Assignment 4 Review

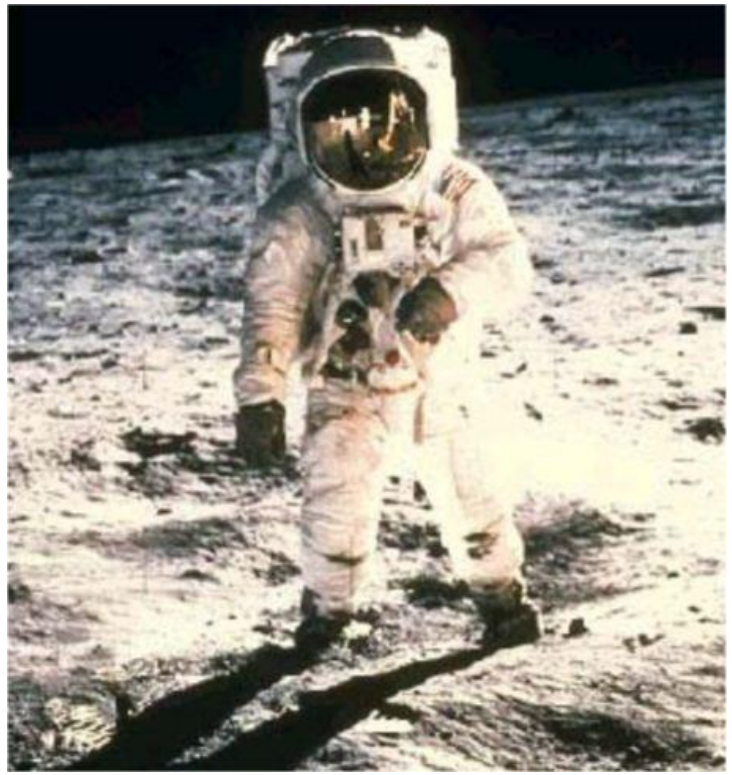
Walk through of my solution

Important points:

- Unification and DFS recursion may be Prolog's only tools, but they are *powerful* tools. Use them!
- Although Prolog is not *perfectly* declarative, it is designed to be used declaratively. So, think of problems in terms of "what is true about a solution?" rather than "how do I get to a solution?".
- Programming declaratively: *Write* it as a predicate; *use* it as a function or generator.
- I didn't show you the "if… then… else…" syntax in Prolog because it invites procedural thinking and bad (Prolog) code. If what you did works, OK, but you might not have learned what you need to learn to use Prolog effectively.

Assignment 5 Walkthrough

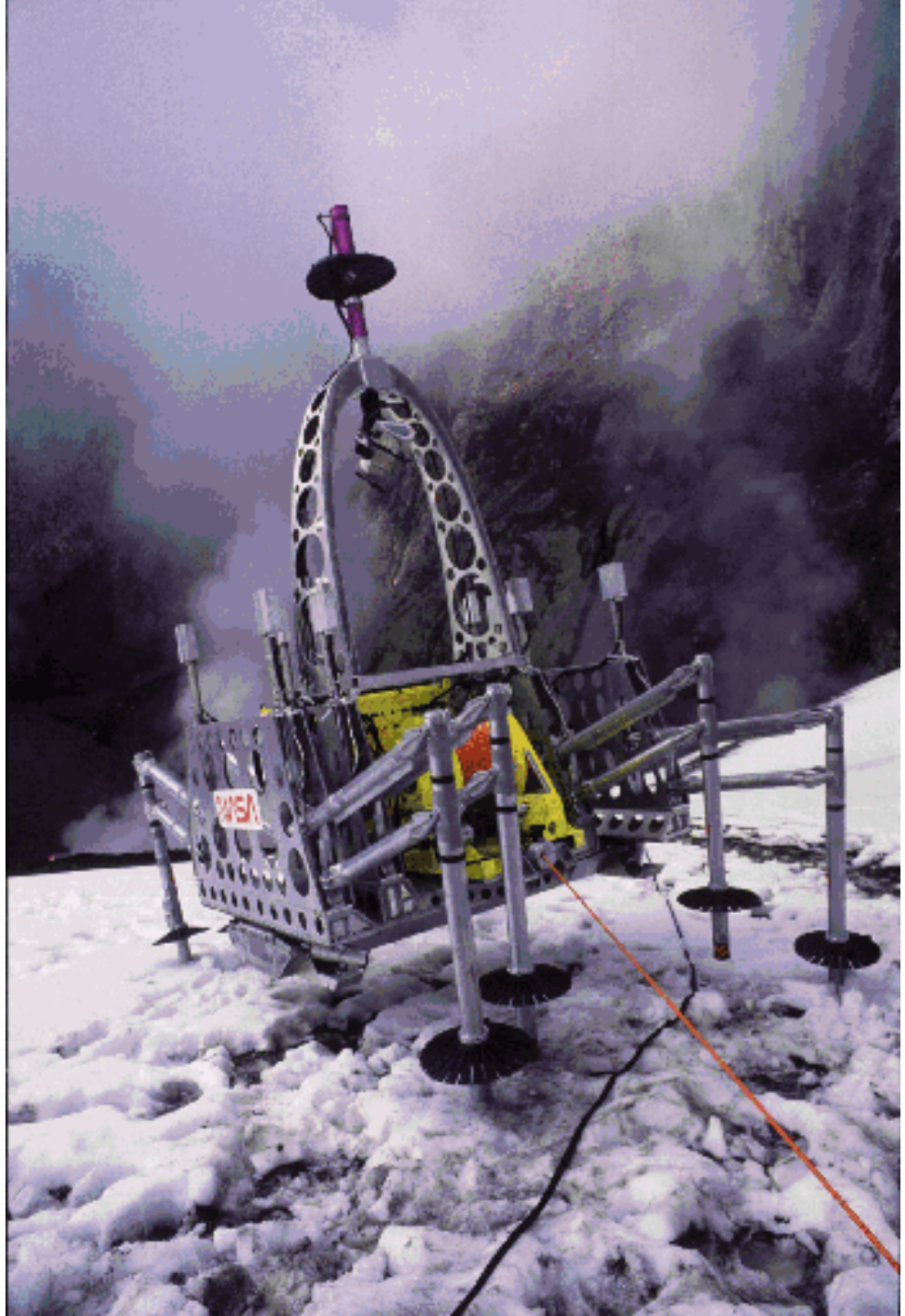
Why Robots?



versus

Robots are tough…

…and we don’t care much when they ‘die’.



Dante II:

Volcano Explorer

Robots are efficient…

…and can be cheap in the long term

Robots at an electronics factory.

Robots don’t get bored…

…or tired, or grumpy, or sick, or…

A mushroom picking robot.

The Bear robot can carry a human in body armor.

Robots can be customized for their tasks

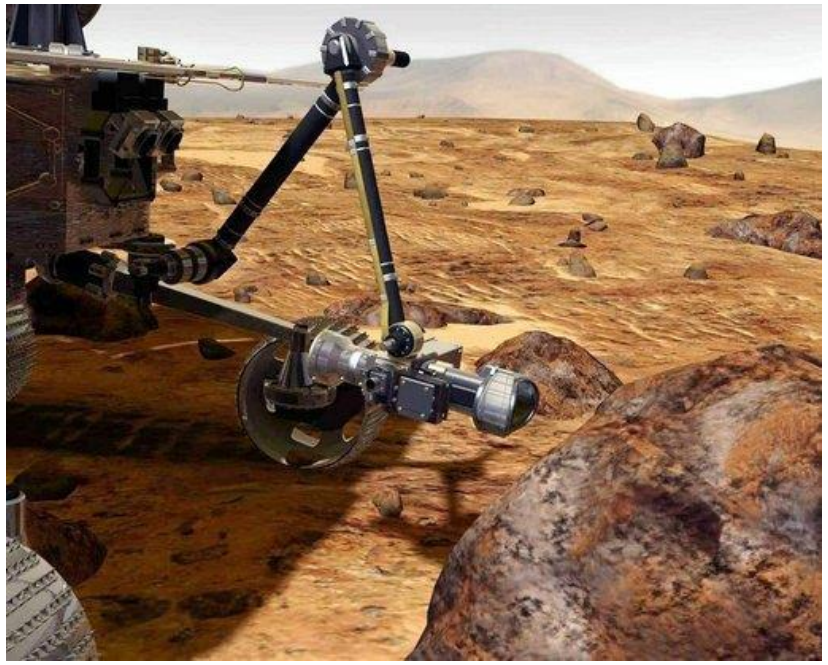
Seaglider robots can be on the open ocean for months at a time.

ON THE OTHER HAND

Robots aren’t very smart (now)

The Mars Rover Mission Control Room at JPL.

Nor can robots improvise.

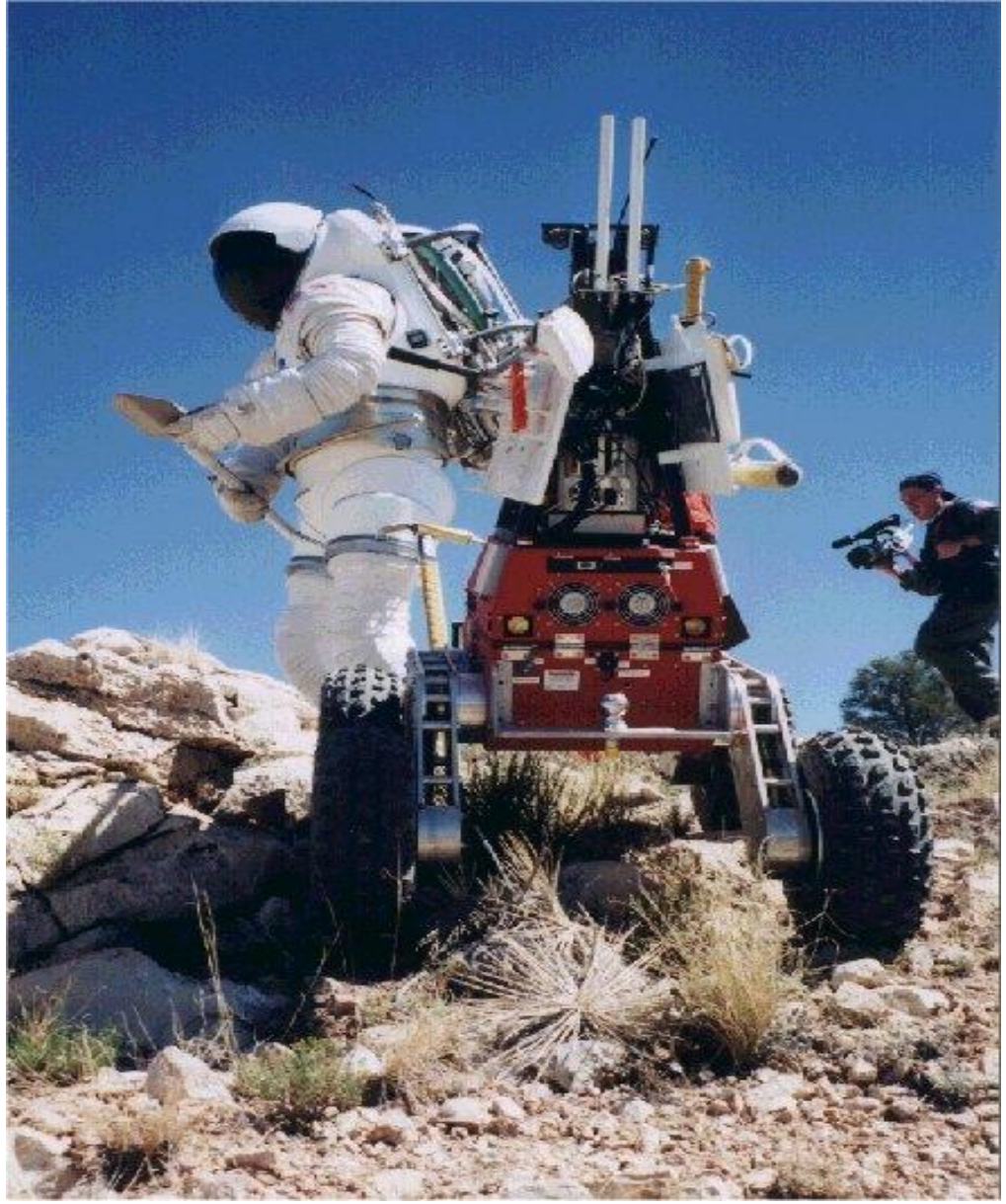


Vs.

Rover rock abrasion tool ($$$)

A rock hammer (¢)

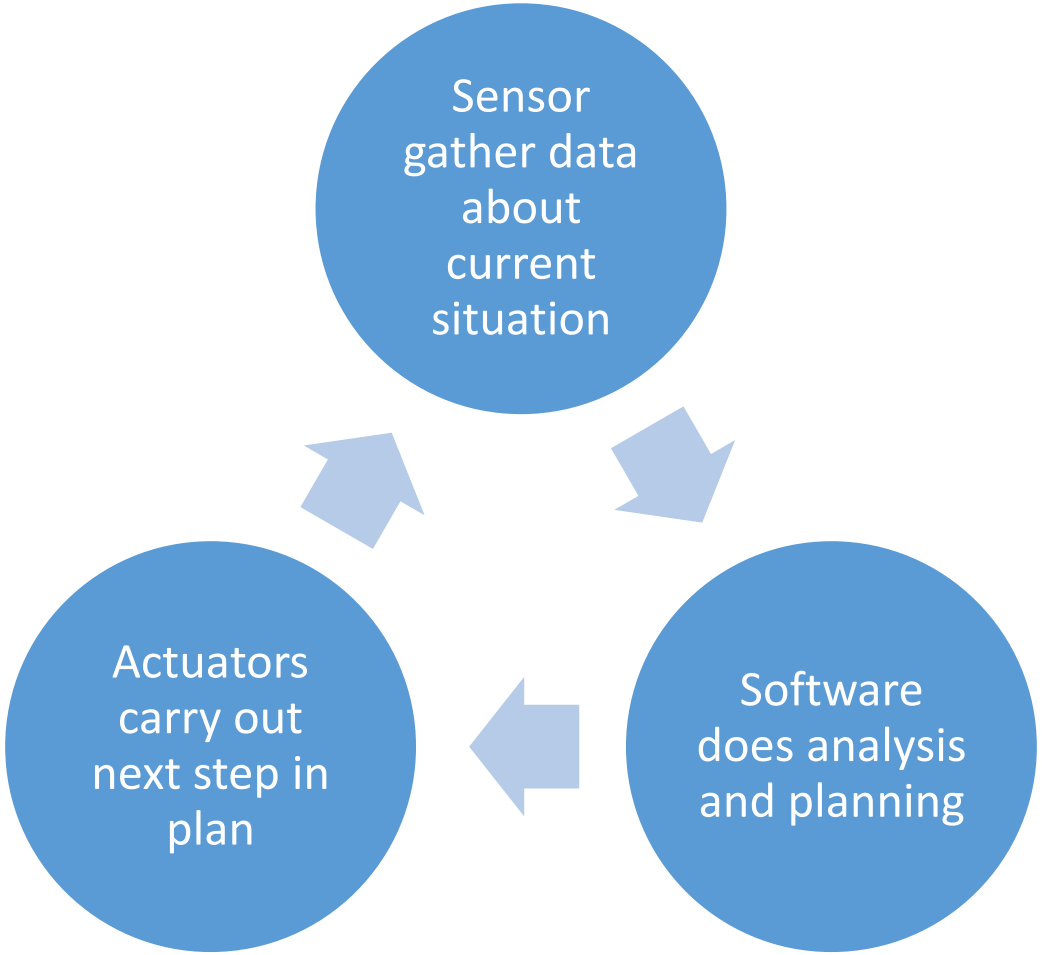
Luckily, robots can collaborate with humans



NASA’s “sidekicks”

Robotic architectures

The sense-think-act cycle



Pros and Cons

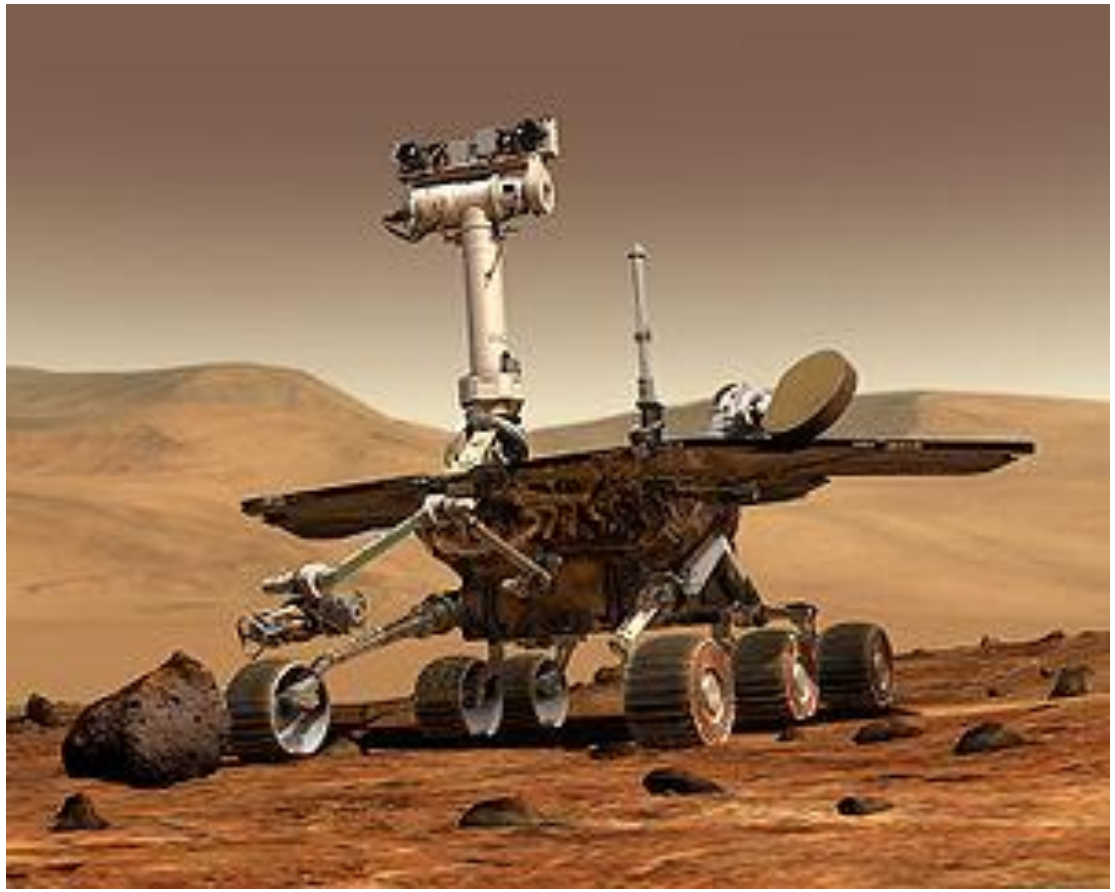
Pro:

- Easy to understand
- Modular at a high level
- Cautious (always working with as much information as possible)

Con:

- Can be very slow
- Stop-and-start
- Doesn’t handle surprises well
- Poor for dynamic reactive processes (e.g. walking)
- Not what people/animals really do

Example



Behavior-based/reactive

- Tight coupling between sensors and actuators, with little “thinking” in between
- Each part is responsible for its own task (e.g. foot must find stable position, leg must balance and push etc.)
- Minimal central control
- Overall behavior is emergent

Pros and Cons

Pro:

- Handles dynamic situations well
- Modular at the level of components
- No micro-management
- Relatively fast
- Reactive

Con:

- Sometimes unpredictable
- Very hardware specific
- Hard to program?
- High-level control needed at some point

Example

Sense-think-act robots:

- Shakey the Robot (very old school!): https://www.youtube.com/watch?v=qXdn6ynwpiI
- Mars Rovers: https://www.youtube.com/watch?v=7zpojhD4hpI
- **'In-class' exercise: Watch one (or both!) of these in full. Post about any connections between them and the course materials.**

Reactive robots:

- https://www.youtube.com/watch?v=sv35ItWLBBk&fea
- https://www.youtube.com/watch?v=M8YjvHYbZ9wture=related