Mario AI Competition @ CIG 2009

Sergey Karakovskiy and Julian Togelius

http://julian.togelius.com/mariocompetition2009

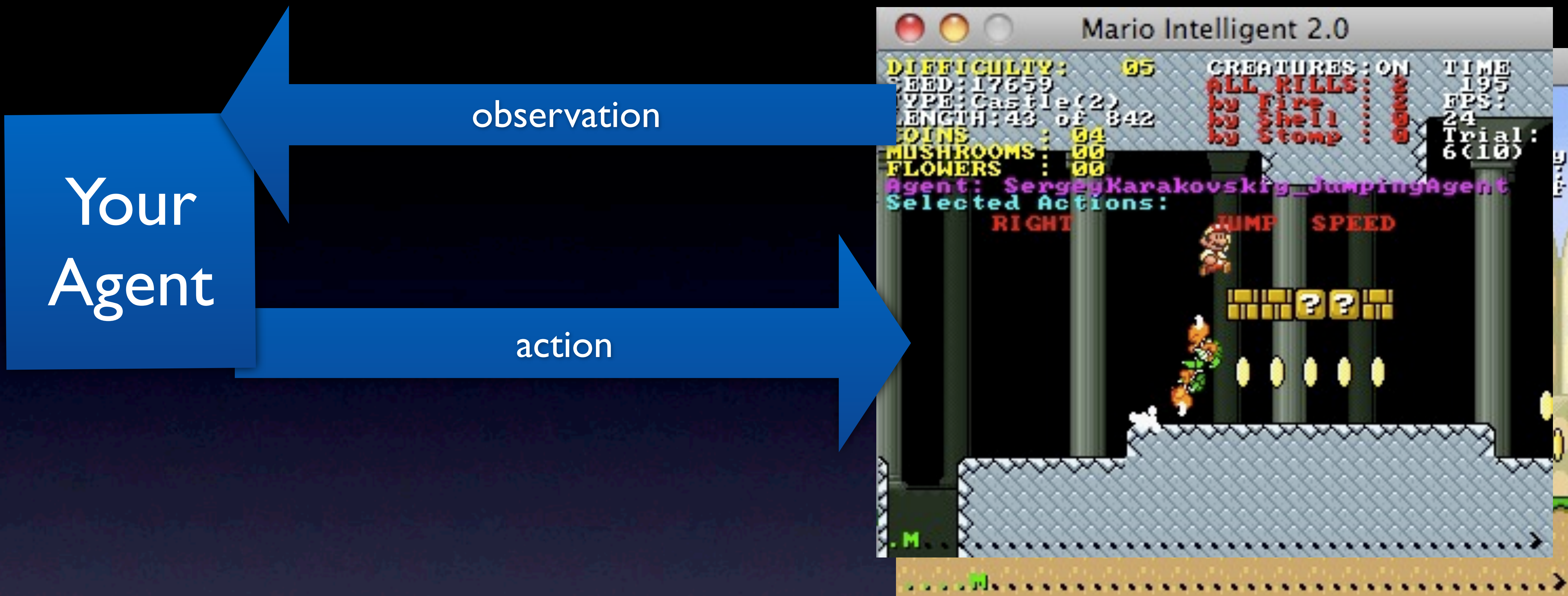
Infinite Mario Bros

- by Markus Persson
- quite faithful SMB 1/3 clone
- in Java
- random level generation
- open source

Making a benchmark

- The control loop rewritten
- Tunable FPS, up to 1000 times faster than real-time
- Created an interface for any type of agents or controllers
- Removed stochasticity and unpredictable randomness in behaviour of the benchmark

Interface



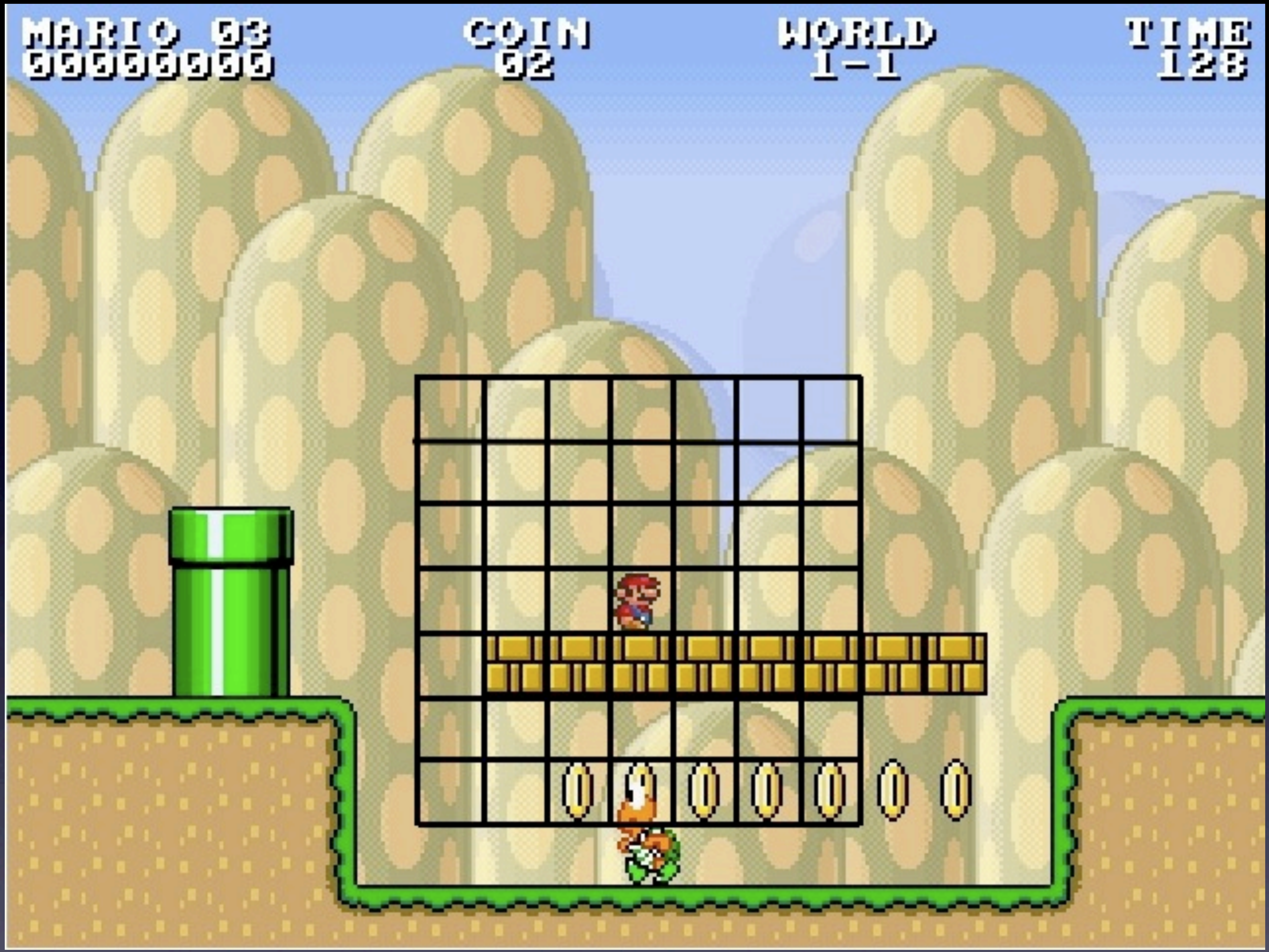
Develop a controller/agent (based on AI/ machine learning?) for "Super Mario Bros"

results

Score: 13998.645
Levels cleared = 9
Total time left = 6780
Total kills = 87
Mario mode = 32
TOTAL SUM = 20906.645

Interface

- Each time step the agent gets a representation of the environment
  - Enemies and “blocks” around Mario
  - Fine position, jumping state
  - If Mario is carrying a shell
- And returns an action
  - 5 bits: left, right, down, A, B



Interface

Environment Interface

- 22x22 arrays describing
  - landscape features (e.g. walls, cannons, gaps)
  - creatures
- Fine position of Mario and creatures
- Booleans: mario is on the ground, may jump, is carrying a shell, is small/big/fire

Agent Interface

- getAction(Environment environment);

Very simple rule-based agent

```
public boolean[] getAction(Environment
observation) {

action[Mario.KEY_SPEED] =
action[Mario.KEY_JUMP] =
observation.mayMarioJump() || !
observation.isMarioOnGround();

return action;}
```

Media

- Reddit
- Slashdot
- New Scientist
- Le Monde
- Discovery Channel / MSNBC
- lots of blogs, gaming news sites etc.

Agent goals

- Develop an agent that gets as far and as fast as possible...
- ...on as many levels as possible...
- ...which are previously unseen
- Scoring: progress on 40 randomly generated levels (of different difficulty, length, type) with seed 17564
- If two agents complete all the levels: tiebreakers

Tiebreakers

- Total time left (in Marioseconds)
- Total kills
- MarioMode sum (small, large, fire)

Rules

- Implement the Agent interface (or connect to the TCP ServerAgent)
- Use only information from the Environment interface
- Don't take more than 40 ms per time step in average

Agent challenges

- Handle a large state/observation space
- Handle very different situations (unlike e.g. car racing)
- Tactical tradeoffs (go back and get the power-up?)

Presentations of competitors

(in alphabetical order)

Robin Baumgarten

Using path-finding to find the optimal jump

AN A* MARIO AI

IDEA

- Analyse Mario's physics engine to obtain movement equations for all objects
- Create our own physics engine that can predict next world state
- Plug engine into an A* algorithm to evaluate fitness of each node
- Heuristic: How long before Mario reaches goal?
- Penalty for falling into gaps or being hurt
- Ignore coins, enemies, power-ups (for now!)

A* ALGORITHM

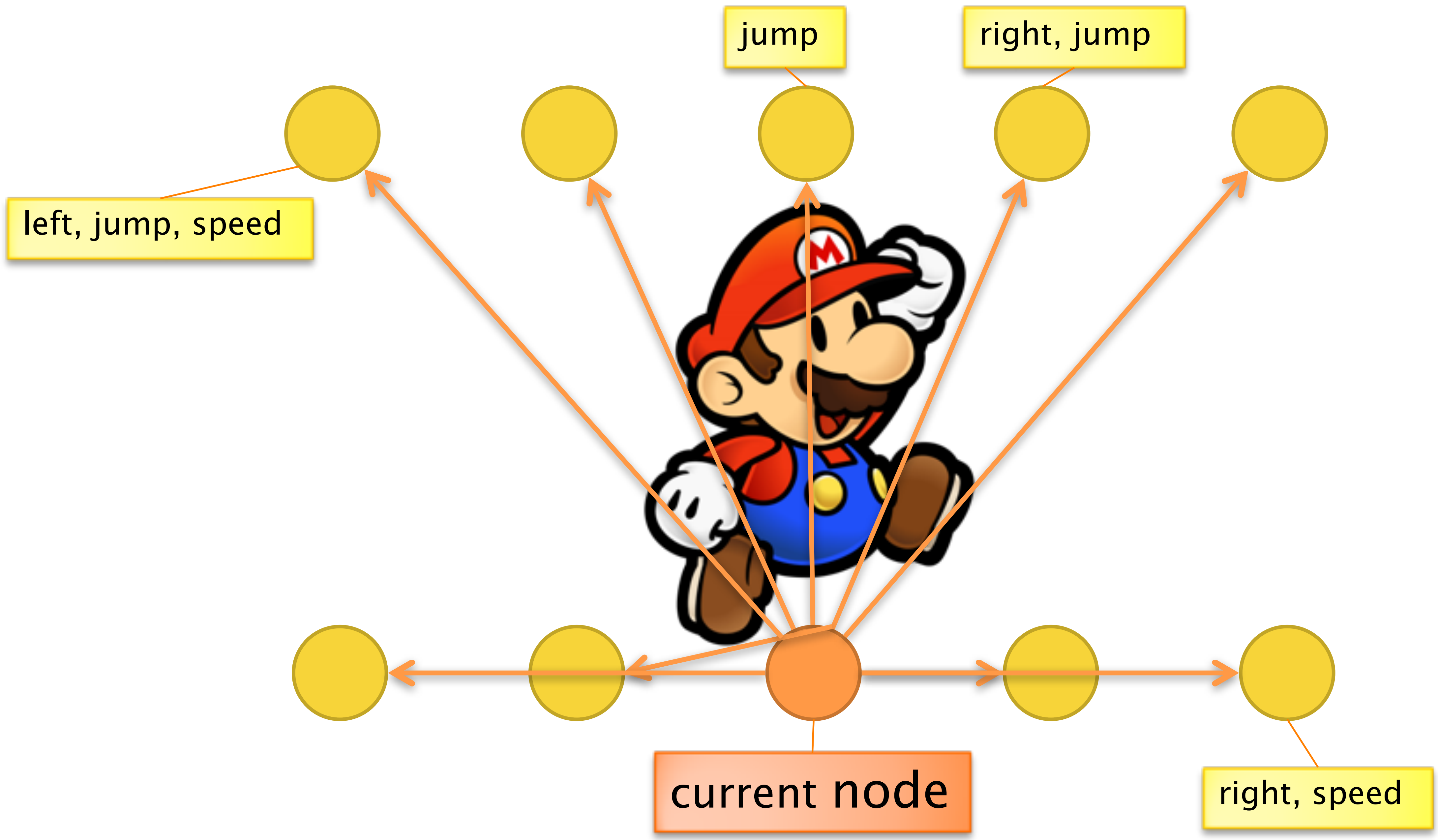
- Best-first graph search algorithm
- Need heuristic that estimates remaining distance
- Keep set of “open” nodes (initially: start node)
- While open set not empty:
  - Pick node in open set with **lowest estimated total distance from start to goal**
  - If node == goal: finish. Create path by backtracking through ancestors.
  - Generate child nodes, put them into open list (only if better than existing nodes for that location)
- If heuristic admissible (always underestimating), we then have the shortest path to goal.

A* IN MARIO: CURRENT POSITION

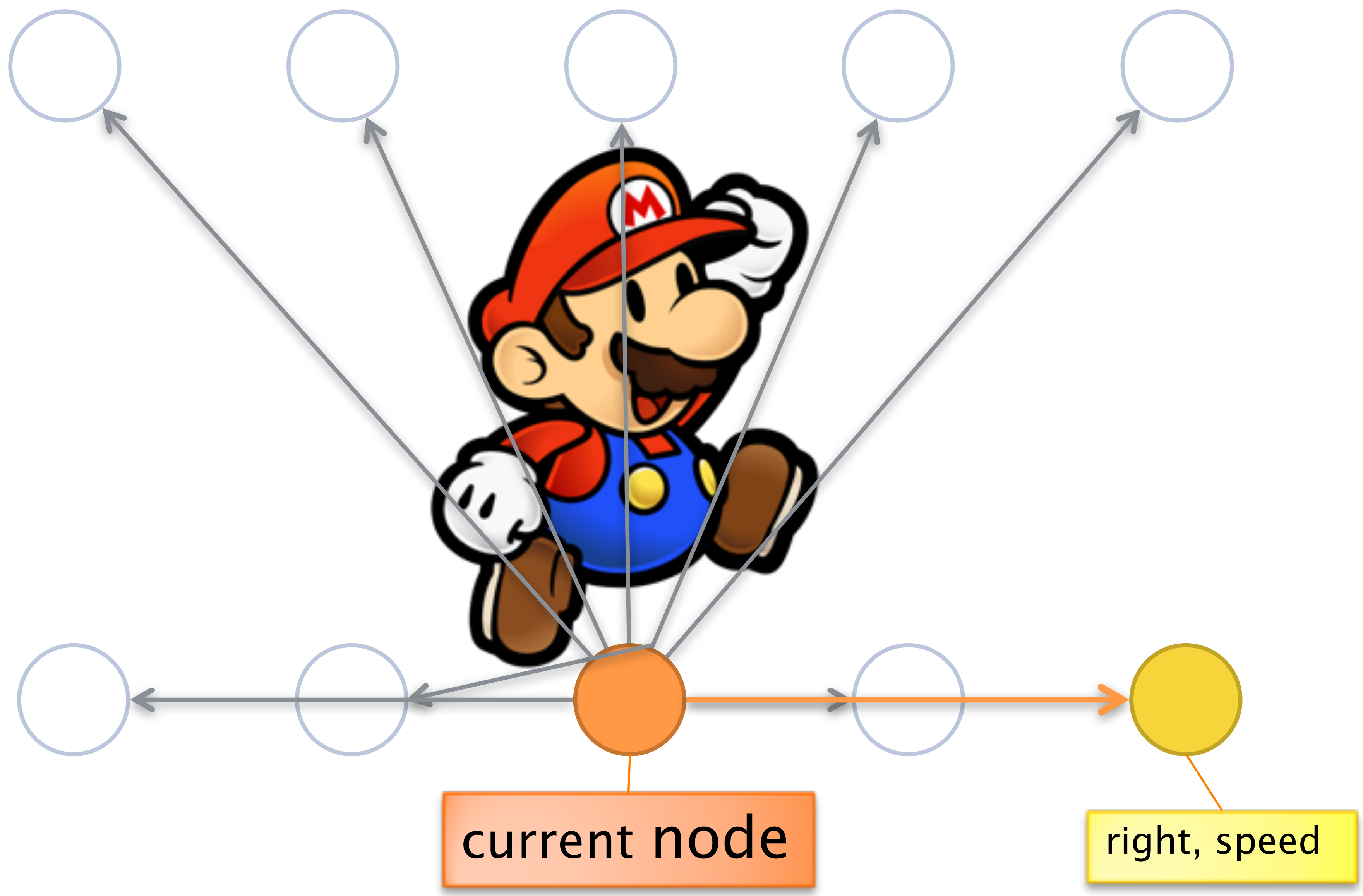
current node

Goal:
right border
of screen

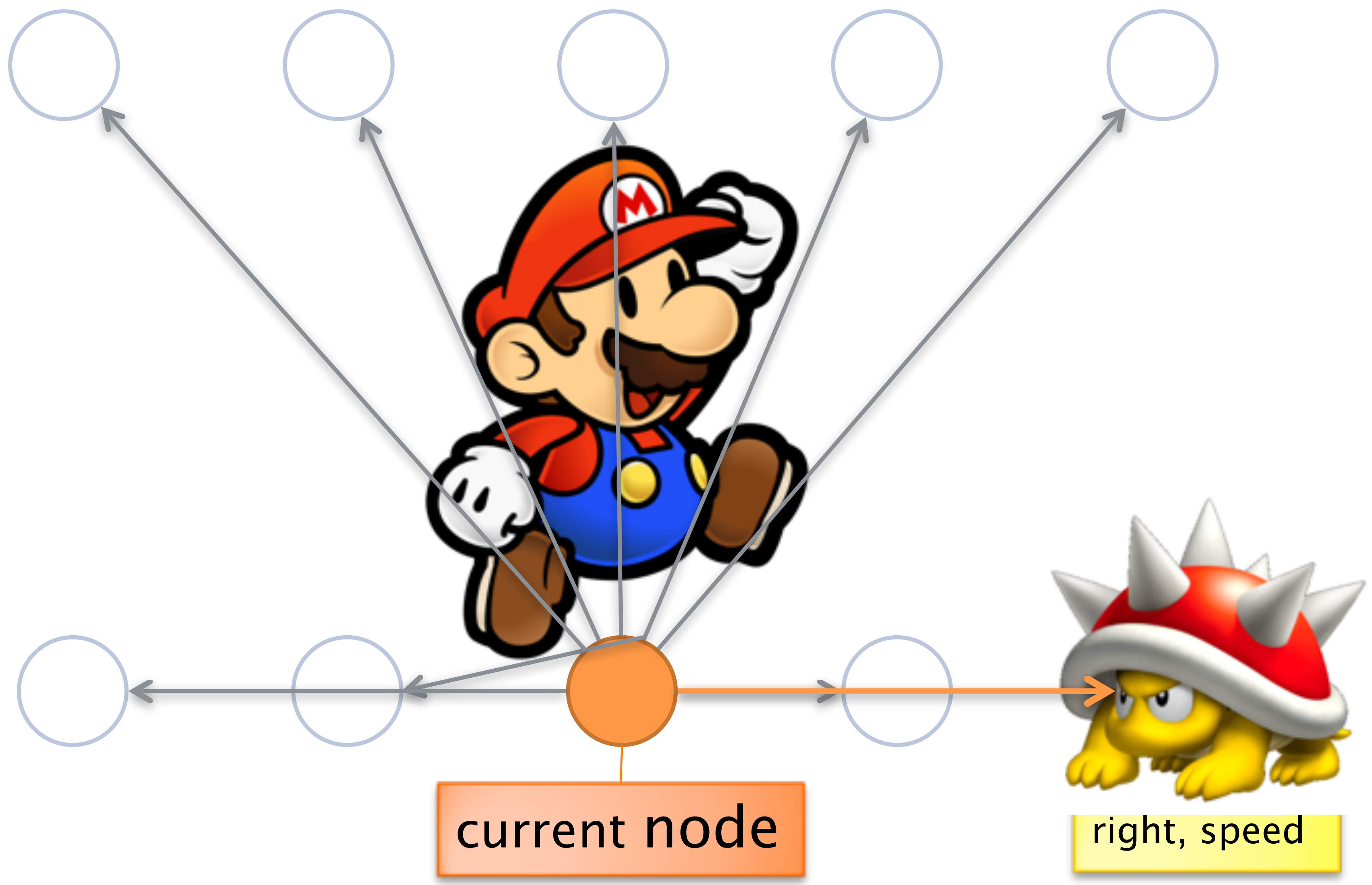
A* IN MARIO: CHILD NODES



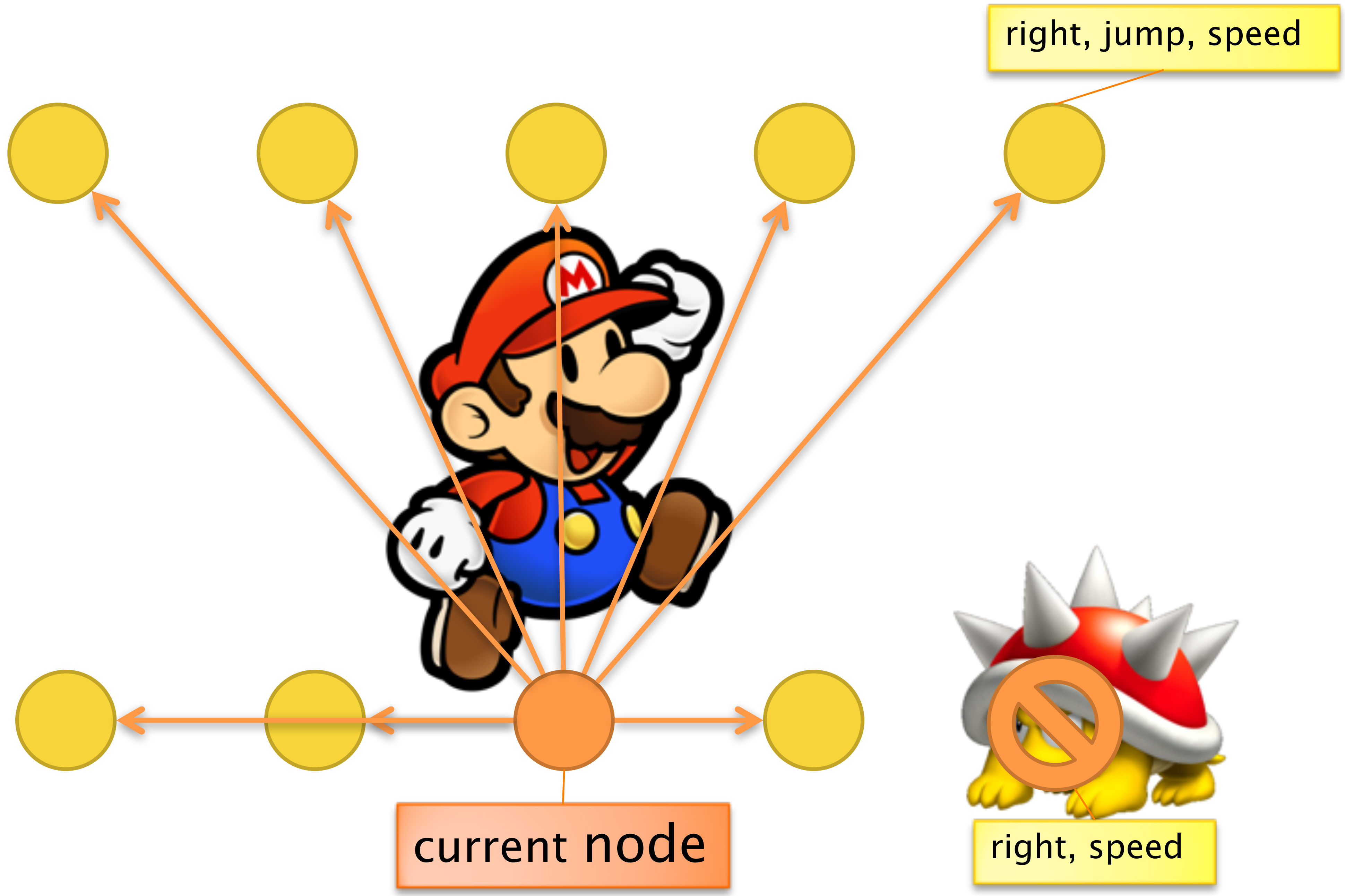
A* IN MARIO: BEST FIRST



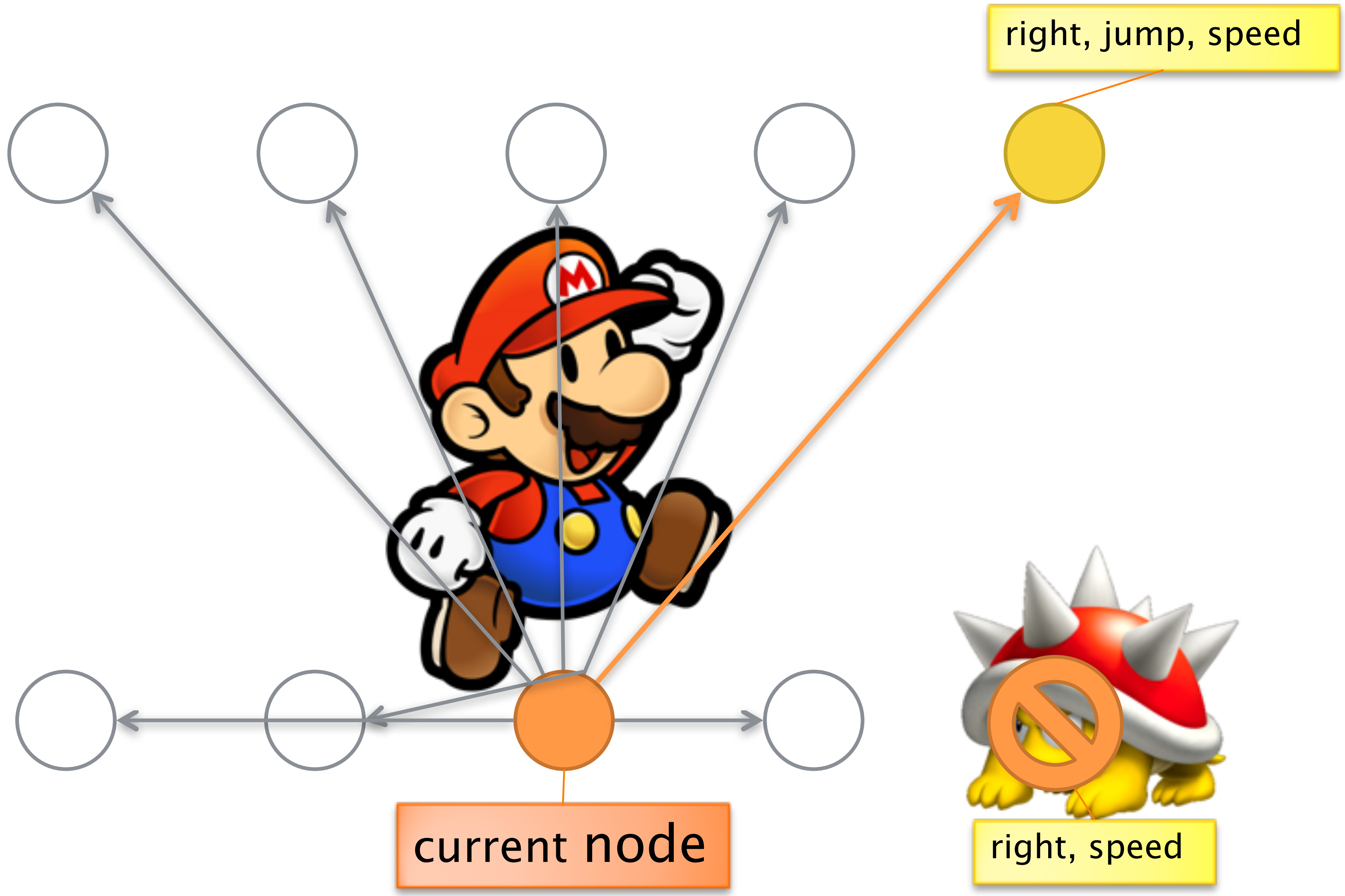
A* IN MARIO: EVALUATE NODE



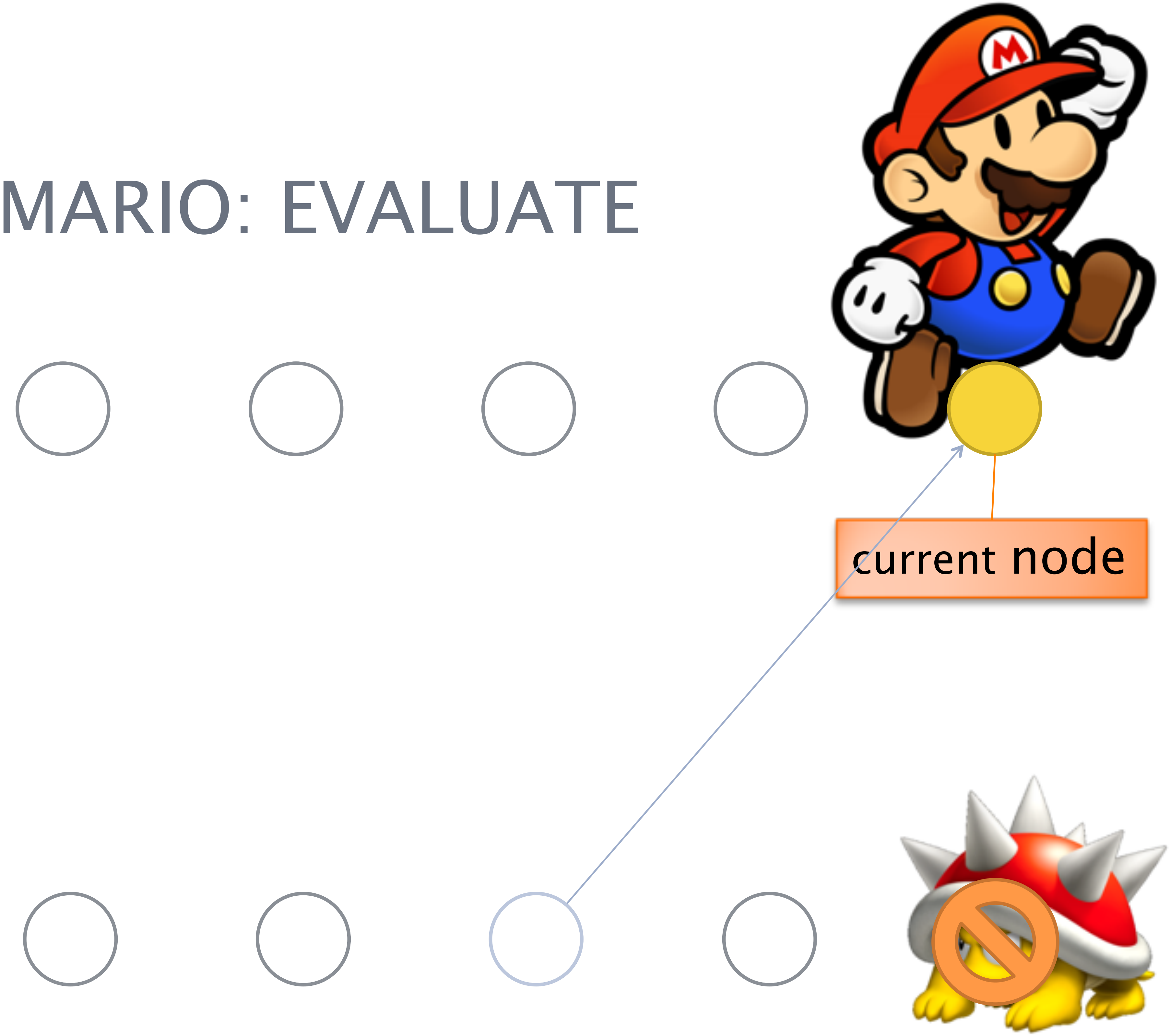
A* IN MARIO: BACKTRACK



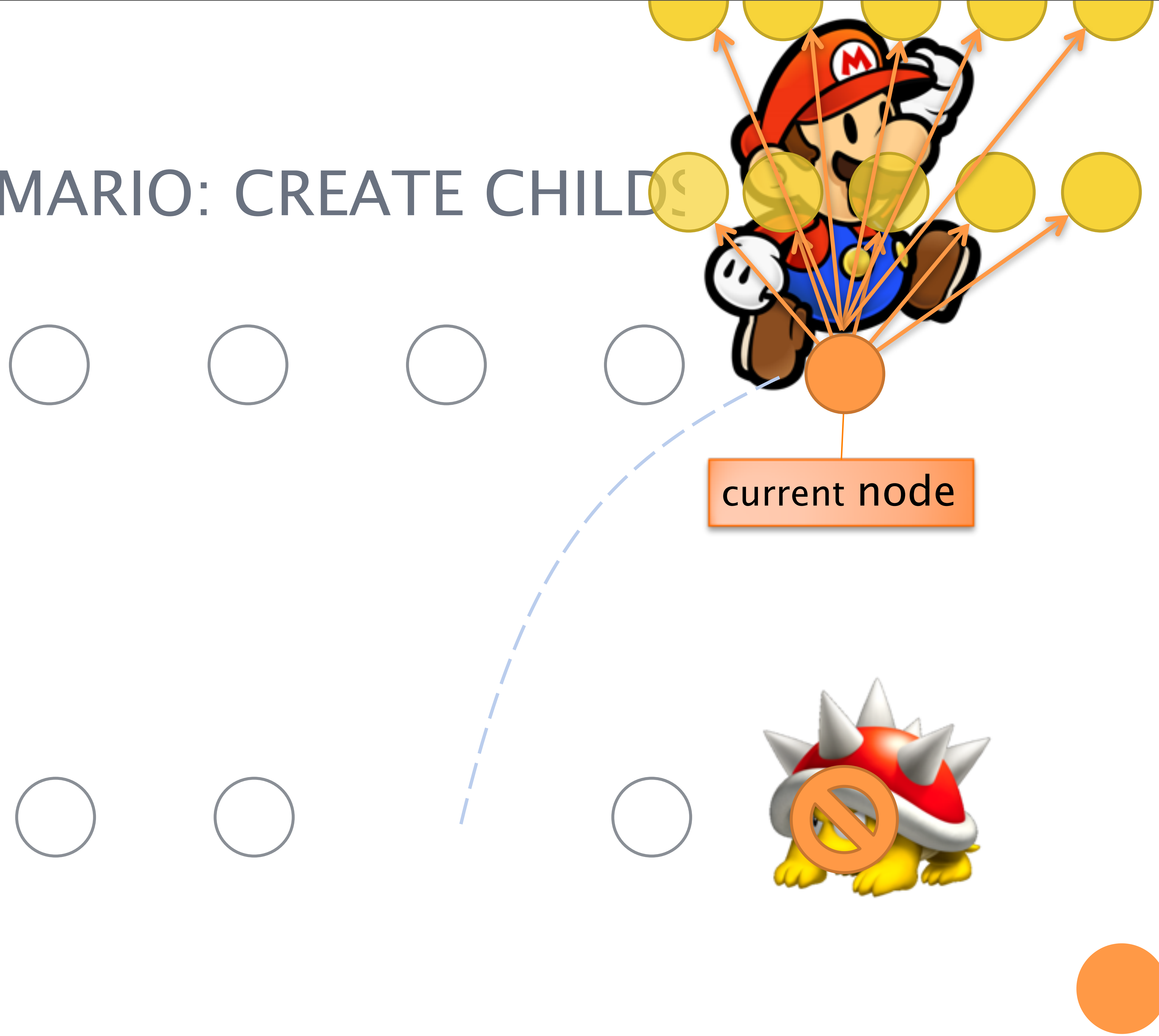
A* IN MARIO: BEST FIRST



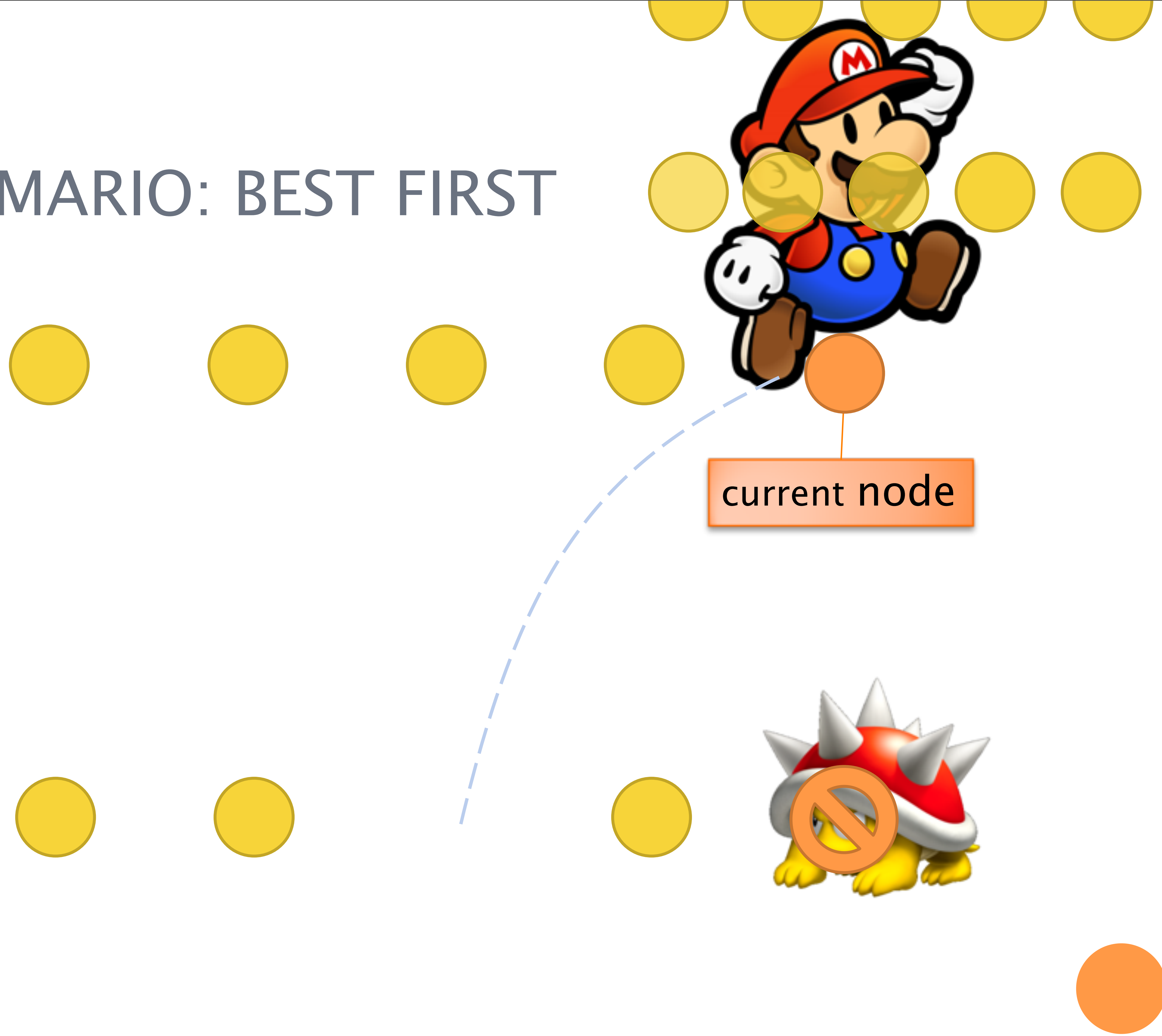
A* IN MARIO: EVALUATE



A* IN MARIO: CREATE CHILDS



A* IN MARIO: BEST FIRST



HEURISTIC

- Using Mario's current speed and acceleration, how long does it take to reach the goal?
- Assume maximum acceleration and no obstacles (admissible heuristic!)

$$xa = xa + 1.2$$
$$x = x + xa$$
$$xa = xa * 0.89$$

- Optimisation: Find a closed form for this.

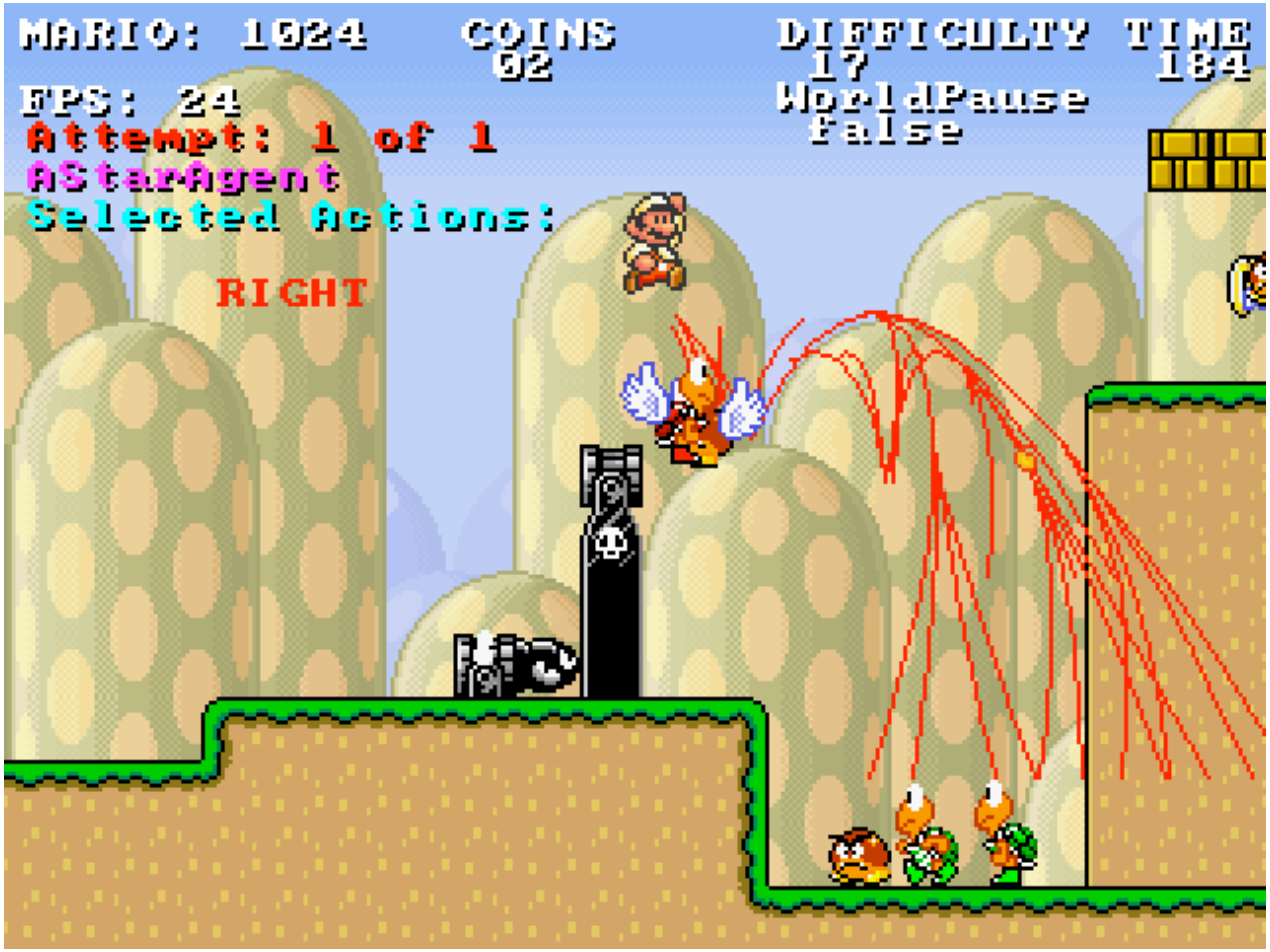
HANDLING NEW EVENTS

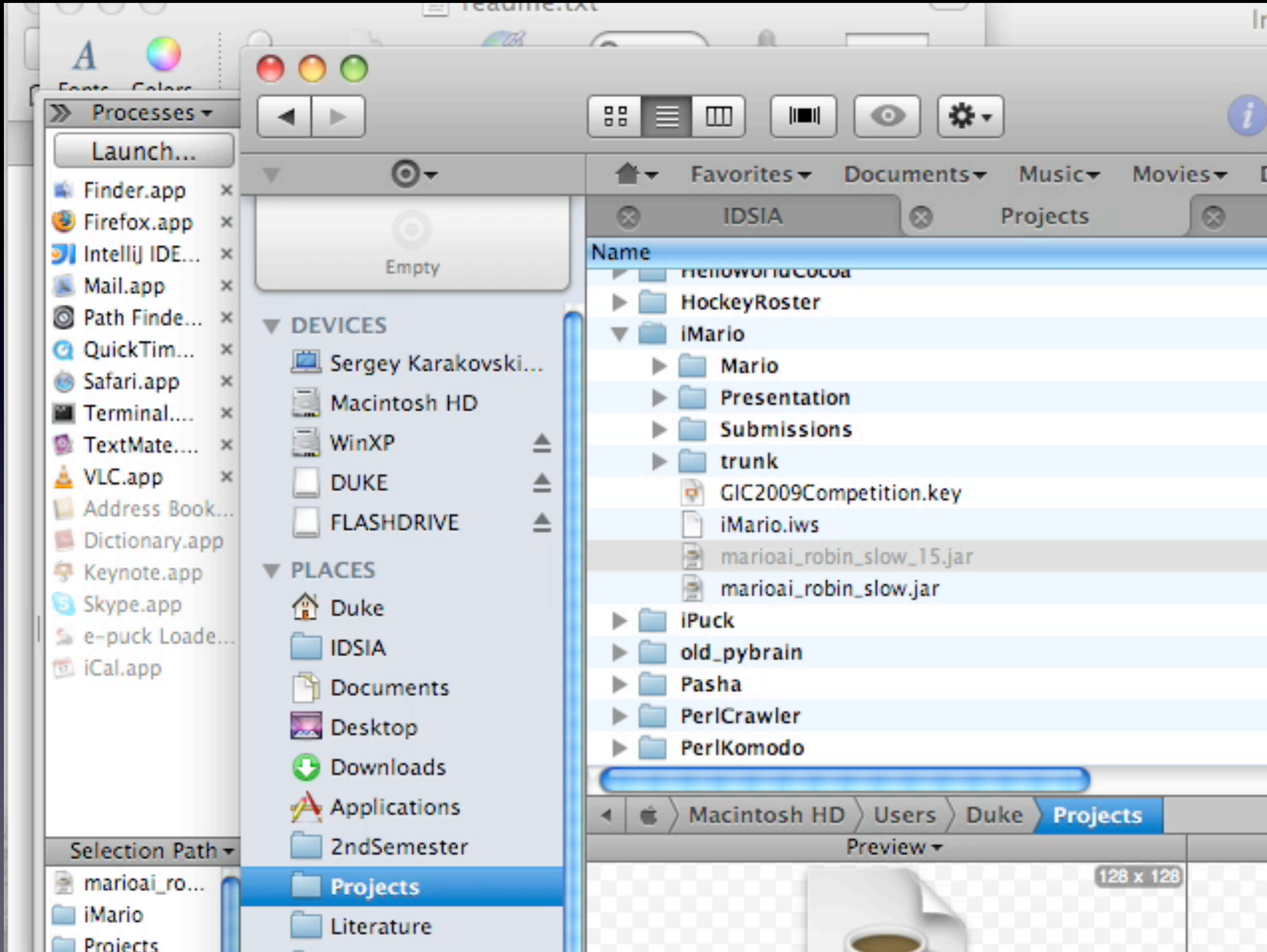
- Plan ahead for two ticks (=1/12 sec)
- Synchronise internal world-state with received enemies and object positions.

Possible Improvements:

- Keep & update old plan instead of starting from scratch each time
- Collect coins & power-ups (e.g., using a high-level planner that pans out the route between power-ups)

VIDEO


MARIO: 1024
COINS
02
DIFFICULTY
17
TIME
184
FPS: 24
WorldPause
false
Attempt: 1 of 1
AStarAgent
Selected Actions:
RIGHT


Processes
Launch...
Finder.app
Firefox.app
IntelliJ IDE...
Mail.app
Path Finde...
QuickTim...
Safari.app
Terminal....
TextMate....
VLC.app
Address Book...
Dictionary.app
Keynote.app
Skype.app
e-puck Loade...
iCal.app
Selection Path
marioai_ro...
iMario
Projects
Empty
DEVICES
Sergey Karakovski...
Macintosh HD
WinXP
DUKE
FLASHDRIVE
PLACES
Duke
IDSIA
Documents
Desktop
Downloads
Applications
2ndSemester
Projects
Literature
Favorites
Documents
Music
Movies
IDSIA
Projects
Name
HockeyRoster
iMario
Mario
Presentation
Submissions
trunk
GIC2009Competition.key
iMario.iws
marioai_robin_slow_15.jar
marioai_robin_slow.jar
iPuck
old_pybrain
Pasha
PerlCrawler
PerlKomodo
Macintosh HD
Users
Duke
Projects
Preview
128 x 128

Trond Ellingsen

- Rule based agent. Estimates the danger of a gap, enemies and tries to avoid them.

Matthew Erickson

- Genetic programming and some simple hard coded detectors.
- Nodes arithmetic if-then, detectors (e.g. closest enemy, next pit)
- Population 500 was used; 90% crossbreeding, 9% cloning and 1% mutation
- Lots of room for improvement, e.g. no detector for blocks yet.

Glenn Hartmann

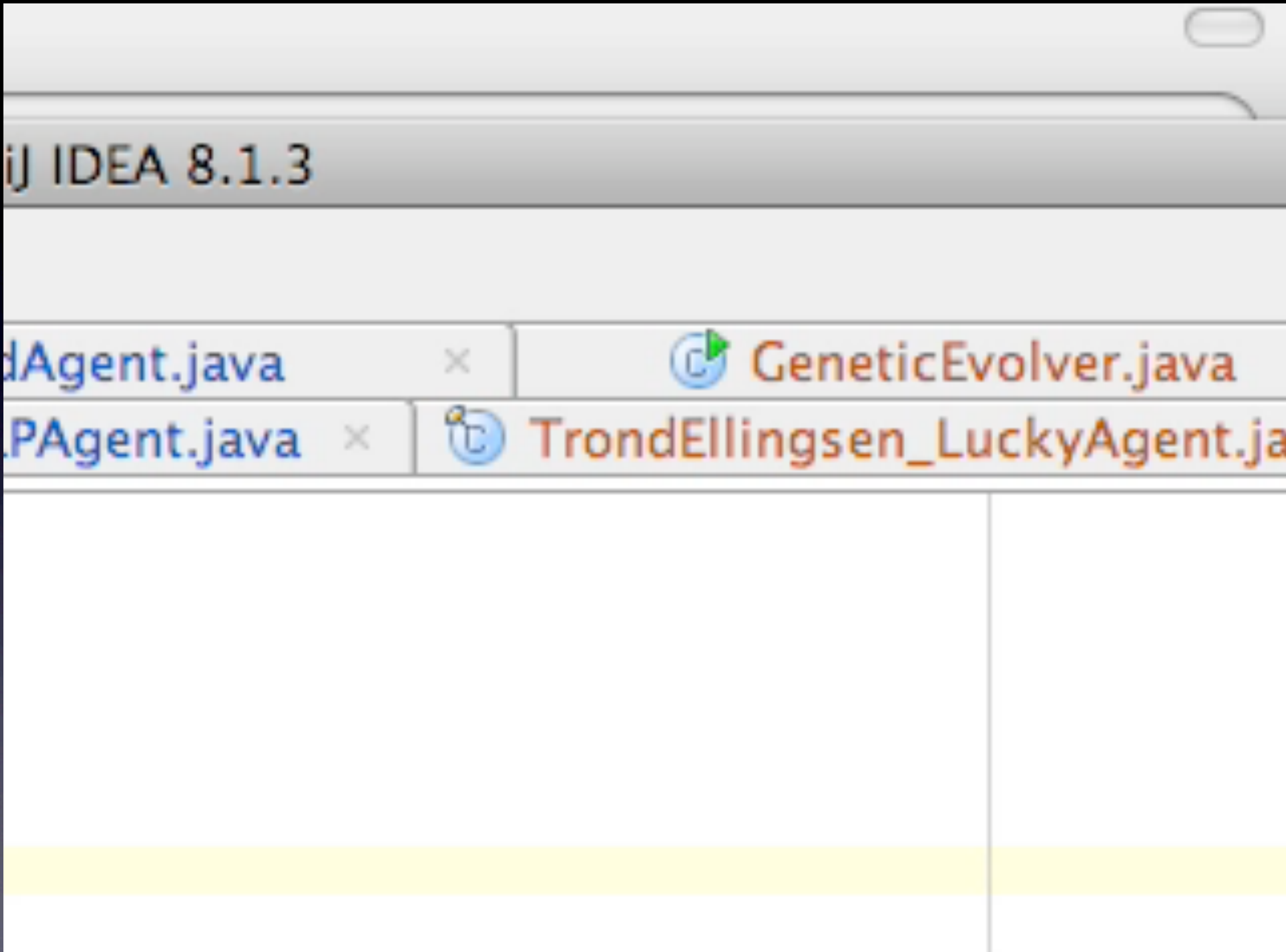
- Modified version of one of the heuristic agents that came with the software
- Move forward
- Jump if in danger of falling
- Jump over enemies if safe
- Shoot continuously

Douglas Hawkins

- Evolved using a genetic algorithm, using a simple stack-based virtual machine.

Peter Lawford

- A-star search to maximize x position
- Partial simulation to anticipate future positions (recalculated if simulation goes out of sync)
- Some pruning of search tree


iJ IDEA 8.1.3
dAgent.java
GeneticEvolver.java
PAgent.java
TrondEllingsen_LuckyAgent.ja

Sergio Lopez

- Rule-based system, to answer 2 questions: “should I jump?” and “which type of jump?”
- Evaluates possible landing points based on environment info and heuristics (no simulation)
- Calculates “danger value” for each action, and “need to jump”
- Special situations, e.g. waiting for flowers and bullets to go away, climbing “stairs”

Rafael Oliveira

- Did not submit any documentation
- Seems to be an elaborate heuristic of a reactive agent.

Michal Tuláček

- State machine with 4 states: walk_forward, walk_backward, jump, jump_hole

Mario Pérez

- Subsumption-type controller: later layers can override the action of earlier layers
- Each layer either a method or a state machine

Andy Sloane

- Joint work with Caleb Anderson and Peter Burns
- Based on A*
- Separate simulation of the game physics (not using the game engine)
- (imperfect) prediction of enemies' movements
- Working towards propagating penalties in the tree

Erek Speed

- Rule-based system
- Maps the whole observation space to the action space
  - antecedent: 22x22 array, consequent: 5 bits action
  - put in hash table
- Evolved with a GA
  - Genome as > 100 Mb XML file!

Spencer Schumann

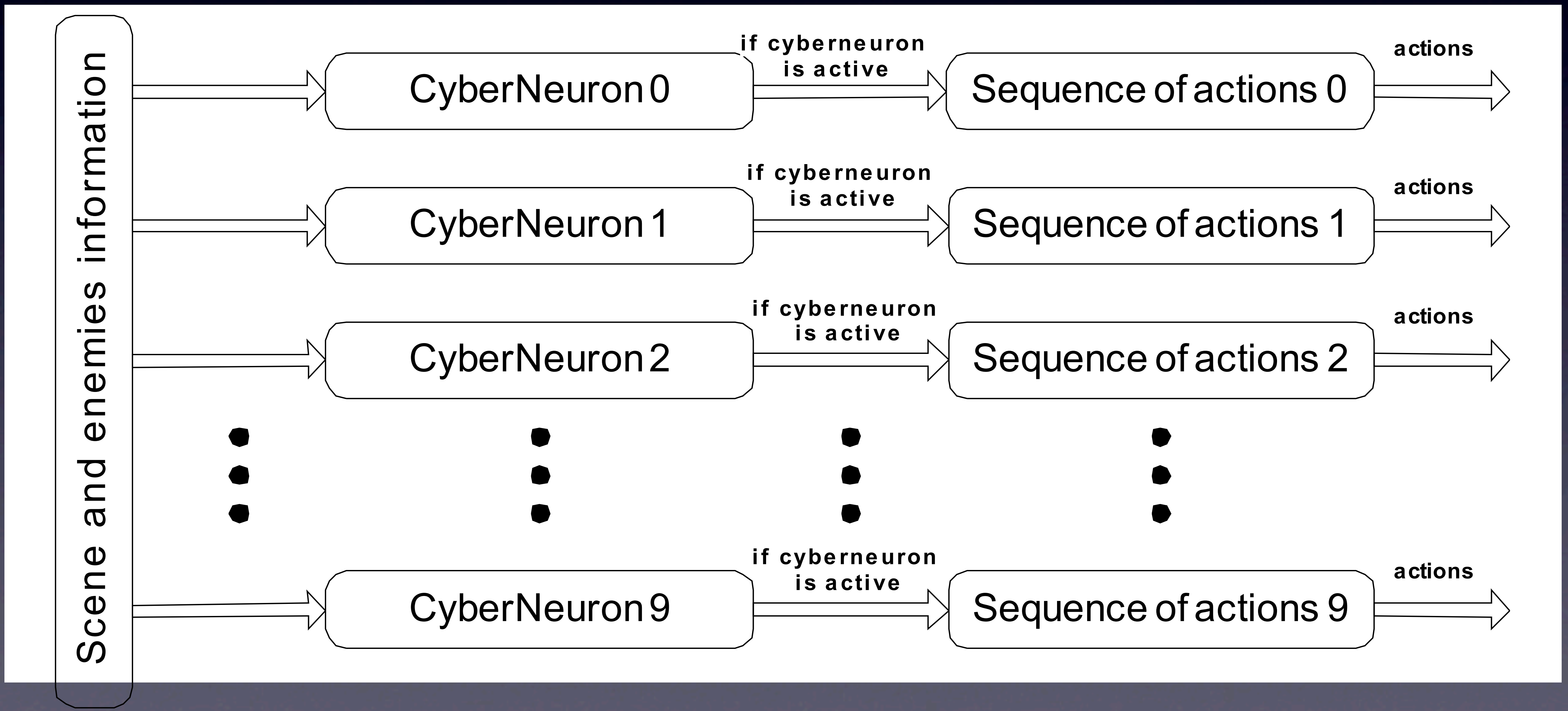
- Simulates Mario's motion
- Converts observation into a vectorized format containing walls, floors, and ceilings
- Limited search space: sorts the floors from right to left, and tries to calculate a jump
- Calculates time needed to run from the current position to left edge of target floor
- For each jump button hold time (0 – 7), calculates when to jump to land on edge

Alexandru Paler

- Trained by a human player NN that should have learned the inverse function of the Mario movement.
- The net gets as input the distance to be traveled by Mario and returns the number of presses one should use to move Mario.
- A* to find the route to the margin of the screen. After route discovery decision on where to move Mario is made.

Sergey Polikarpov

- Based on “Cyberneurons”



Results

	Name	Alg	Score	lvls	time left	kills total	mode
1	Robin Baumgarten	A*	46564.8	40	4878	373	76
2	Peter Lawford	A*	46564.8	40	4841	421	69
3	Andy Sloane	A*	44735.5	38	4822	294	67
4	Trond Ellingsen	RB	20599.2	11	5510	201	22
5	Sergio Lopez	RB	18240.3	11	5119	83	17
6	Spencer Schumann	RB, H	17010.5	8	6493	99	24
7	Matthew Erickson	Ev, GP	12676.3	7	6017	80	37
8	Douglas Hawkins	Ev, GP	12407.0	8	6190	90	32
9	Sergey Polikarpov	CN	12203.3	3	6303	67	38
10	Mario Pérez	SM, Lrs	*12060.2*	*4*	*4497*	*170*	*23*
11	Alexandru Paler	NN, A*	*7358.9*	*3*	*4401*	*69*	*43*
12	Michal Tuláček	SM	*6571.8*	*3*	*5965*	*52*	*14*
13	Rafael Oliveira	RB, H	*6314.2*	*1*	*6692*	*36*	*9*
14	Glenn Hartmann	RB, H	*1060.0*	*0*	*1134*	*8*	*71*
15	Erek Speed	GA	Out of memory				

	Name	Alg	Score	lvls	time left	kills total	mode
1	Robin Baumgarten	A*	46564.8	40	4878	373	76
2	Peter Lawford	A*	46564.8	40	4841	421	69
3	Andy Sloane	A*	44735.5	38	4822	294	67
4	Trond Ellingsen	RB	20599.2	11	5510	201	22
5	Sergio Lopez	RB	18240.3	11	5119	83	17
6	Spencer Schumann	RB, H	17010.5	8	6493	99	24
7	Matthew Erickson	Ev, GP	12676.3	7	6017	80	37
8	Douglas Hawkins	Ev, GP	12407.0	8	6190	90	32
9	Sergey Polikarpov	CN	12203.3	3	6303	67	38
10	Mario Pérez	SM, Lrs	*12060.2*	*4*	*4497*	*170*	*23*
11	Alexandru Paler	NN, A*	*7358.9*	*3*	*4401*	*69*	*43*
12	Michal Tuláček	SM	*6571.8*	*3*	*5965*	*52*	*14*
13	Rafael Oliveira	RB, H	*6314.2*	*1*	*6692*	*36*	*9*
14	Glenn Hartmann	RB, H	*1060.0*	*0*	*1134*	*8*	*71*
15	Erek Speed	GA	Out of memory				

	Name	Alg	Score	lvls	time left	kills total	mode
1	Robin Baumgarten	A*	46564.8	40	4878	373	76
2	Peter Lawford	A*	46564.8	40	4841	421	69
3	Andy Sloane	A*	44735.5	38	4822	294	67
4	Trond Ellingsen	RB	20599.2	11	5510	201	22
5	Sergio Lopez	RB	18240.3	11	5119	83	17
6	Spencer Schumann	RB, H	17010.5	8	6493	99	24
7	Matthew Erickson	Ev, GP	12676.3	7	6017	80	37
8	Douglas Hawkins	Ev, GP	12407.0	8	6190	90	32
9	Sergey Polikarpov	CN	12203.3	3	6303	67	38
10	Mario Pérez	SM, Lrs	*12060.2*	*4*	*4497*	*170*	*23*
11	Alexandru Paler	NN, A*	*7358.9*	*3*	*4401*	*69*	*43*
12	Michal Tuláček	SM	*6571.8*	*3*	*5965*	*52*	*14*
13	Rafael Oliveira	RB, H	*6314.2*	*1*	*6692*	*36*	*9*
14	Glenn Hartmann	RB, H	*1060.0*	*0*	*1134*	*8*	*71*
15	Erek Speed	GA	Out of memory				

	Name	Alg	Score	lvls	time left	kills total	mode
1	Robin Baumgarten	A*	46564.8	40	4878	373	76
2	Peter Lawford	A*	46564.8	40	4841	421	69
3	Andy Sloane	A*	44735.5	38	4822	294	67
4	Trond Ellingsen	RB	20599.2	11	5510	201	22
5	Sergio Lopez	RB	18240.3	11	5119	83	17
6	Spencer Schumann	RB, H	17010.5	8	6493	99	24
7	Matthew Erickson	Ev, GP	12676.3	7	6017	80	37
8	Douglas Hawkins	Ev, GP	12407.0	8	6190	90	32
9	Sergey Polikarpov	CN	12203.3	3	6303	67	38
10	Mario Pérez	SM, Lrs	*12060.2*	*4*	*4497*	*170*	*23*
11	Alexandru Paler	NN, A*	*7358.9*	*3*	*4401*	*69*	*43*
12	Michal Tuláček	SM	*6571.8*	*3*	*5965*	*52*	*14*
13	Rafael Oliveira	RB, H	*6314.2*	*1*	*6692*	*36*	*9*
14	Glenn Hartmann	RB, H	*1060.0*	*0*	*1134*	*8*	*71*
15	Erek Speed	GA	Out of memory				

Observations

- The best-performing agents take much longer time per time step (frame)
- This is due to usage of A* search!
  - ...works well because of completely observable states and lack of dead ends
- But some heuristic controllers do very well
- Not many learning/optimization techniques (though many competitors claim to be working on it)

After the competition

- Competition web page will remain, complete with competition software
    - ...which you can use in your teaching or research!
- Complete source code of all submitted controllers

The future of the Mario Competition

- Mario AI Championship 2010
- Run at 2 to 4 different conferences, including EvoStar and CIG
- New physics: levels with water?
- More than one track, ideas include:
  - Standard track with more evil levels
  - Online learning of unseen level track
  - Personalized level generation track
  - (your ideas are welcome)
- Should let learning algorithms be more competitive.