

CSCM603130: Sistem Cerdas

Intelligent Agents

Fariz Darari, Aruni Yasmin Azizah

Fakultas Ilmu Komputer
Universitas Indonesia

2019/2020 • Semester Ganjil



Outline

- 1 Konsep rational agent
- 2 Task environment
- 3 Merancang sebuah agent
- 4 Jenis-jenis agent program

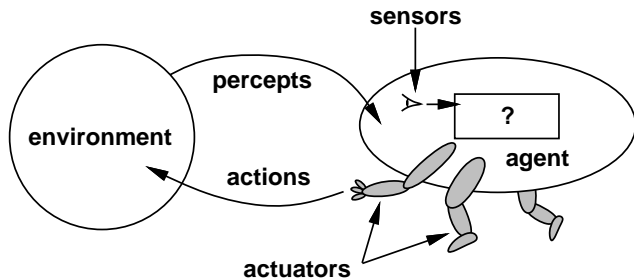


Outline

- 1 Konsep rational agent
- 2 Task environment
- 3 Merancang sebuah agent
- 4 Jenis-jenis agent program



Konsep *agent*



- *Percept (via Sensors)*: masukan “indra” si agent. Dkl: **input**
- *Percept sequence*: **sejarah** input si agent
- *Action (via Actuators)*: **tindakan** yang dilakukan oleh si agent
- *Environment*: **lingkungan** di mana si agent berada



Konsep *rational agent*

- *Rational*: melakukan hal yang **terbaik**.
- Konsekuensi dari apa yang dilakukan agent:
sequence of actions → sequence of *environment* states
- **Performance measure**: ukuran kinerja si agent melalui evaluasi
sequence of environment states.
- Aturan umum dalam merancang performance measure:
*what one actually wants in the environment, rather than
how one thinks the agent should behave.*



Rational agent

Definisi

Rational agent: suatu agent yang selalu bertindak **memaksimalkan performance measure**, berdasarkan apa yang ia amati tentang lingkungan (sejarah input) dan pengetahuan yang telah dimilikinya.

- Keywords: performance measure, pengetahuan yang telah dimiliki, tindakan yang bisa dilakukan, pengamatan tentang lingkungannya.
- **Rational tidak berarti sempurna**: ada aspek lingkungan yang tidak diketahui, di luar kendali.
- Terkadang agent bertindak (action) tanpa pengetahuan lingkungan → **exploration, learning, autonomy**



Outline

- 1 Konsep rational agent
- 2 Task environment**
- 3 Merancang sebuah agent
- 4 Jenis-jenis agent program



Task environment

- Ketika merancang sebuah agent, kita harus mendefinisikan lingkungan masalah (**task environment**), yakni:
 - **P**erformance measure: apa ukuran pencapaian tujuan si agent?
 - **E**nvironment: di manakah si agent berperan?
 - **A**ctuators: melalui apa agent melakukan tindakan?
 - **S**ensors: melalui apa agent menerima input?



Contoh: Agent taksi otomatis

Bayangkan sebuah agent taksi otomatis yang menerima penumpang dan mengantarkannya ke tujuan.

- Performance:

- Environment:

- Actuators:

- Sensors:



Contoh: Agent taksi otomatis

Bayangkan sebuah agent taksi otomatis yang menerima penumpang dan mengantarkannya ke tujuan.

- **P**erformance: sampai sesuai tujuan penumpang, mencapai setoran, hemat bensin, tidak nabrak, tidak ditilang
- **E**nvironment:
- **A**ctuators:
- **S**ensors:



Contoh: Agent taksi otomatis

Bayangkan sebuah agent taksi otomatis yang menerima penumpang dan mengantarkannya ke tujuan.

- **P**erformance: sampai sesuai tujuan penumpang, mencapai setoran, hemat bensin, tidak nabrak, tidak ditilang
- **E**nvironment: jalanan, lalu lintas, rambu lalu lintas, pejalan kaki, cuaca
- **A**ctuators:
- **S**ensors:



Contoh: Agent taksi otomatis

Bayangkan sebuah agent taksi otomatis yang menerima penumpang dan mengantarkannya ke tujuan.

- **P**erformance: sampai sesuai tujuan penumpang, mencapai setoran, hemat bensin, tidak nabrak, tidak ditilang
- **E**nvironment: jalanan, lalu lintas, rambu lalu lintas, pejalan kaki, cuaca
- **A**ctuators: stir arah, gas, rem, klakson, sinyal kiri/kanan
- **S**ensors:



Contoh: Agent taksi otomatis

Bayangkan sebuah agent taksi otomatis yang menerima penumpang dan mengantarkannya ke tujuan.

- **P**erformance: sampai sesuai tujuan penumpang, mencapai setoran, hemat bensin, tidak nabrak, tidak ditilang
- **E**nvironment: jalanan, lalu lintas, rambu lalu lintas, pejalan kaki, cuaca
- **A**ctuators: stir arah, gas, rem, klakson, sinyal kiri/kanan
- **S**ensors: kamera, speedometer, accelerometer, GPS, keyboard



Dimensi task environment

- **Fully-observable** vs. **partially-observable**: info relevan (terhadap performance measure) diketahui secara lengkap?
- **Single agent** vs. **multiagent**: apakah ada interaksi (kooperatif/kompetitif) dengan agen lainnya?
- **Deterministic** vs. **stochastic**: next state = current state + action?
- **Episodic** vs. **sequential**: apakah tindakan saat ini bergantung pada tindakan sebelumnya?
- **Static** vs. **dynamic**: environment berubah-ubah terlepas dari tindakan si agent?
- **Discrete** vs. **continuous**: apakah state, percept, dan action terhingga?
- Tentunya, tantangan terbesar: partially-observable, stochastic, sequential, dynamic, continuous, dan multiagent!



Outline

- 1 Konsep rational agent
- 2 Task environment
- 3 Merancang sebuah agent**
- 4 Jenis-jenis agent program



Struktur sebuah agent

Agent function

Sebuah fungsi yang memetakan sejarah input (**percept sequence**) terhadap tindakan yang dilakukan (**action**).

$$f : \mathcal{P}^* \rightarrow \mathcal{A}$$

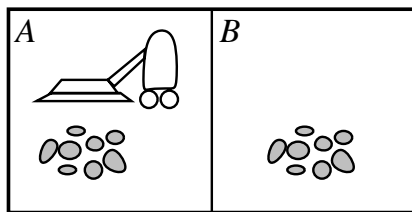
Agent program

Sebuah **program** yang mengimplementasikan fungsi f di atas suatu **arsitektur** (dengan sensors dan actuators).

- Agent = Arsitektur (Body) + Program (Brain)
- **Agent program** menerima input percept terkini (dengan kemungkinan penyimpanan percept sequence)



Contoh: VacuumCleanerWorld



Definisi **task environment**:

- Performance: ruangan A dan B selalu bersih
- Environment: ruangan A dan B beserta keberadaan debunya
- Actuators: untuk tindakan *DoKeKiri*, *DoKeKanan*, *DoSedot*
- Sensors: untuk percept *lokasi* dan *status kebersihan*

Contoh percept sequence:

$\langle [A, Kotor], [A, Bersih], [B, Kotor], [B, Bersih], \dots \rangle$

$\langle [A, Kotor], [A, Kotor], [A, Kotor], [A, Bersih], \dots \rangle$



Contoh agent: $\text{AGENRAJIN}^{\text{TM}}$

Agent function $\text{AGENRAJIN}^{\text{TM}}$

$$f(\langle \dots, [* , Kotor] \rangle) \rightarrow \text{DoSedot}$$
$$f(\langle \dots, [A, Bersih] \rangle) \rightarrow \text{DoKeKanan}$$
$$f(\langle \dots, [B, Bersih] \rangle) \rightarrow \text{DoKeKiri}$$

Agent program $\text{AGENRAJIN}^{\text{TM}}$

```
function AgenRajin ([lokasi,status]) returns action  
  if status = kotor then return DoSedot  
  else if lokasi = A then return DoKeKanan  
  else return DoKeKiri
```



Contoh agen lainnya: *AGENMALASTM*

Misalkan, tindakan si agent ditambahkan *DoSantai*.

Agent program *AGENMALASTM*

```
function AgenMalas ([lokasi,status]) returns action
  if status = kotor then return DoSedot
  else if random(1.0)  $\geq$  0.8 then return DoSantai
  else if lokasi = A then return DoKeKanan
  else return DoKeKiri
```

Mana yang lebih **rational**, *AGENRAJINTM* atau *AGENMALASTM*?



Contoh agen lainnya: *AGENMALAS*TM

Misalkan, tindakan si agent ditambahkan *DoSantai*.

Agent program *AGENMALAS*TM

```
function AgenMalas ([lokasi,status]) returns action
  if status = kotor then return DoSedot
  else if random(1.0) ≥ 0.8 then return DoSantai
  else if lokasi = A then return DoKeKanan
  else return DoKeKiri
```

Mana yang lebih **rational**, *AGENRAJIN*TM atau *AGENMALAS*TM?
Tergantung dari:

- Performance: Apa hemat energi bagian dari performance measure dalam mencapai goal “menjaga kebersihan”?
- Environment: Apakah ruangan yang sudah dibersihkan bisa kotor lagi? Seberapa cepat/sering?



Outline

- 1 Konsep rational agent
- 2 Task environment
- 3 Merancang sebuah agent
- 4 Jenis-jenis agent program**

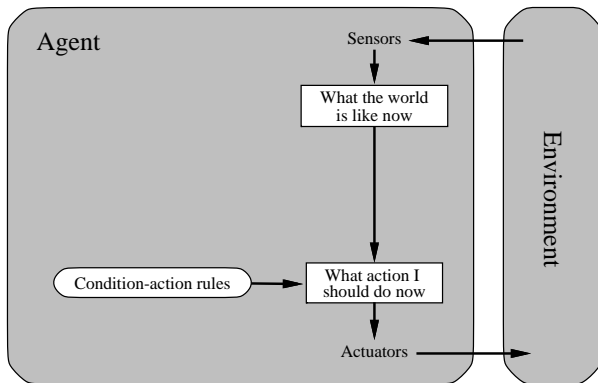


Jenis-jenis agent program

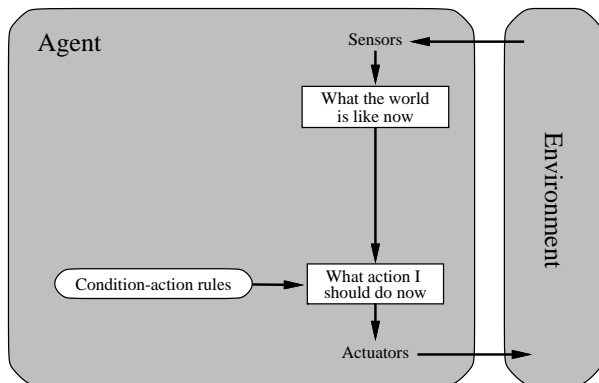
- **Simple reflex agents:** hanya berdasarkan percept terakhir.
- **Model-based reflex agents:** memiliki representasi internal mengenai keadaan lingkungan.
- **Goal-based agents:** memiliki informasi mengenai tujuan, memilih tindakan yang mencapai tujuan.
- **Utility-based agents:** melakukan penilaian kuantitatif terhadap suatu keadaan lingkungan → utility function. Berkaitan dengan performance measure.
- **Learning agents:** belajar dari pengalaman, meningkatkan kinerja.



Simple reflex agent



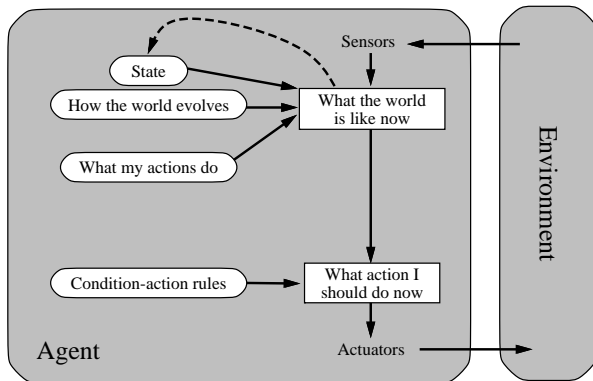
Simple reflex agent



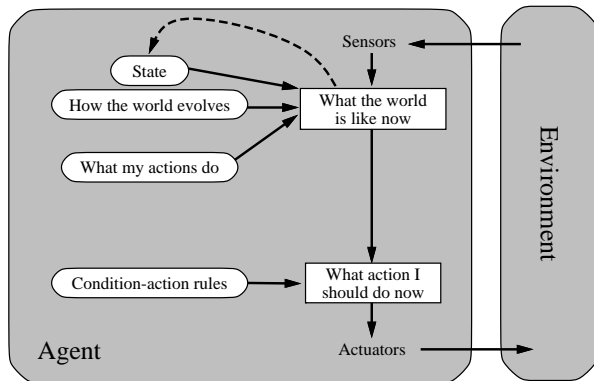
Bagaimana jika lingkungannya *partially observable*?



Model-based agent



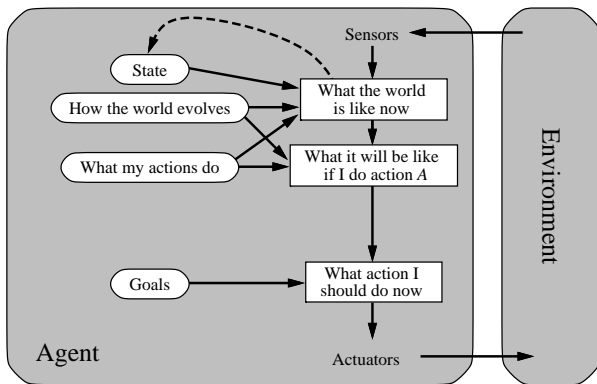
Model-based agent



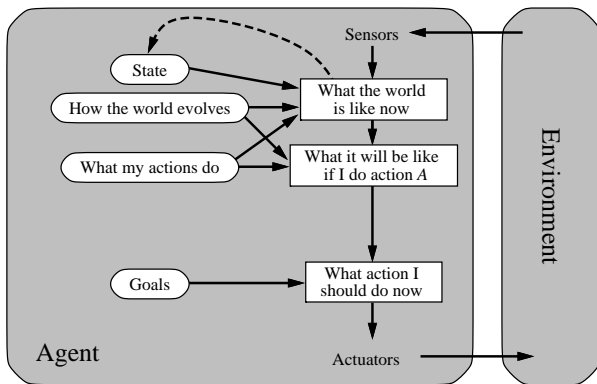
Bagaimana jika ada beberapa alternatif tindakan yang dapat dilakukan?



Goal-based agent



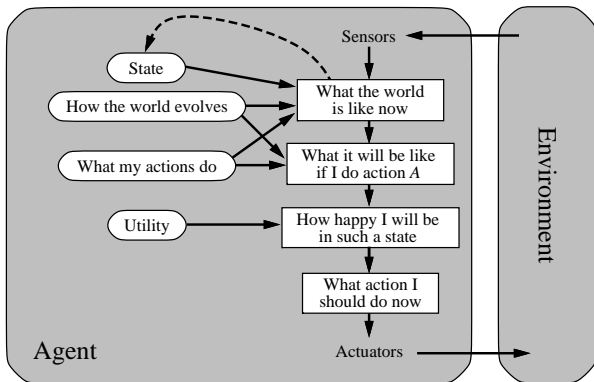
Goal-based agent



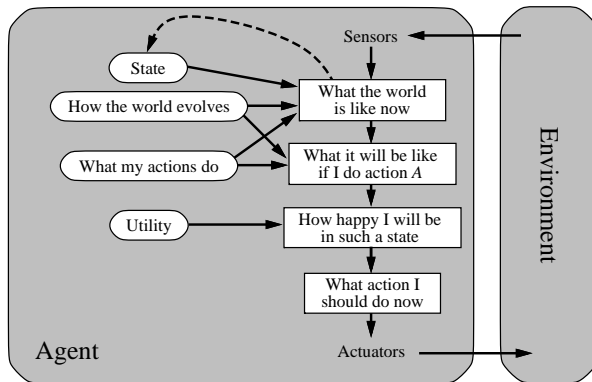
Bagaimana jika ada conflicting goals atau ketidakpastian dalam pemenuhan goal-goal yang ingin dicapai?



Utility-based agent



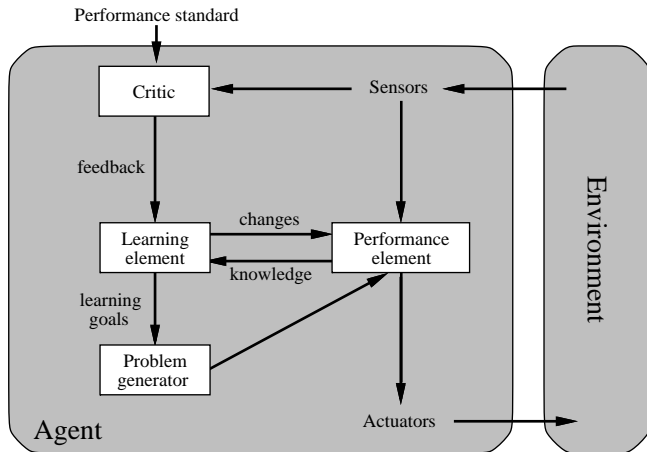
Utility-based agent



That's it?



Learning agent



Ringkasan

Konsep *rational agent*

- **Rational agent** memaksimalkan performance measure dengan bertindak berdasarkan apa yang ia amati tentang lingkungan dan pengetahuan yang telah dimiliki.
- **Task environment** mendefinisikan *performance measure*, *environment*, *actuator*, dan *sensor*.
- **Agent function** memetakan sejarah percept menjadi tindakan. **Agent program** mengimplementasikan agent function.

$$\text{agent} = \text{arsitektur} + \text{program}$$

- Ada beragam jenis agent: **simple reflex**, **model-based reflex**, **goal-based**, **utility-based**, dan **learning**.

