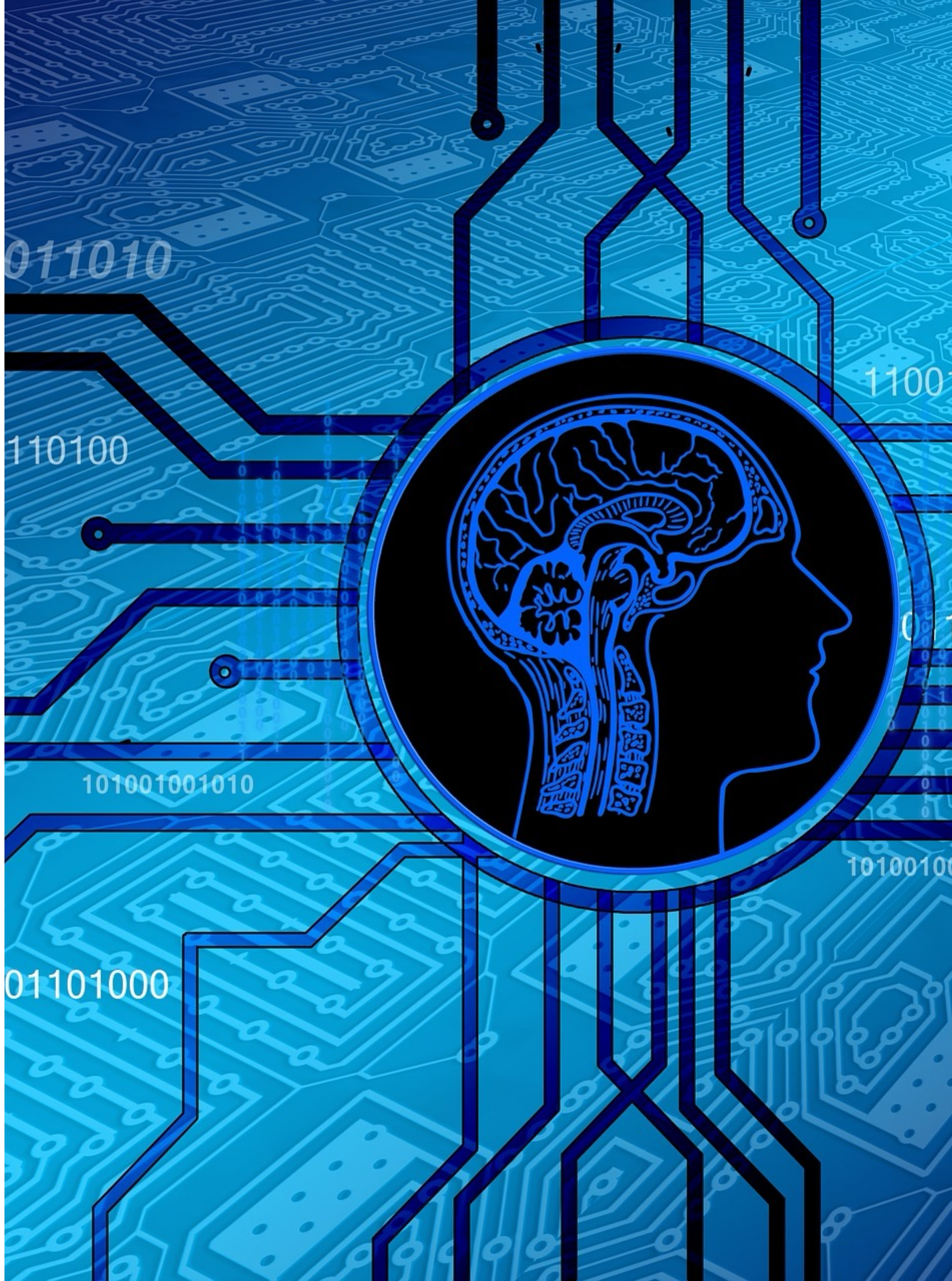
Revision
+ Broad Picture

Informatics 2D: Reasoning and Agents
The basics of search

Adapted from slides provided by Dr Petros Papapanagiotou



Revision

Intelligent Agents and their Environments

- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents
- Learning agents

- Properties of environments
  - Partially vs. fully observable
  - Deterministic vs. stochastic
  - Episodic vs. sequential
  - Static vs. dynamic
  - Discrete vs. continuous
  - Single vs. multi-agent

Problem Solving by Searching

- Problem formulation usually requires abstracting away real-world details to define a state space that can feasibly be explored.

- Variety of uninformed search strategies:
  - breadth-first, depth-first, iterative deepening

- Iterative deepening search uses only linear space and not much more time than other uninformed algorithms.

Evaluating search strategies

completeness: does it always find a solution if one exists?

time complexity: number of nodes generated / expanded

space complexity: maximum number of nodes in memory

optimality: does it always find a least-cost solution?

Time and space complexity are measured in terms of:

- ***b***: maximum branching factor of the search tree
- ***d***: depth of the least-cost solution
- ***m***: maximum depth of the state space (may be ∞)

If finite

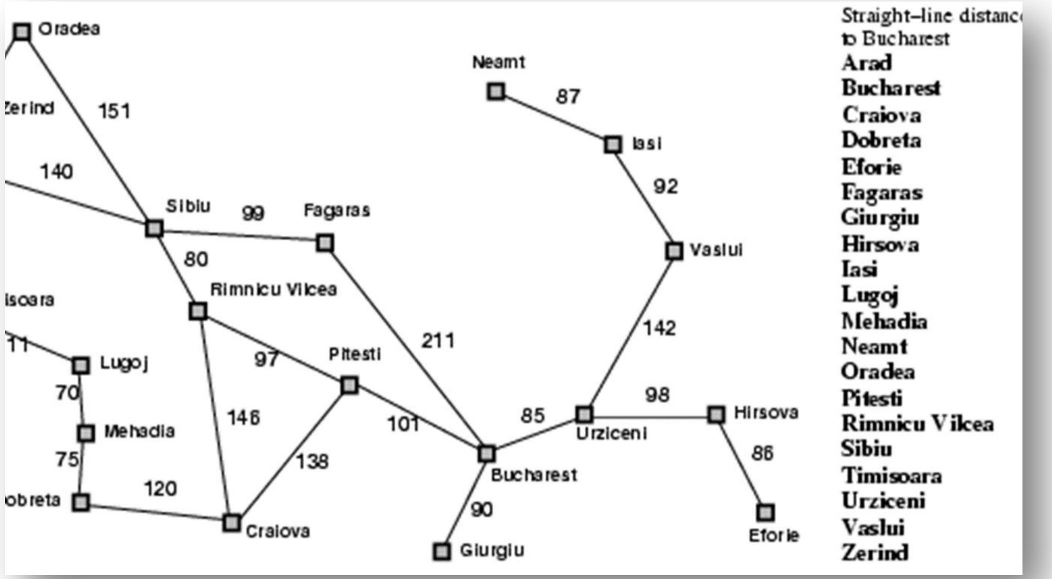
Criterion	Breadth-First	Uniform-Cost	Depth-First	Depth-Limited	Iterative Deepening	Bidirectional (if applicable)
Complete?	Yes	Yes[a,b]	No	No	Yes	Yes[a,d]
Time	$O(b^d)$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(b^m)$	$O(b^\ell)$	$O(b^d)$	$O(b^{d/2})$
Space	$O(b^d)$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(bm)$	$O(b\ell)$	$O(bd)$	$O(b^{d/2})$
Optimal?	Yes	Yes	No	No	Yes	Yes[c,d]

If cost = 1 per step

Summary of Base Algorithms

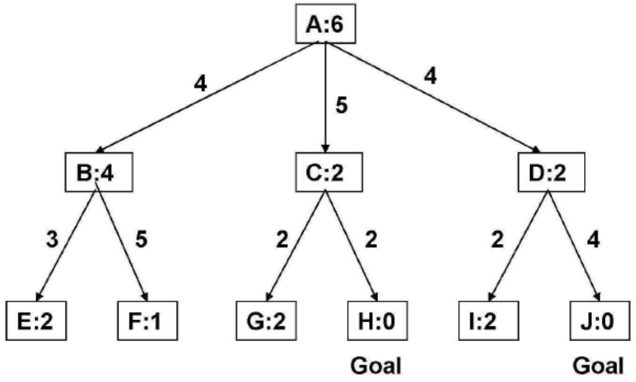
Informed Search

- Smart search based on heuristic scores
  - Best-first search
  - Greedy best-first search
  - A* search
  - Admissible heuristics and optimality.



A* Search

3. Consider the following search tree in which the nodes represent states and the arcs represent the moves connecting these states. Each node is labelled by a letter. The numbers on the arcs represent the *true* cost of the associated move. The numbers on the nodes represent the *estimated* cost of reaching the goal state from that node.



In which order would the A^* algorithm explore this search tree?

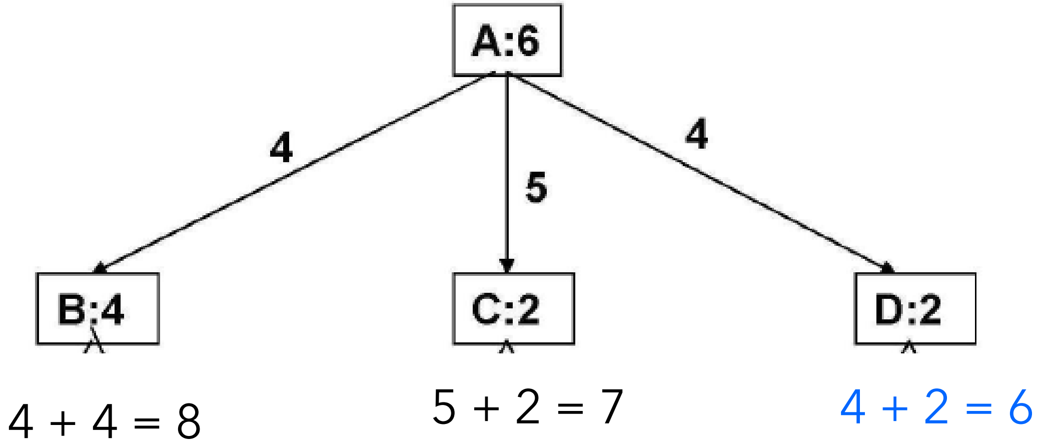
(a) A, B, C, D, I, J, G, H.
(b) A, B, C, D, I, J.
(c) A, C, H.
(d) A, D, I, J.
(e) A, C, G, H.

Example

A* search

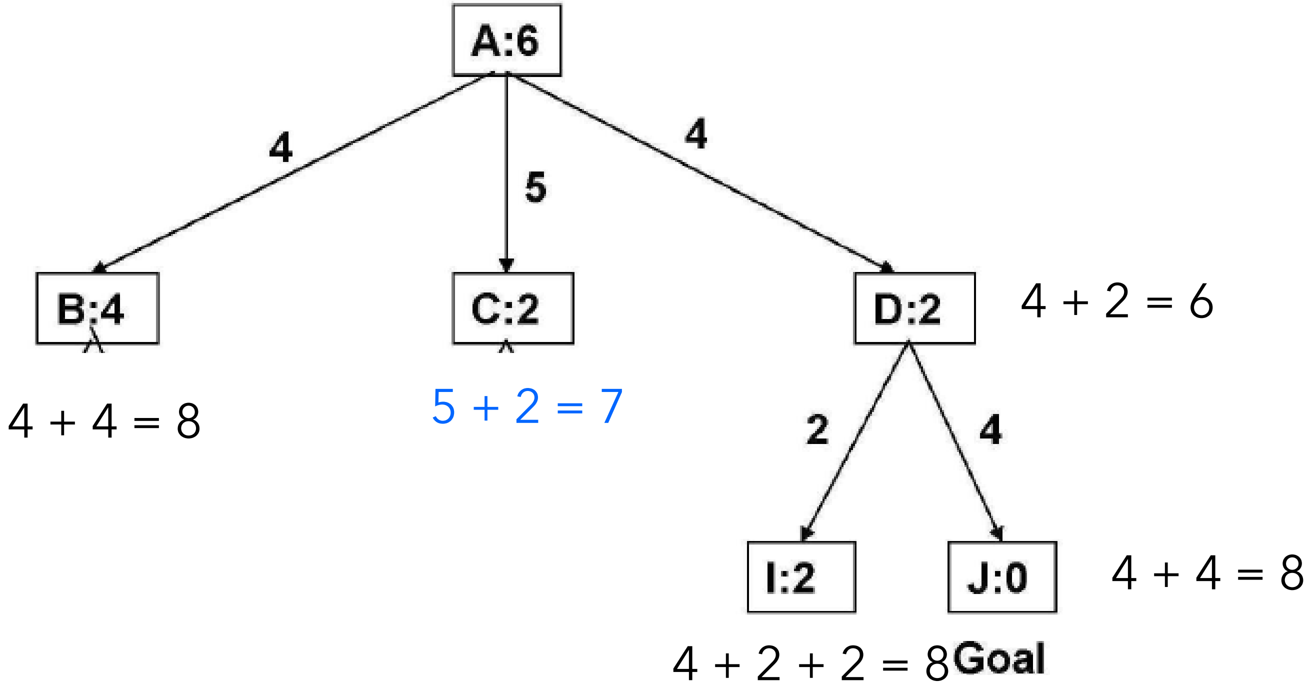
- Evaluation function $f(n) = g(n) + h(n)$
  - $g(n)$ = cost so far to reach n
  - $h(n)$ = estimated cost from n to goal
  - $f(n)$ = estimated total cost of path through n to goal
- Avoid expanding paths that are already expensive

Example



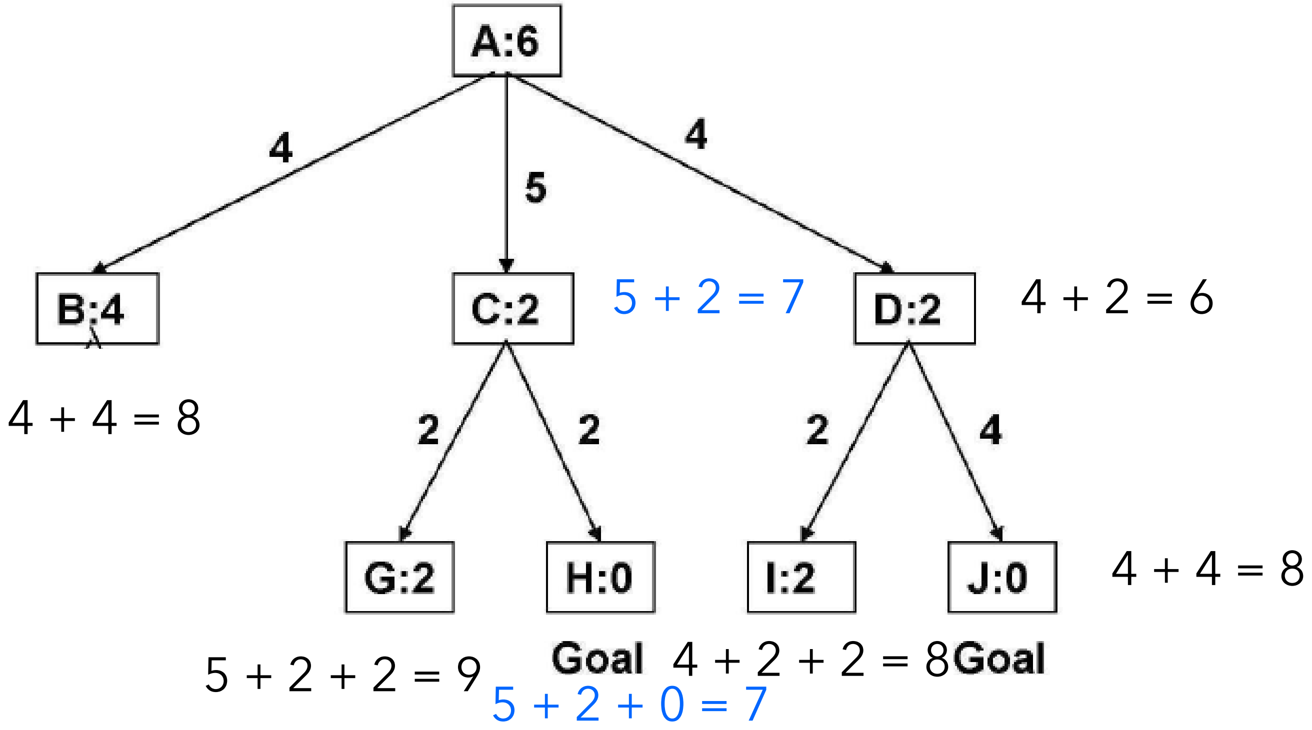
A
B C D

Example



A
B C D
I J

Example

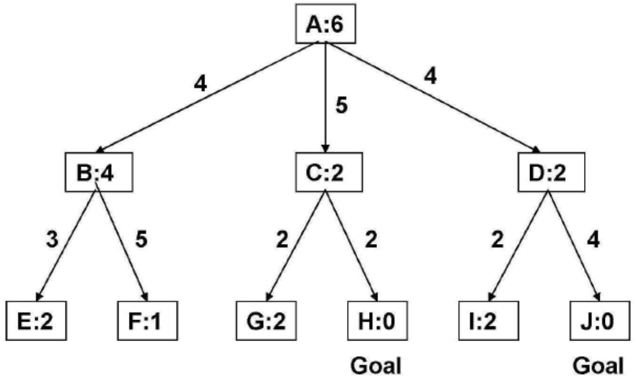


A
B C D
I J
G H

We're done as we've expanded a node containing a goal state

A* Search

3. Consider the following search tree in which the nodes represent states and the arcs represent the moves connecting these states. Each node is labelled by a letter. The numbers on the arcs represent the *true* cost of the associated move. The numbers on the nodes represent the *estimated* cost of reaching the goal state from that node.



In which order would the A^* algorithm explore this search tree?

(a) A, B, C, D, I, J, G, H.
(b) A, B, C, D, I, J.
(c) A, C, H.
(d) A, D, I, J.
(e) A, C, G, H.

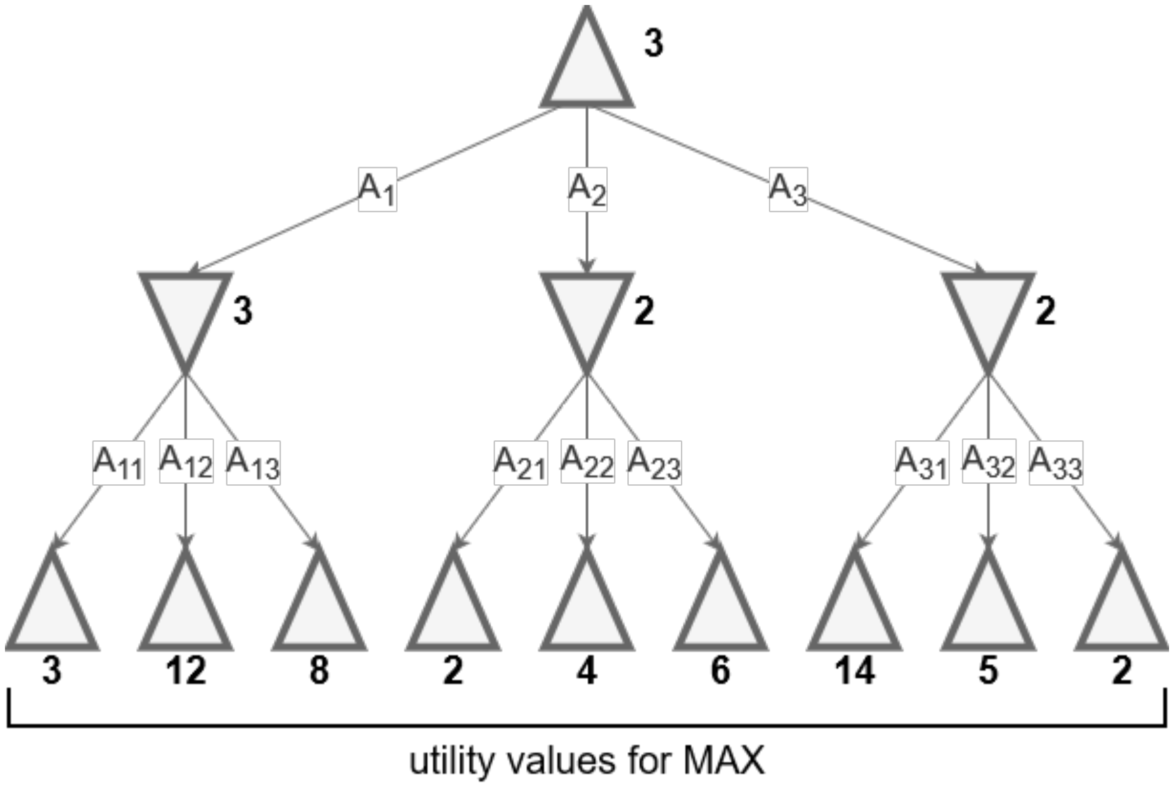
Example

Smart Searching Using Constraints

- Constraint Satisfaction Problems (CSPs):
  - states defined by values of a fixed set of variables
  - goal test defined by constraints on variable values
- Backtracking = depth-first search with one variable assigned per node.
- Variable ordering and value selection heuristics help significantly.
- Forward checking prevents assignments that guarantee later failure.
- Constraint propagation (e.g., arc consistency) does additional work to constrain values and detect inconsistencies.

Adversarial Search

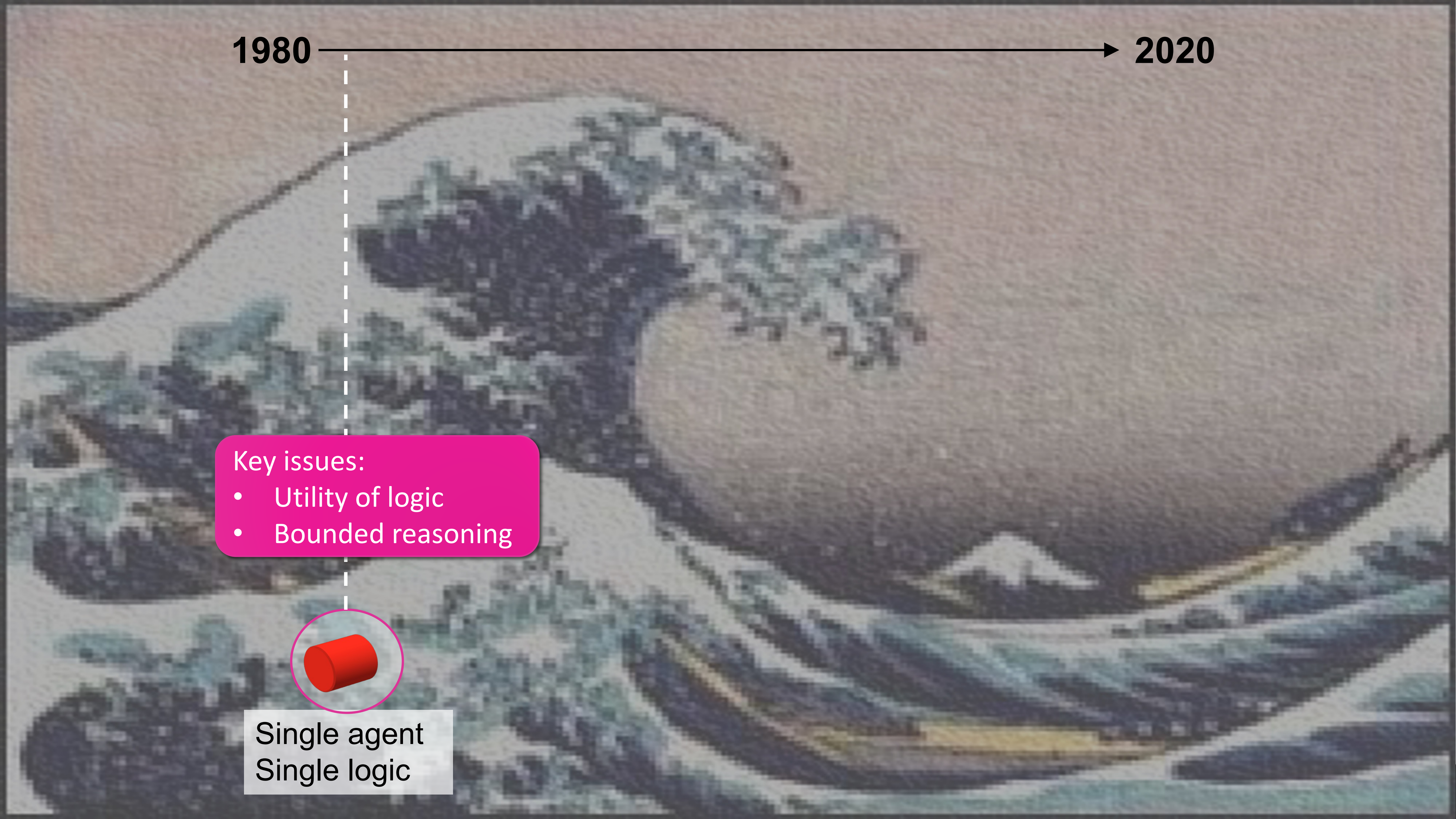
- Minimax assumes that both players play optimally
- Informally: Each agent is making its decision for the next move based on the assumption that the other agent is playing as well as it can.

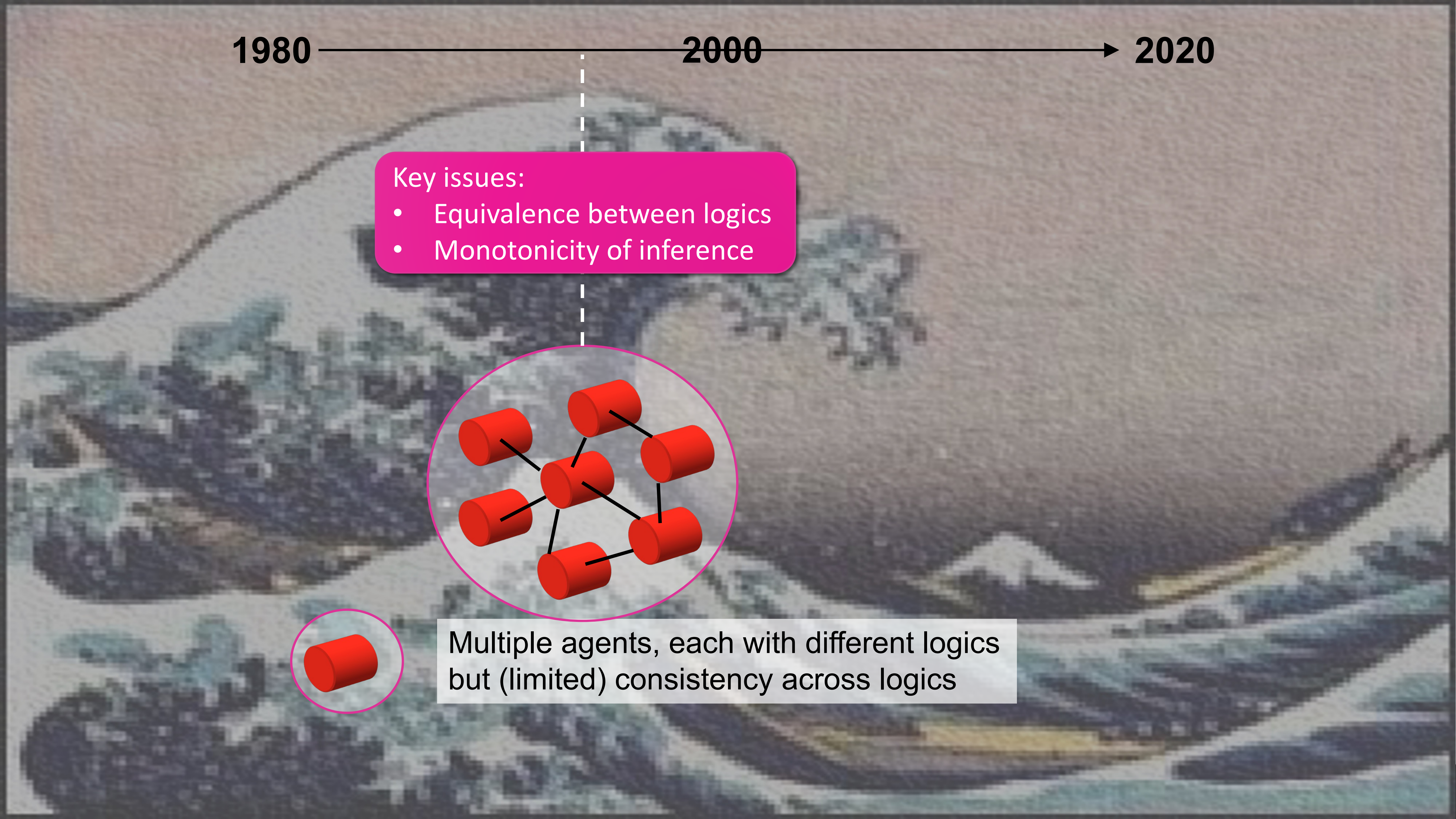


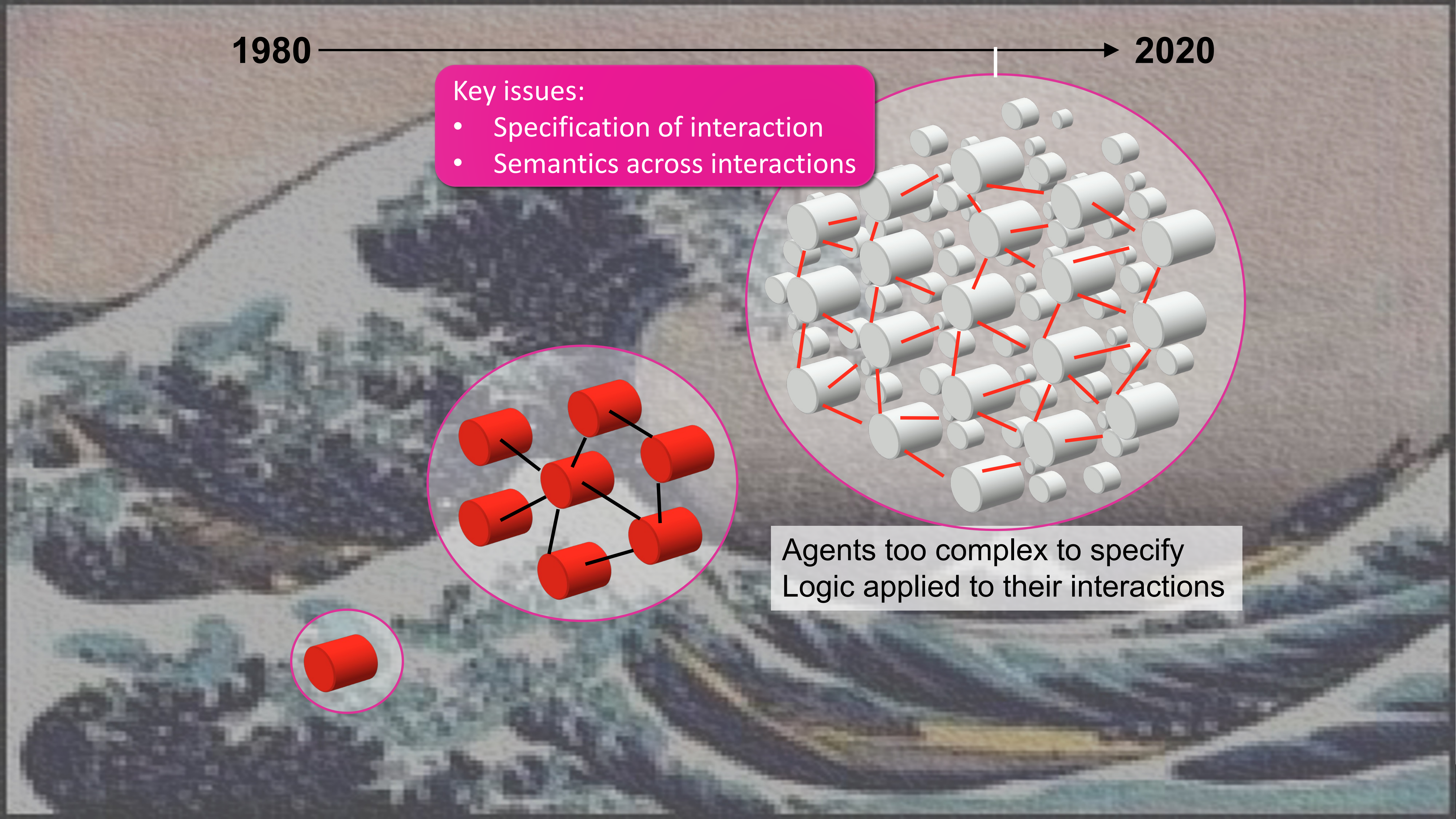
Adversarial Search (Contd)

- α-β Pruning and its properties
- Reasoning about relevant computations only enables search space to be pruned.
- How to deal with deep trees: need for evaluation functions.

Broad Picture


1980
2020
Key issues:
• Utility of logic
• Bounded reasoning
Single agent
Single logic


1980
2000
2020
Key issues:
• Equivalence between logics
• Monotonicity of inference
Multiple agents, each with different logics
but (limited) consistency across logics

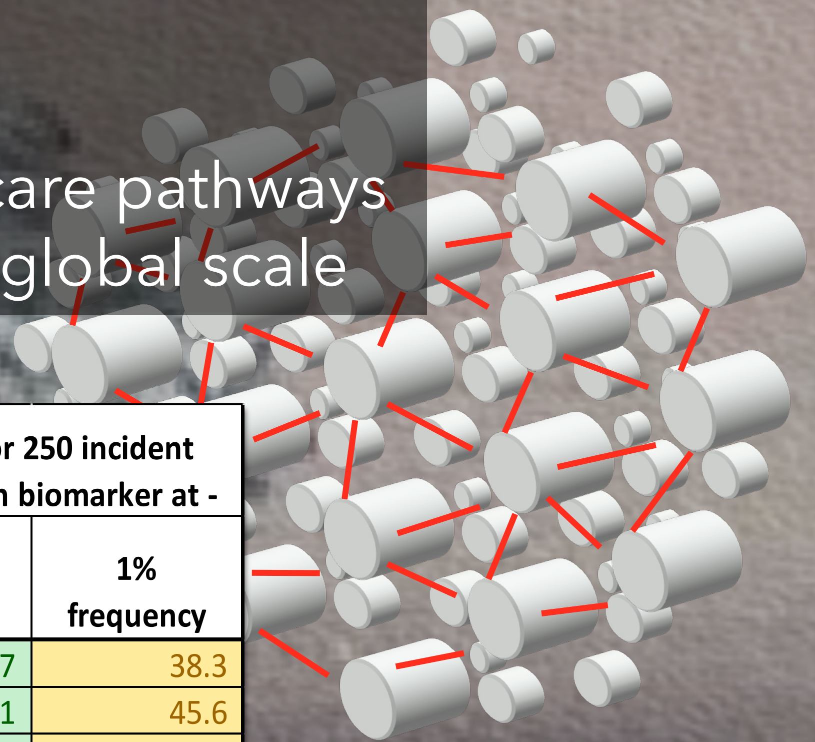

1980
2020
Key issues:
• Specification of interaction
• Semantics across interactions
Agents too complex to specify
Logic applied to their interactions

1980 → 2020

In healthcare:

- (cylinder) are data sources
- (red line) are created by experiments and care pathways

and precision medicine drives to global scale



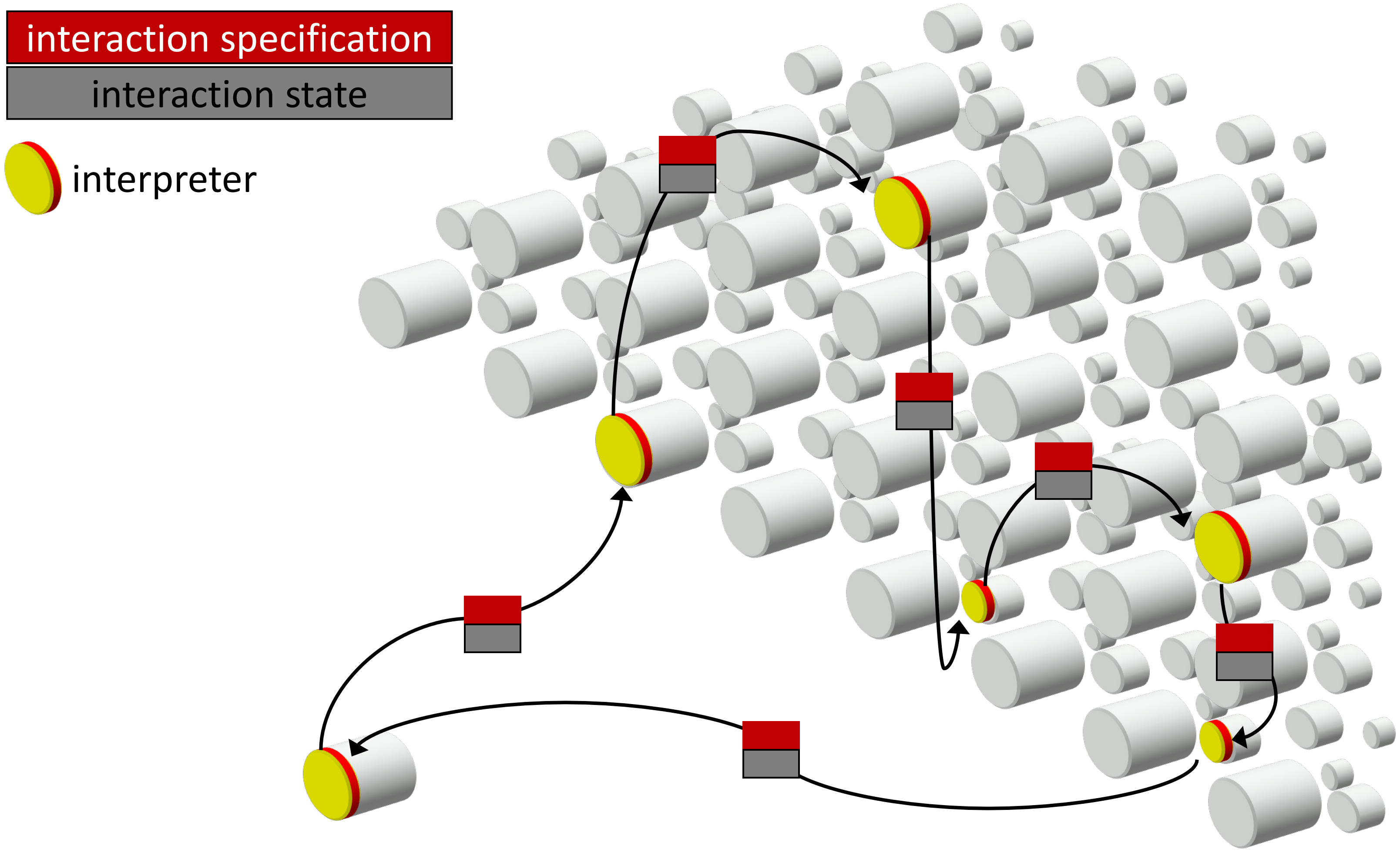
Rank	Site	Incidence (per M)	Mortality	Millions of population for 250 incident cancer patients a year with biomarker at - 20% frequency	5% frequency	1% frequency
1	Breast	653	23%	1.9	7.7	38.3
2	Prostate	548	26%	2.3	9.1	45.6
3	Lung	530	83%	2.4	9.4	47.1
4	Colon	340	37%	3.7	14.7	73.5
5	Melanoma	167	17%	7.5	29.9	149.4
10	Pancreas	111	94%	11.3	45.1	225.6
14	Ovary	87	62%	14.4	57.5	287.7
20	Liver	56	89%	22.4	89.5	447.3
25	Cervix	36	33%	34.5	138.2	690.9
30	Larynx	29	32%	43.3	173.4	866.9

Key:		
	8.0	*Green - achievable in a large region like Scotland or East of England*
	40.0	*Amber -achievable across the whole of the UK*
	64.0	*White - requires international cooperation*

Global Alliance for Genomics & Health

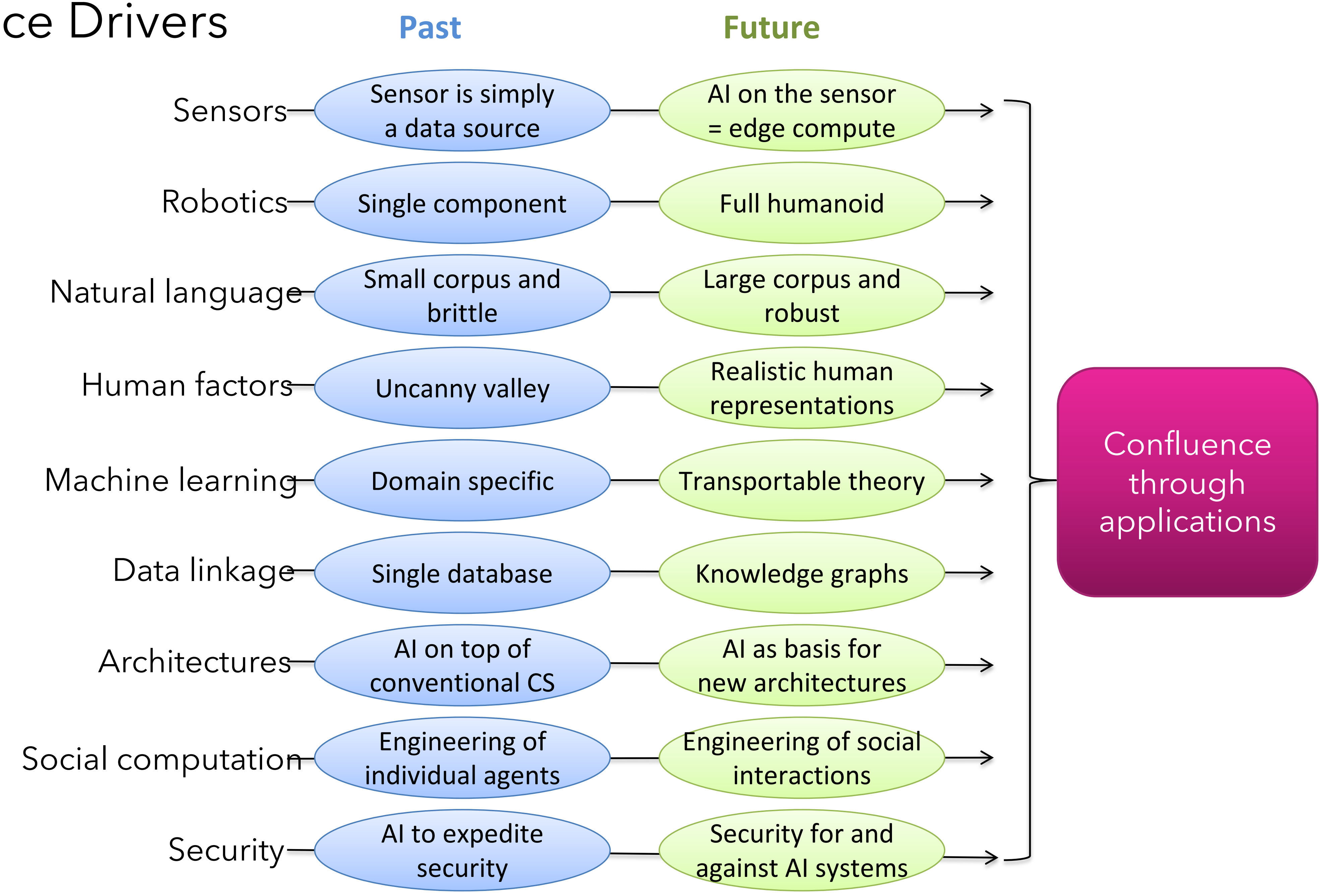
HDRUK Health Data Research UK

If Interactions are Specifications then They Can be Distributed to Autonomous Agents (if the Agents can Interpret Them)



Numerous peer to peer infrastructures were built for this
(similar to the idea of smart contracts in distributed ledger systems)

Science Drivers



Examples of Confluence