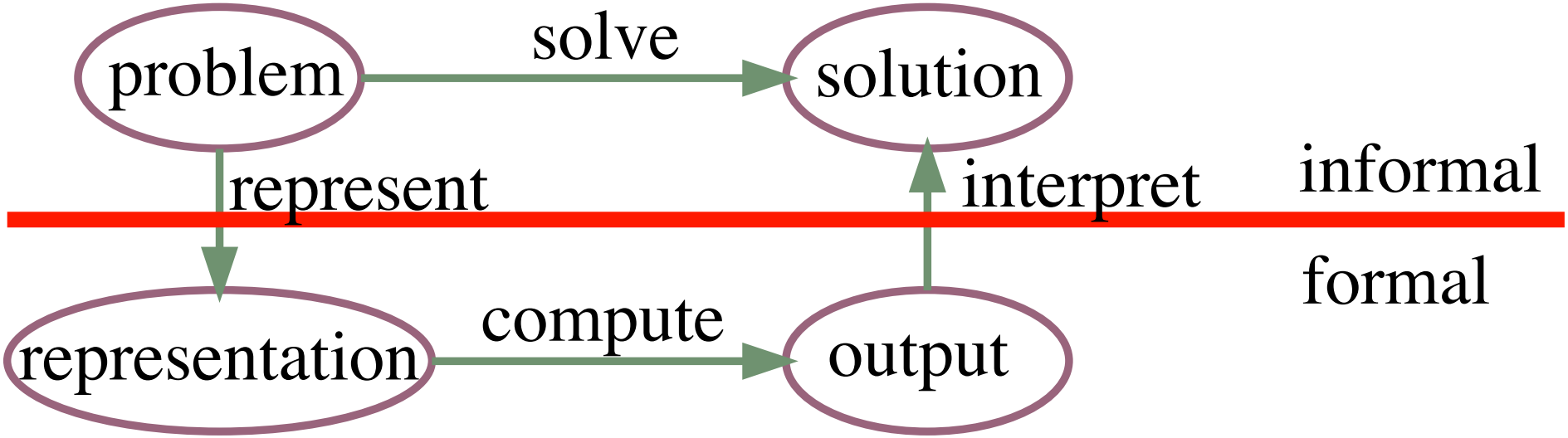
Representations



What do we want in a representation?

We want a representation to be

- rich enough to express the knowledge needed to solve the problem;
- as close to the problem as possible: compact, natural and maintainable;
- amenable to efficient computation
  - able to express features of the problem that can be exploited for computational gain
  - able to trade off accuracy and computation time and/or space
- able to be acquired from people, data and past experiences.

Defining a Solution

- Given an informal description of a problem, what is a solution?
- Typically much is left unspecified, but the unspecified parts can't be filled in arbitrarily.
- Much work in AI is motivated by common-sense reasoning. The computer needs to make common-sense conclusions about the unstated assumptions.

Quality of Solutions

- Does it matter if the answer is wrong or answers are missing?

Classes of solution:

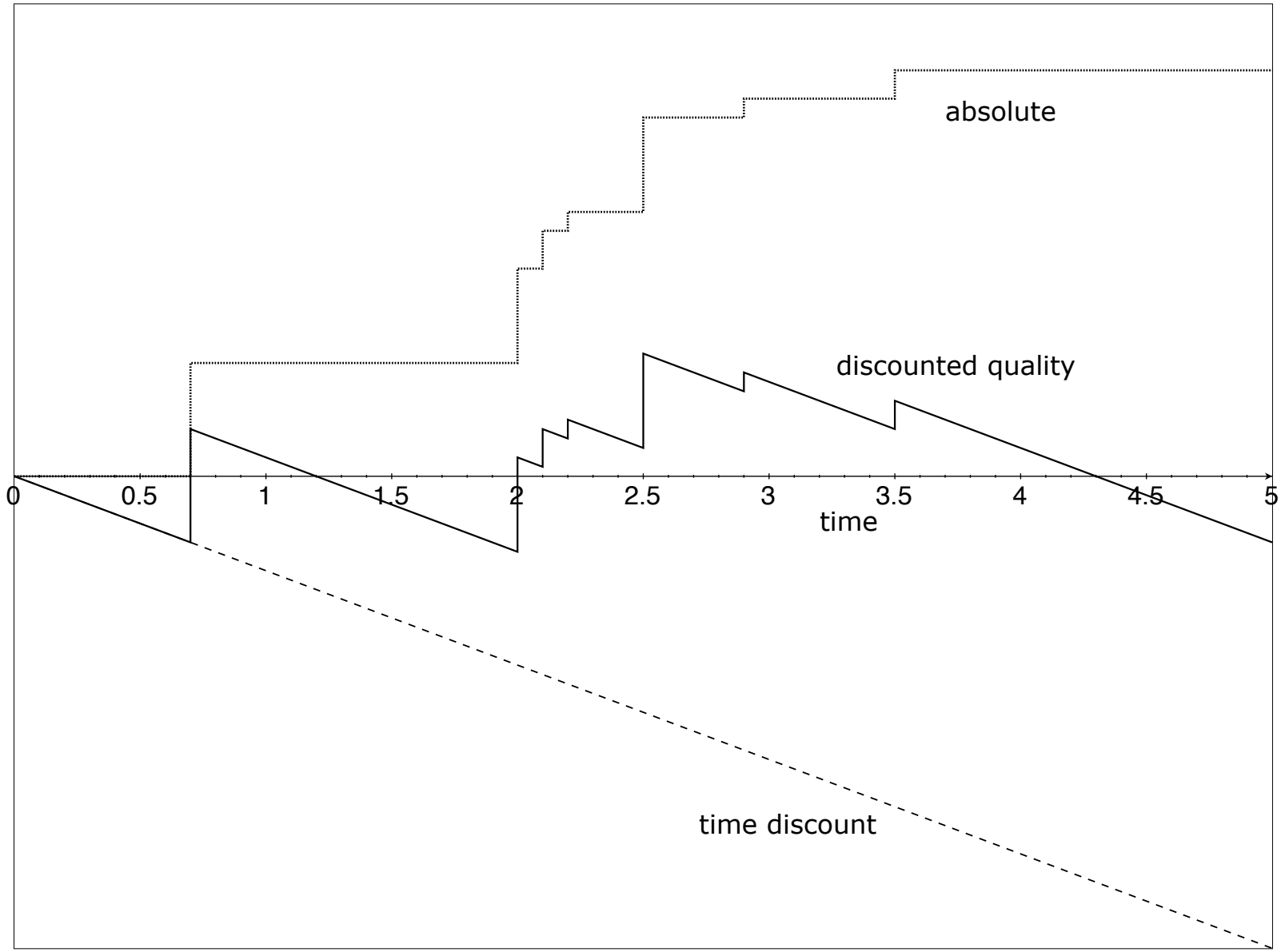
- An optimal solution is a best solution according some measure of solution quality.
- A satisficing solution is one that is good enough, according to some description of which solutions are adequate.
- An approximately optimal solution is one whose measure of quality is close to the best theoretically possible.
- A probable solution one that is likely to be a solution.

Decisions and Outcomes

- Good decisions can have bad outcomes. Bad decisions can have good outcomes.
- Information can be valuable because it leads to better decisions: value of information.
- We can often trade off computation time and solution quality. An anytime algorithm can provide a solution at any time; given more time it can produce better solutions.

An agent isn't just concerned about finding the right answer, but about acquiring the appropriate information, and computing it in a timely manner.

Solution quality and computation time



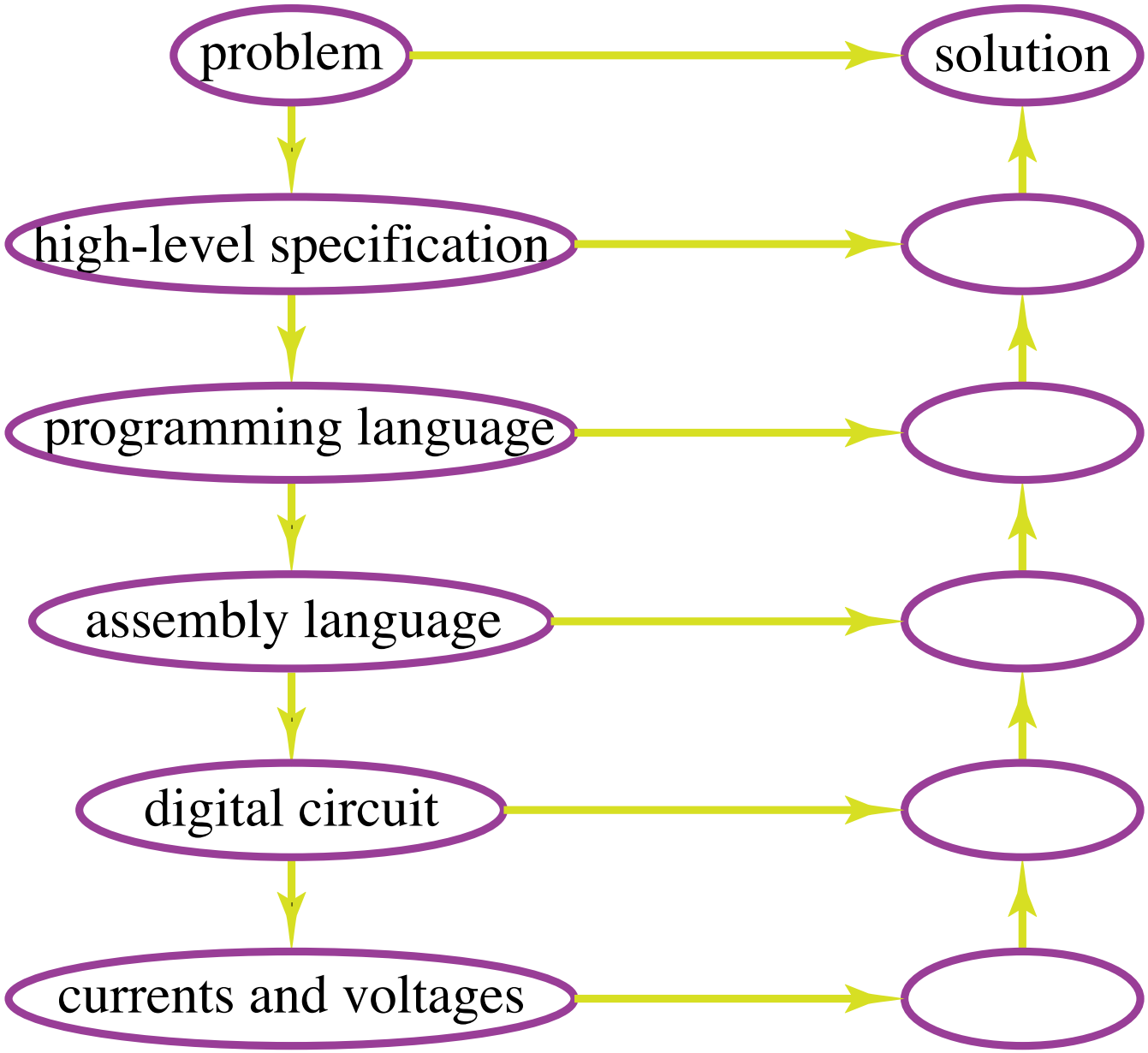
Choosing a Representation Language

We need to represent a problem to solve it on a computer.

$$\left[\begin{array}{l} \text{problem} \\ \quad \rightarrow \text{ specification of problem} \\ \quad\quad \rightarrow \text{ appropriate computation} \end{array}\right]$$

Example representation languages: Machine Language, C++, Java, Prolog, English

Hierarchy of representations



Physical symbol system hypothesis

- A symbol is a meaningful physical pattern that can be manipulated.
- A symbol system creates, copies, modifies and destroys symbols.

Physical symbol system hypothesis:

- A physical symbol system has the necessary and sufficient means for general intelligent action.

Knowledge & Symbol Levels

Two levels of abstraction seem to be common among biological and computational entities:

- The knowledge level is in terms of what an agent knows and what its goals are.
- The symbol level is a level of description of an agent in terms of what reasoning it is doing.

The knowledge level is about the external world to the agent.
The symbol level is about what symbols an agent uses to implement the knowledge level.

Mapping from Problem to Representation

- What level of abstraction of the problem to represent?
- What objects and relations in the world to represent?
- How can an agent represent the knowledge to ensure that the representation is natural, modular, and maintainable?
- How can an agent acquire the information from data, sensing, experience, or other agents?

Choosing a level of abstraction

- A high-level description is easier for a human to specify and understand.
- A low-level description can be more accurate and more predictive. High-level descriptions abstract away details that may be important for actually solving the problem.
- The lower the level, the more difficult it is to reason with.
- You may not know the information needed for a low-level description.

It is sometime possible to use multiple levels of abstraction.

Reasoning and acting

Reasoning is the computation required to determine what an agent should do.

- Design time reasoning and computation is carried out by the designer the agent.
- Offline computation is the computation done by the agent before it has to act.
  Background knowledge and data $\rightsquigarrow$ knowledge base.
- Online computation is the computation that's done by an agent between receiving information and acting.