estado resultante ao realizar a ação a no estado s .

Estado Sucessor (Successor State):

- Qualquer estado que possa ser alcançado a partir de um dado estado, através de uma **única ação**.

Exemplo de Transição (No Mapa)



(Se $s = \text{Arad}$ e $a = \text{Dirigir para Sibiu}$, $\text{RESULT}(s, a) = \text{Sibiu}$)

Definição: O Espaço de Estados (*State Space*)

Definição:

- Juntos, o **estado inicial**, as **ações** e o **modelo de transição** definem implicitamente o **Espaço de Estados** do problema.
- É o **conjunto de todos os estados** alcançáveis a partir do estado inicial por qualquer sequência de ações.

Representação (Grafo de Estados):

- O espaço de estados forma uma **rede ou grafo dirigido** (*directed graph*).
 - Os **nós** (*nodes*) são os estados.
 - Os **links** (*links*) ou arestas são as ações.

Exemplo do Mapa da Romênia:

- O mapa pode ser interpretado como um **grafo de espaço de estados**.
- Cada estrada entre duas cidades representa **duas ações de direção** (uma em cada direção).

Definição: Caminho (*Path*) no Espaço de Estados

Definição Formal de Caminho

- Um **Caminho** (*Path*) no espaço de estados é uma **sequência de estados**.
- Essa sequência de estados é conectada por uma **sequência correspondente de ações**.

Representação do Caminho

- O caminho começa no **Estado Inicial** (S_0).
- Ações ligam os estados: $S_0 \xrightarrow{A_1} S_1 \xrightarrow{A_2} S_2 \xrightarrow{A_3} \dots \xrightarrow{A_n} S_n$.

O Conceito de Solução

- Uma **Solução** para o problema é um caminho que leva o agente do Estado Inicial a um **Estado Objetivo**.
- A busca consiste em encontrar a sequência de ações $\{A_1, A_2, \dots, A_n\}$ que forma este caminho.

3. Teste de Objetivo (*Goal Test*)

Definição:

- O **Teste de Objetivo** é a função que **determina (verifica)** se um dado estado é um **estado objetivo**.
- O teste retorna TRUE se a condição for satisfeita.

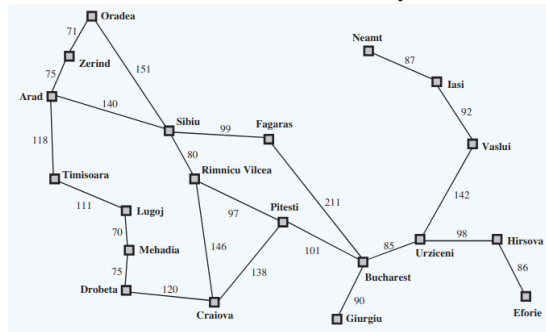
O Objetivo como Conjunto de Estados:

- O objetivo do agente na Romênia é o conjunto unitário:
- $\{In(Bucharest)\}$

Aplicação do Teste:

- O agente aplica o teste a cada estado que encontra durante a busca.

O Estado-Alvo no Mapa



(Bucareste é o alvo do teste de objetivo)

Objetivo Definido por Propriedade Abstrata

Objetivo Abstrato vs. Conjunto Explícito

- Às vezes, o objetivo é especificado por uma **propriedade abstrata**.
- Isso ocorre em vez de ser um conjunto de estados enumerado explicitamente.

Exemplo Clássico: O Jogo de Xadrez

- O objetivo é alcançar um estado chamado "**Xeque-Mate**" (*Checkmate*).

Definição da Propriedade (O Teste de Objetivo):

- O estado de Xeque-Mate é definido pela propriedade abstrata:
- "**O Rei do oponente está sob ataque e não pode escapar.**"

Implicação para a Busca:

- O **Teste de Objetivo** deve ser uma função complexa que verifica essa propriedade, e não apenas uma simples comparação de localização (como em *In(Bucareste)*).

4. Função de Custo do Caminho (*Path Cost*)

Definição:

- A **Função de Custo do Caminho** atribui um **custo numérico** a cada caminho no espaço de estados.

Relação com o Agente:

- Para agentes de resolução de problemas, a função de custo reflete o que o agente está tentando **minimizar** em sua **medida de desempenho**.

Cálculo do Custo:

- O custo total do caminho é a **soma dos custos das ações individuais** ao longo desse caminho.
- $\text{Custo}(S_0 \rightarrow S_n) = \sum_{i=0}^{n-1} \text{Custo}(\text{Ação}_i)$

Exemplo Prático (Romênia):

- Se o tempo é essencial para o agente (voo em Bucareste), o custo do caminho pode ser sua **distância em quilômetros** (que se correlaciona com o tempo de direção).

Custo de Ação Individual (*Step Cost*)

Definição:

- O **Custo do Passo** ($c(s, a, s')$) é o custo de realizar a ação a no estado s para alcançar o estado s' .

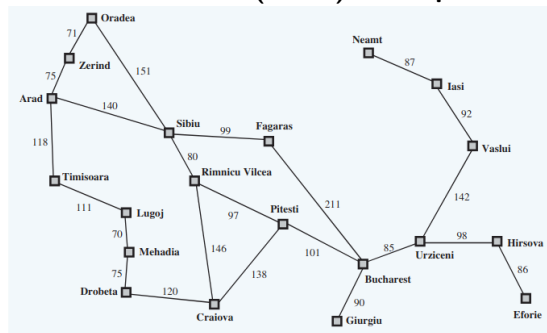
Regra Essencial:

- O custo do passo deve ser **não negativo** ($c \geq 0$).
- *Implicação:* Não é possível obter ganho de custo (benefício) ao executar uma ação; o custo deve ser sempre zero ou positivo.

Exemplo Prático:

- Os custos dos passos para a Romênia são definidos como as **distâncias da rota** em quilômetros.

Distâncias (Custo) no Mapa



custo $c(\text{Arad}, \text{Dirigir para Sibiu}, \text{Sibiu})$ é 140 km

(0

Definição de Problema e Qualidade da Solução

Definindo o Problema (Estrutura de Dados)

- Os cinco elementos definidos (**Estado Inicial**, **Ações**, **Modelo de Transição**, **Teste de Objetivo** e **Custo**) definem um problema.
- Todos podem ser reunidos em uma **única estrutura de dados**.
- Esta estrutura é passada como **entrada** para um algoritmo de resolução de problemas (*Search Algorithm*).

Definição de Solução

- Uma **Solução** para um problema é uma **sequência de ações**.
- A sequência leva o agente do Estado Inicial a um Estado Objetivo.

Medindo a Qualidade da Solução

- A qualidade da solução é medida pela **Função de Custo do Caminho** (*Path Cost Function*).
- Uma **Solução Ótima** é aquela que possui o **menor custo de caminho** entre todas as soluções possíveis.

Obrigado!

Prof. Reginaldo Fernandes
Inteligência Computacional (ADS)