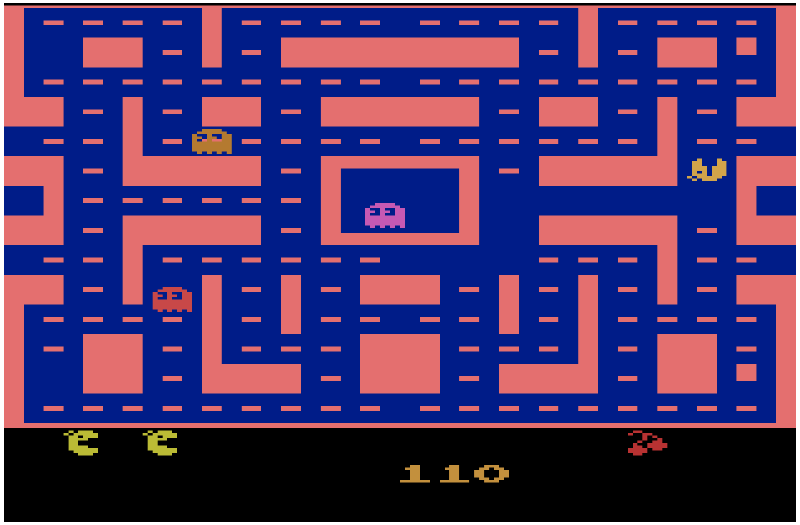
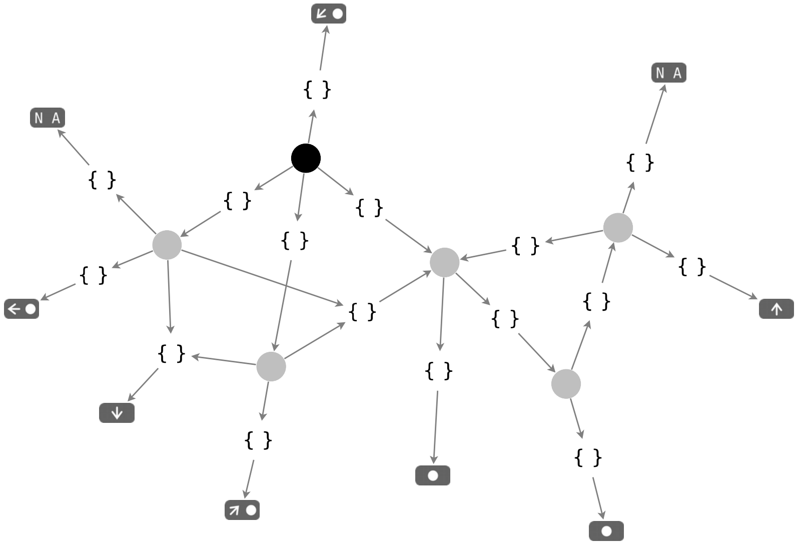
Emergent Tangled Graph Representations for Atari Game Playing Agents

Stephen Kelly and Malcolm I. Heywood
Dalhousie University, NS, Canada

Overview





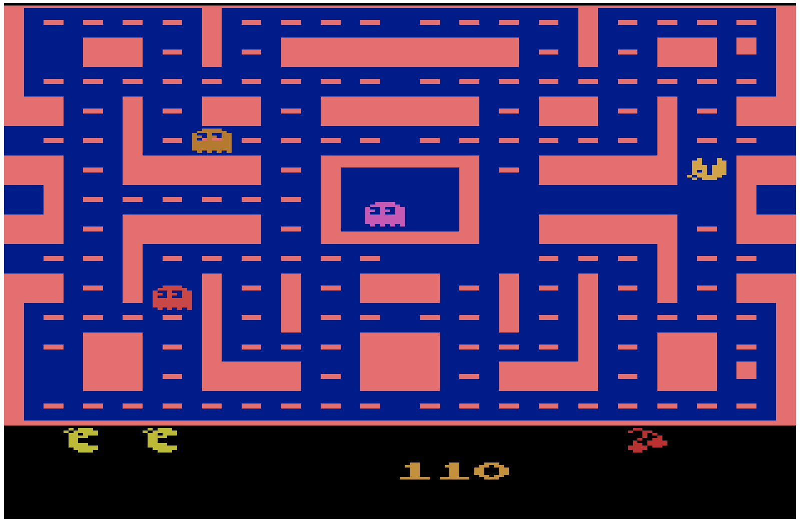
Atari Video Games

- high-dimensional
- partially observable, stochastic
- delayed rewards

Emergent Tangled Program Graphs (TPG)

- emergent modularity, open-ended evolution
- solution complexity scales through interaction with environment
- agent behaviours competitive with deep learning while being significantly simpler

Atari 2600 Video Games



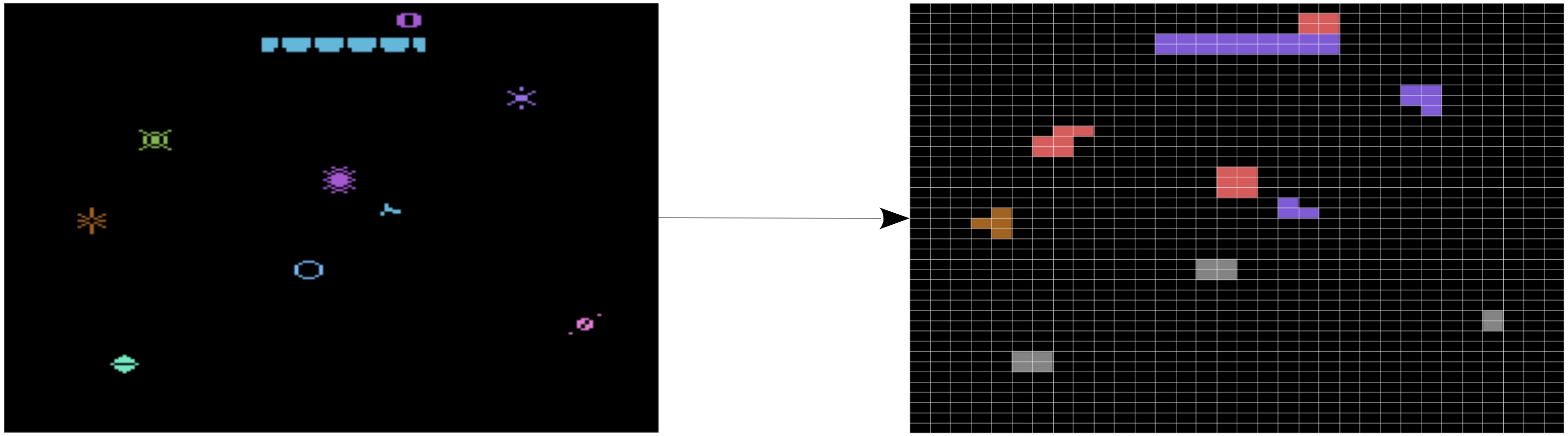
Hundreds of game titles

Humans and artificial agents use the same game interface:

- High-dimensional input space: Screen as Pixel Matrix, updated at 60Hz
- 18 actions (Joystick Positions):

↑ ↗ ● N A

Atari 2600 Screen Preprocessing



Raw Game Screen

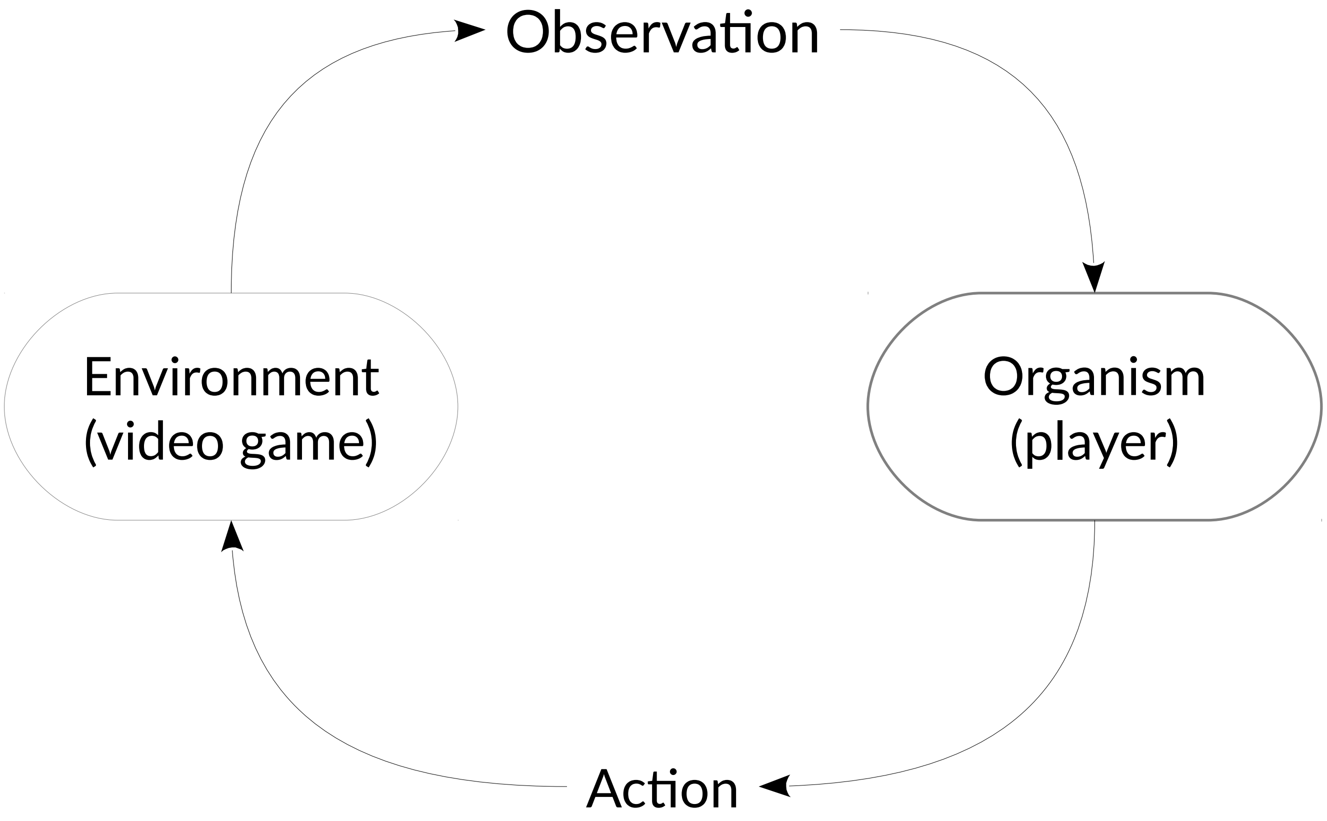
Decimal Feature Grid

210 x 160 pixels
128 colours / pixel

42 x 32 bytes
=
1344 decimal values {0-255}

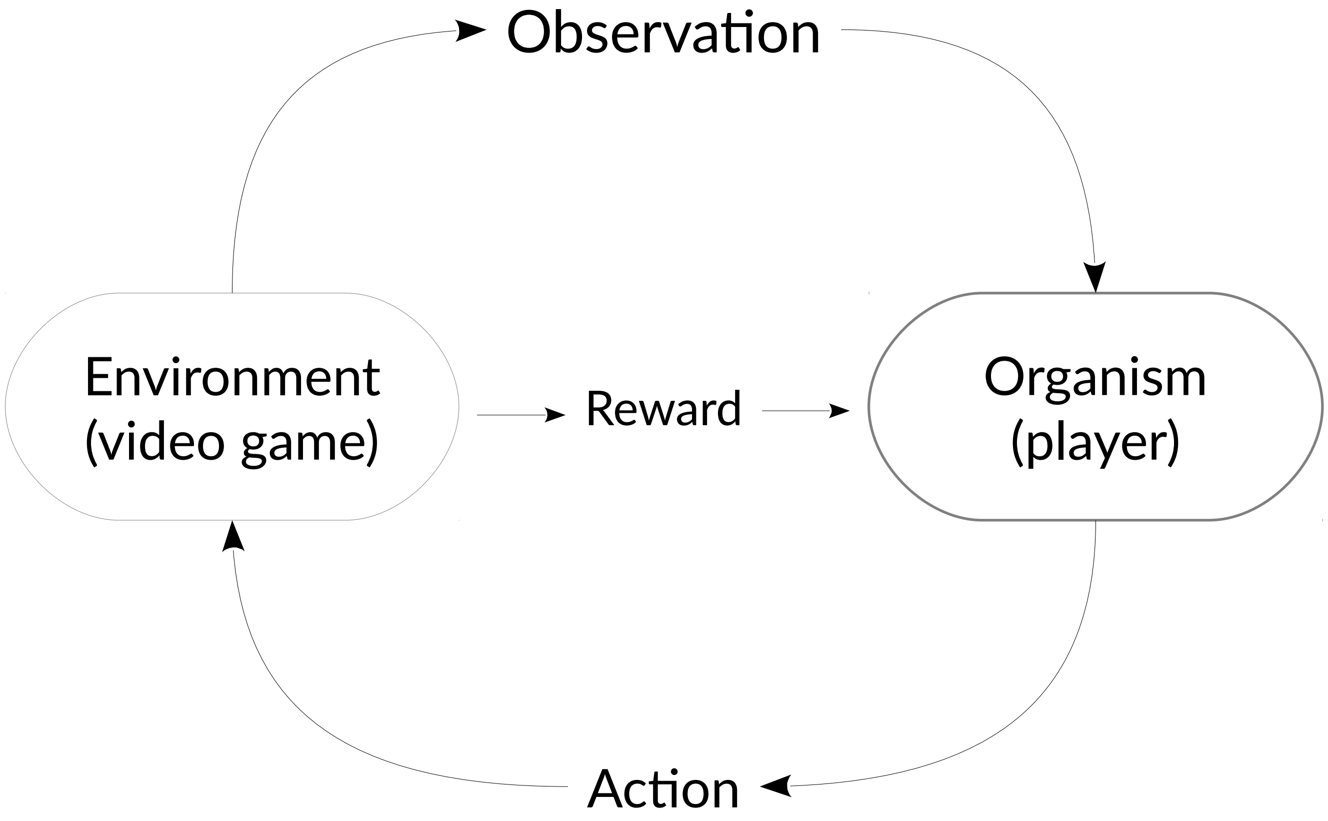
- Game entities ‘flicker’ over sequential frames, implies state is partially observable

Emergent Behaviour



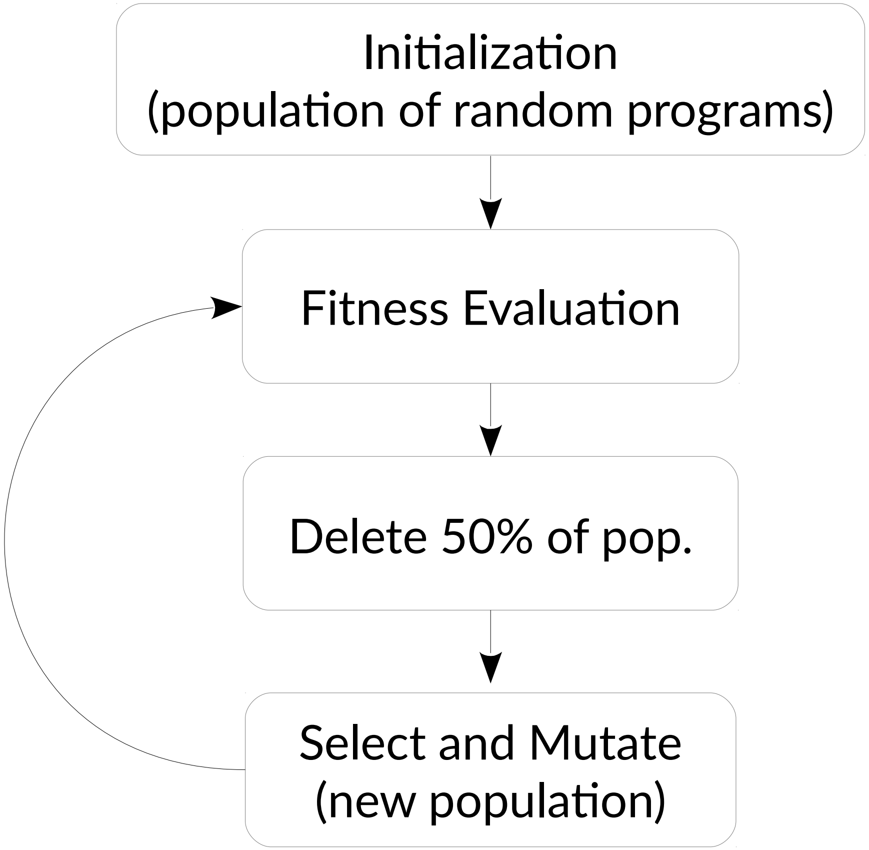
- Organism adapts through interaction with environment

Emergent Behaviour



- Reward (game score) only informative after many interactions
- Organism’s objective: maximize <u>long-term</u> reward

Genetic Programming (GP)



Initialization
(population of random programs)

Fitness Evaluation

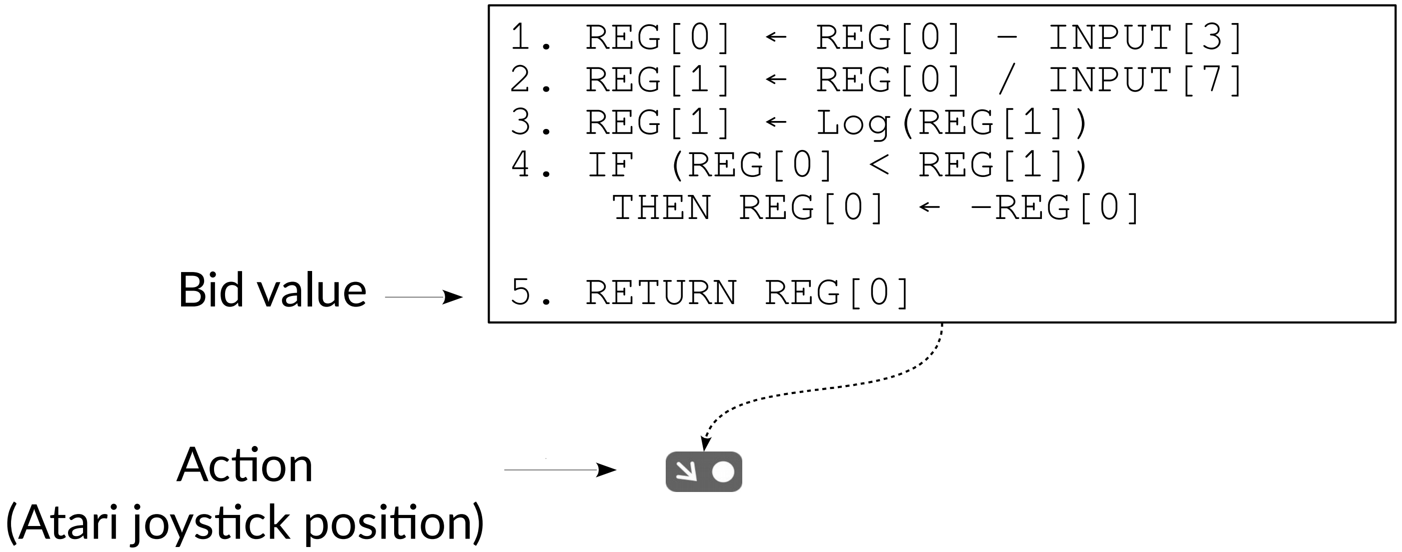
Delete 50% of pop.

Select and Mutate
(new population)

Teams of Programs

Symbiotic Bid-Based (SBB) Framework (Lichodzijewski, 2011)

Program

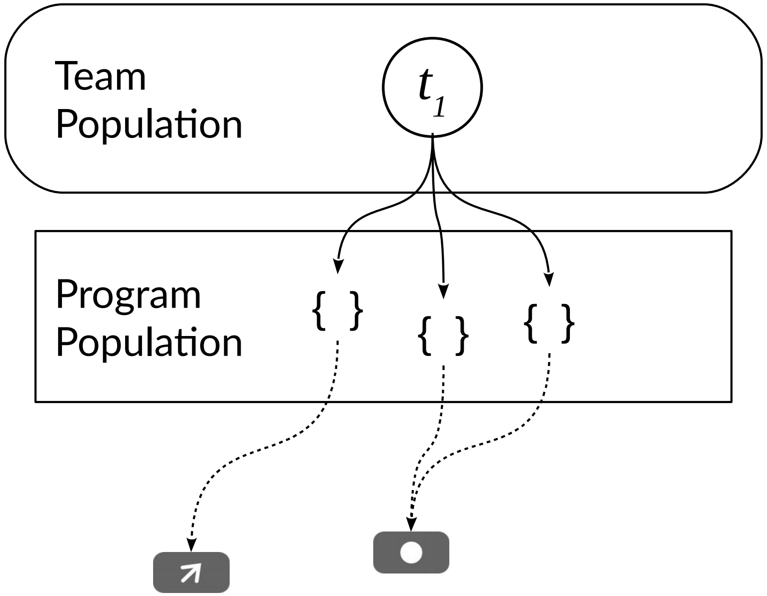


```
REG[0] ← REG[0] - INPUT[3]
REG[1] ← REG[0] / INPUT[7]
REG[1] ← Log(REG[1])
IF (REG[0] < REG[1])
 THEN REG[0] ← -REG[0]

RETURN REG[0]
```

- defines context for <u>one</u> action

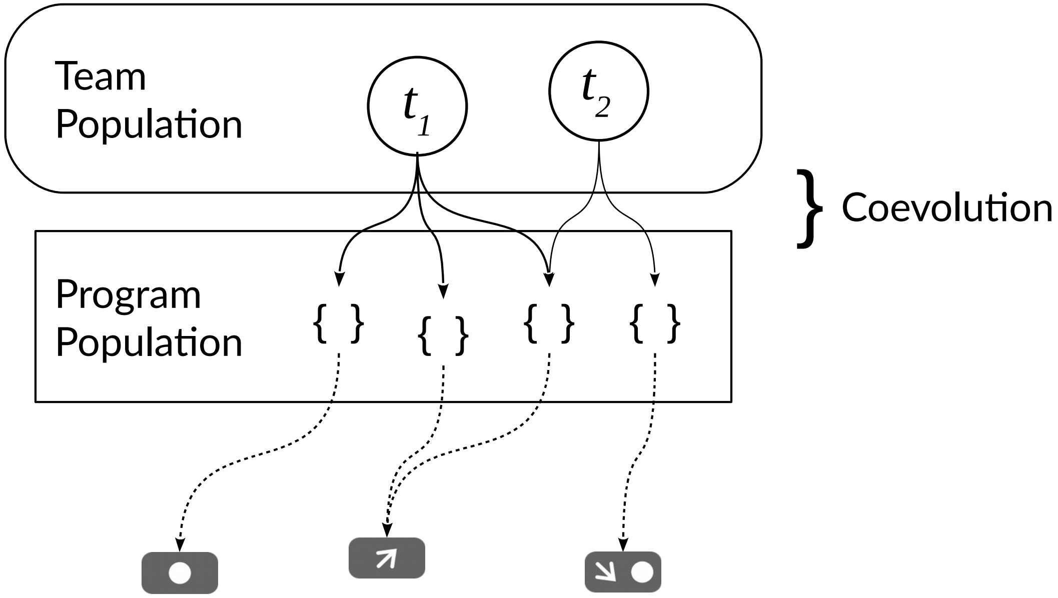
Teams of Programs



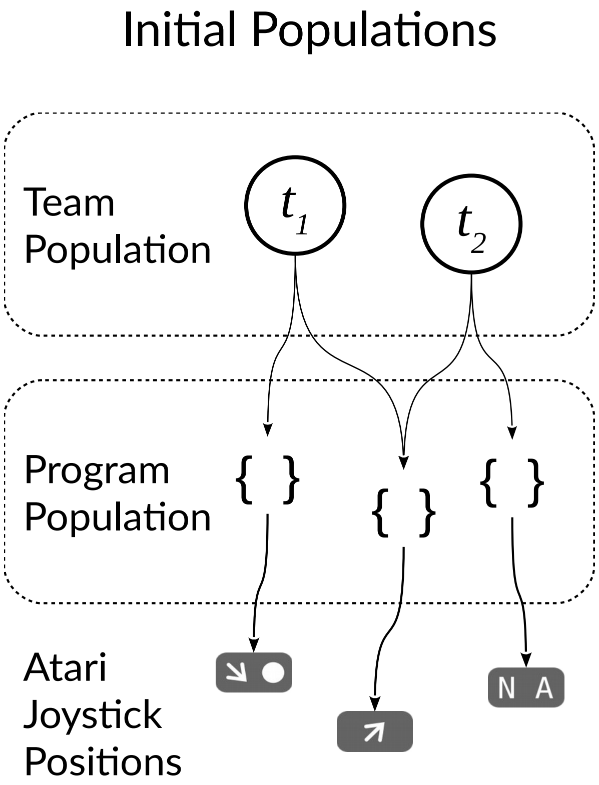
- team represents complete decision-making policy
- program with highest bid at current time runs action

Teams of Programs

- Team and Program populations coevolved
- Fitness assigned to teams only
- Fixed number of teams are deleted/introduced each generation

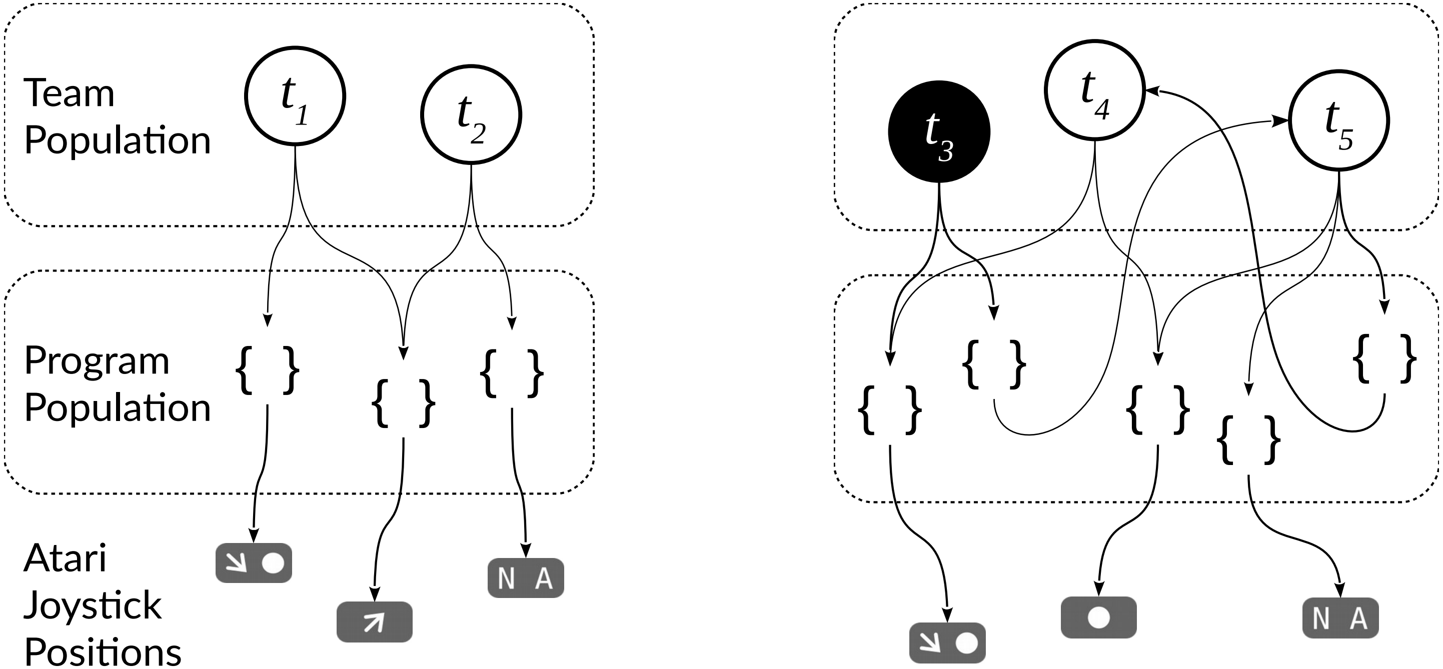


Tangled Program Graphs (TPG)



- Single team of programs represents smallest stand-alone decision-making entity (module)

Tangled Program Graphs (TPG)

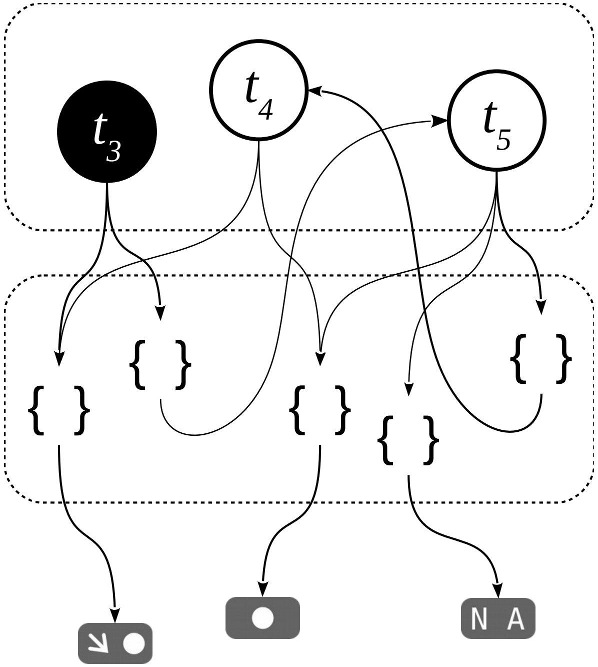


Initial Populations

Evolved Populations

- Multi-team *policy graphs* emerge
- Decision-making begins at root team (t_3)

Tangled Program Graphs (TPG)

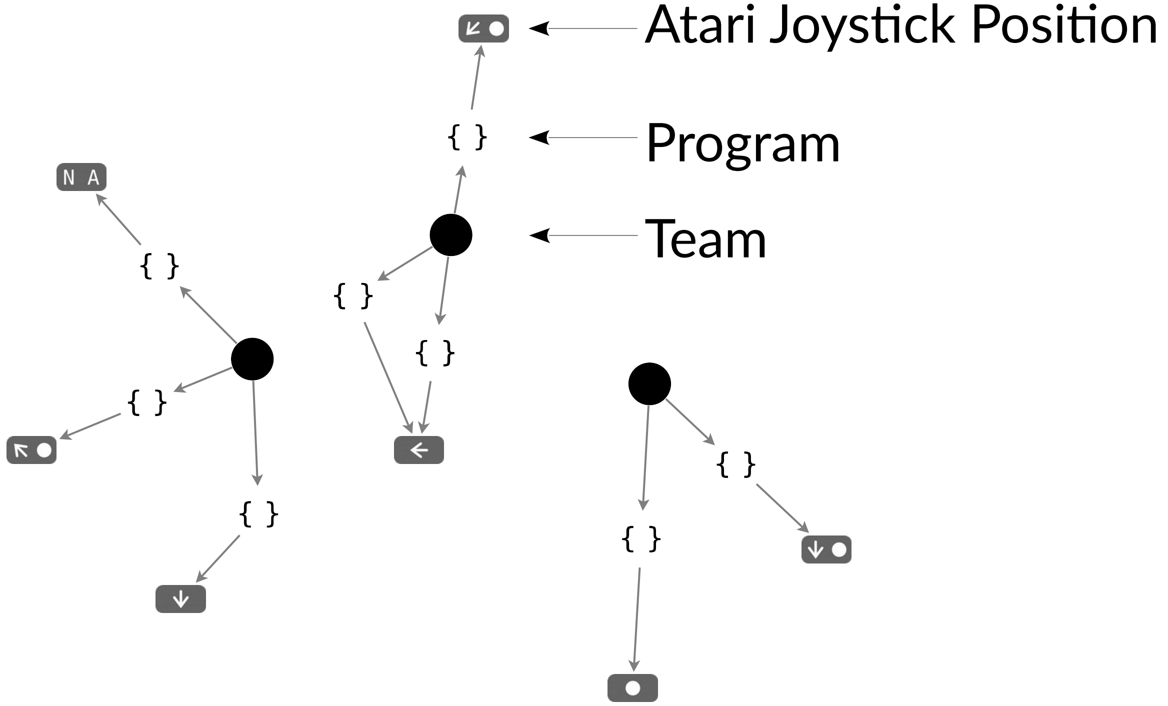


Only *root* teams (t_3) have fitness evaluated and are modified by variation operators:

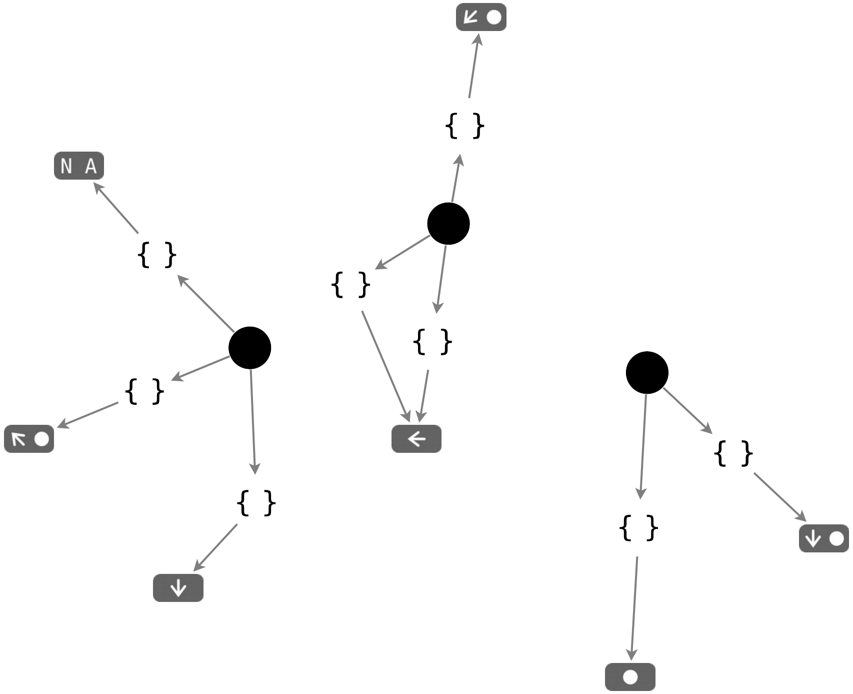
- Manageable search space
- Incremental development of policies (protects 'lower-level' complex structures)

Tangled Program Graphs (TPG)

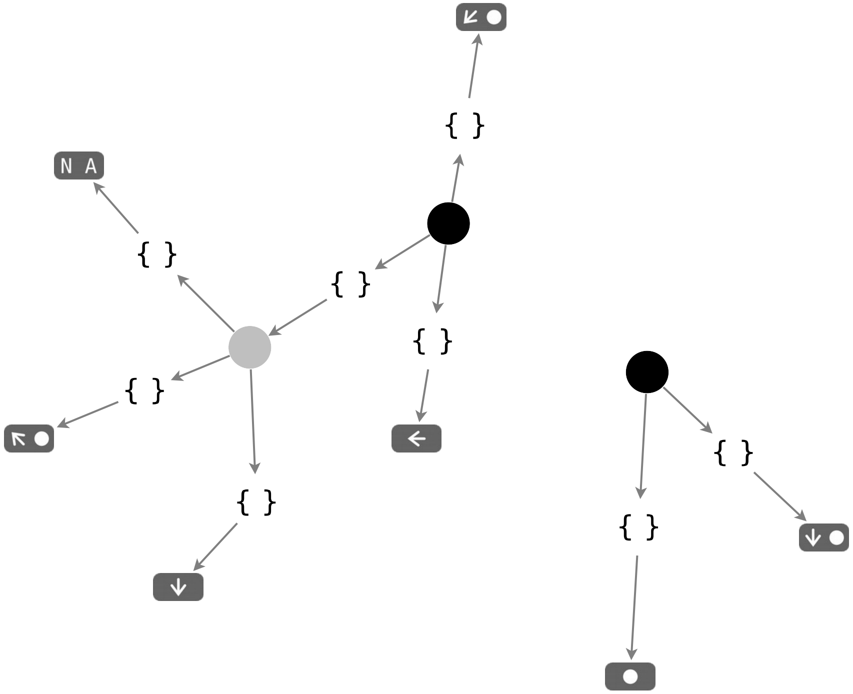
Initial Policies



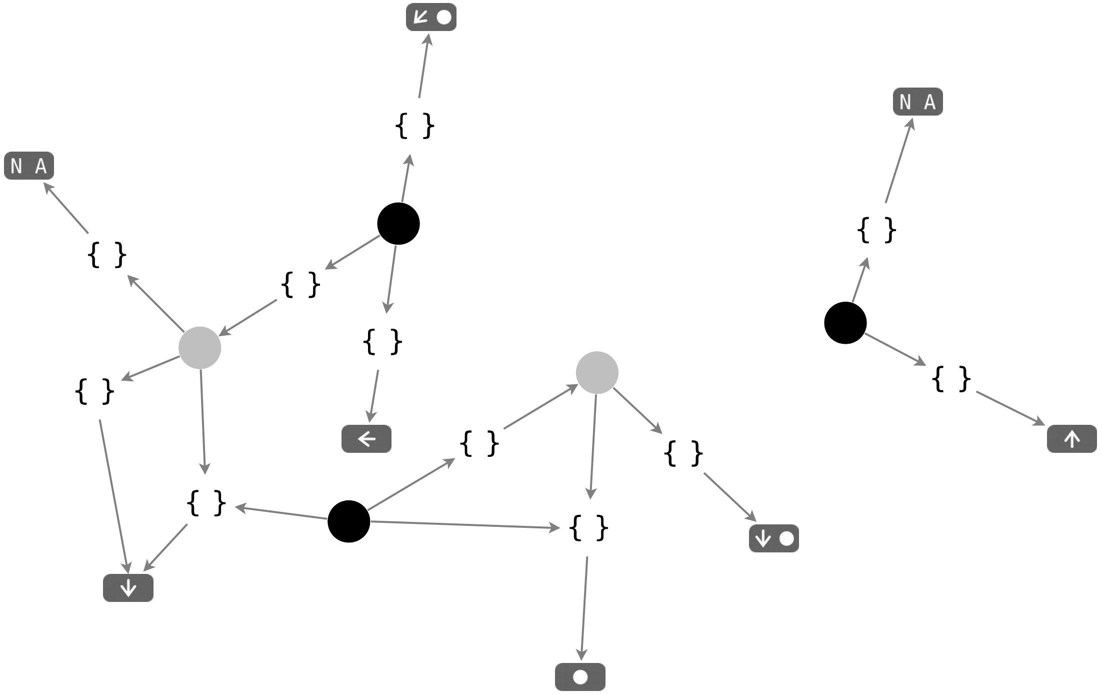
Tangled Program Graphs: Development



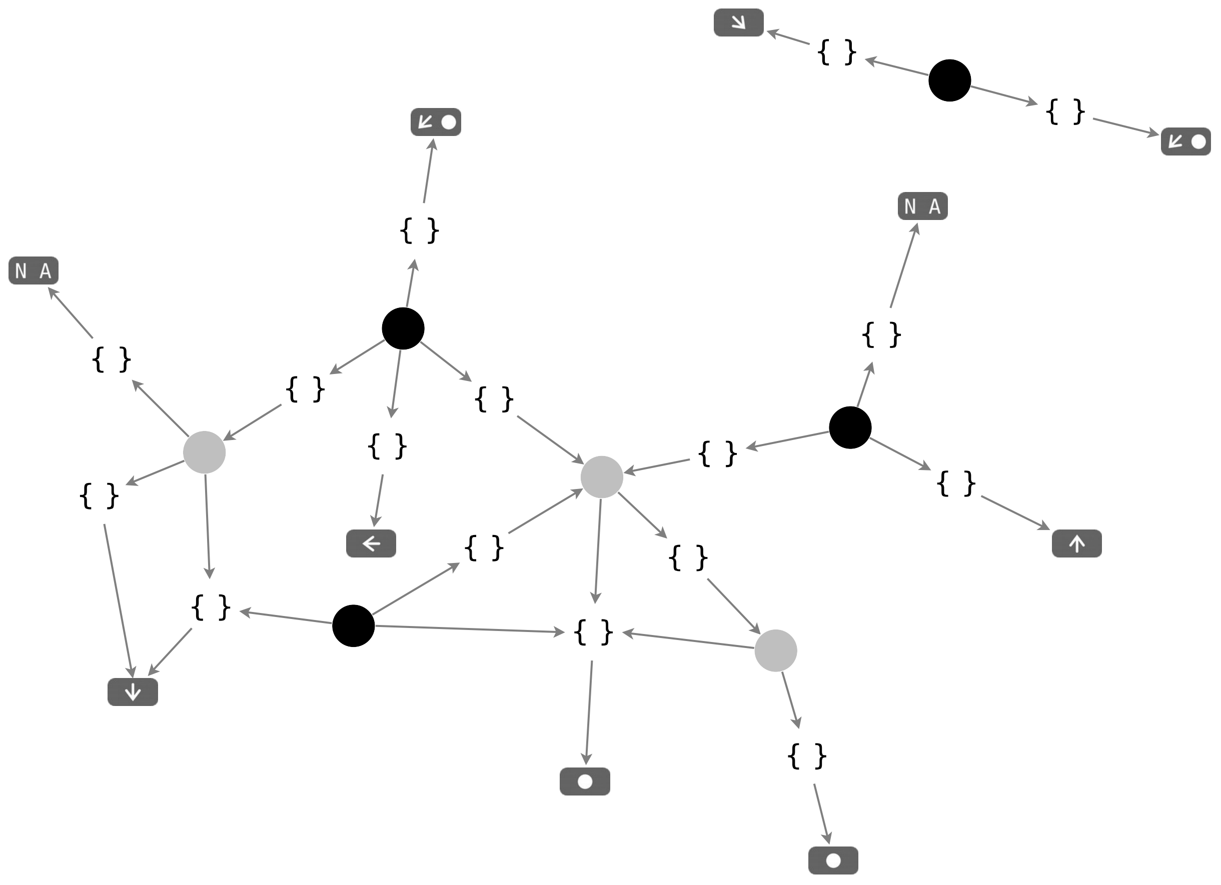
Tangled Program Graphs: Development



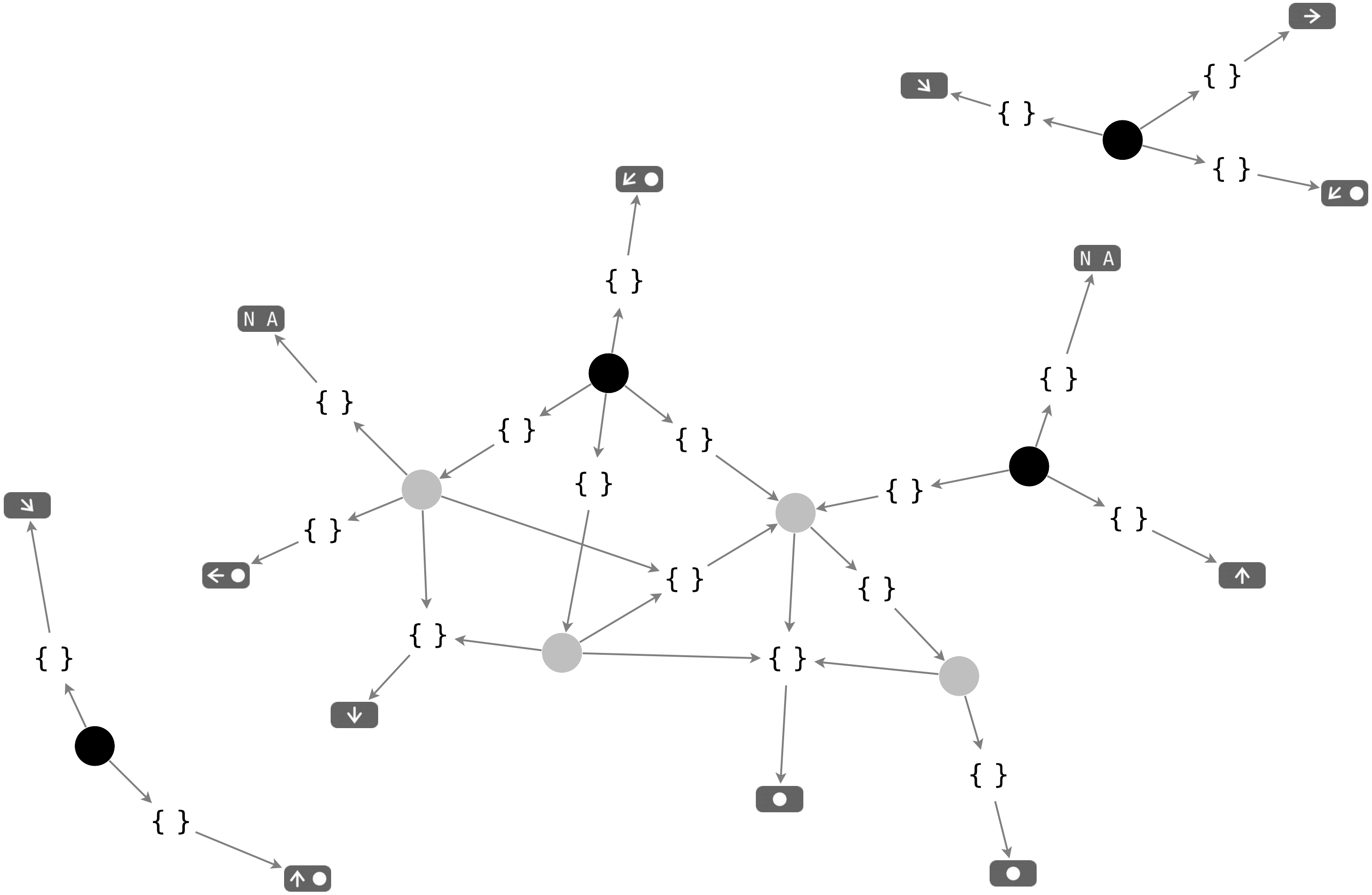
Tangled Program Graphs: Development



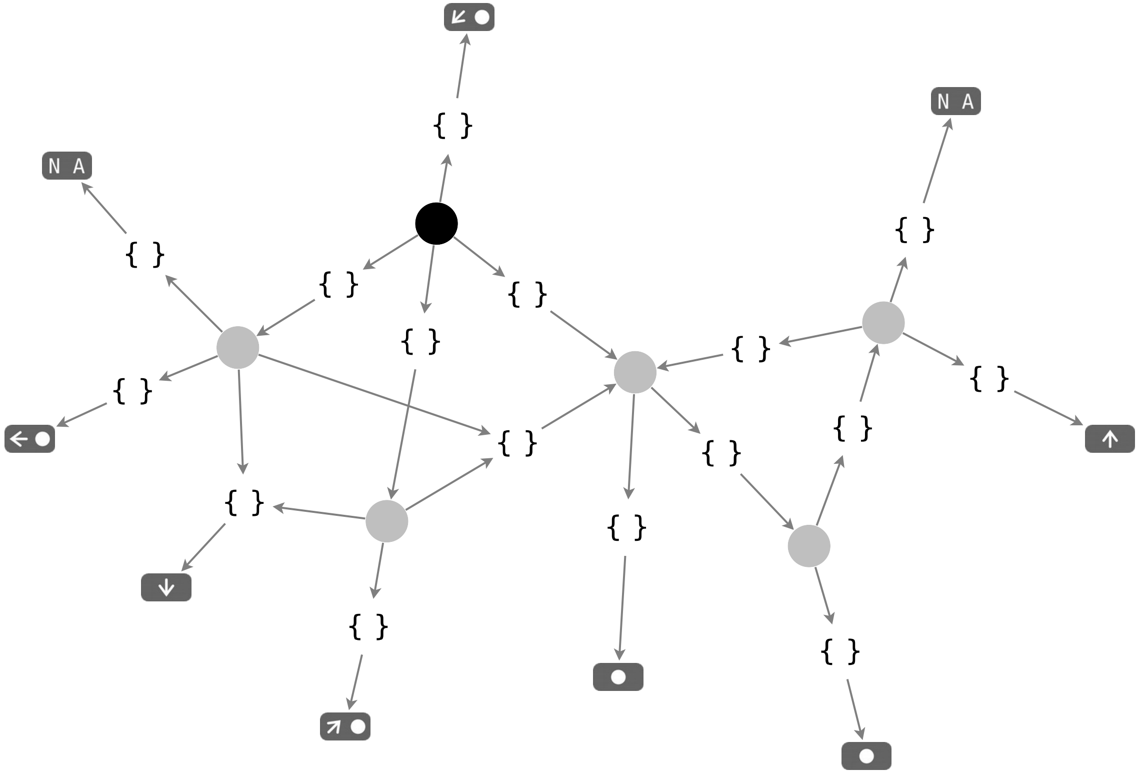
Tangled Program Graphs: Development



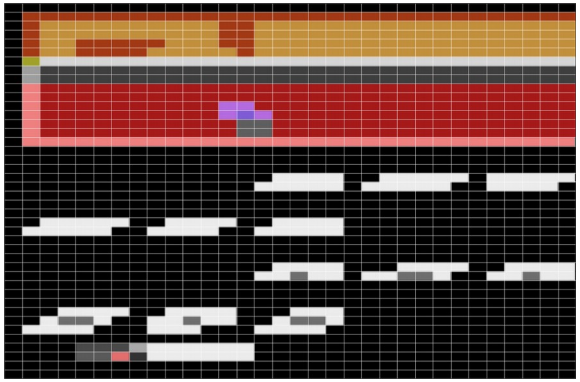
Tangled Program Graphs: Development

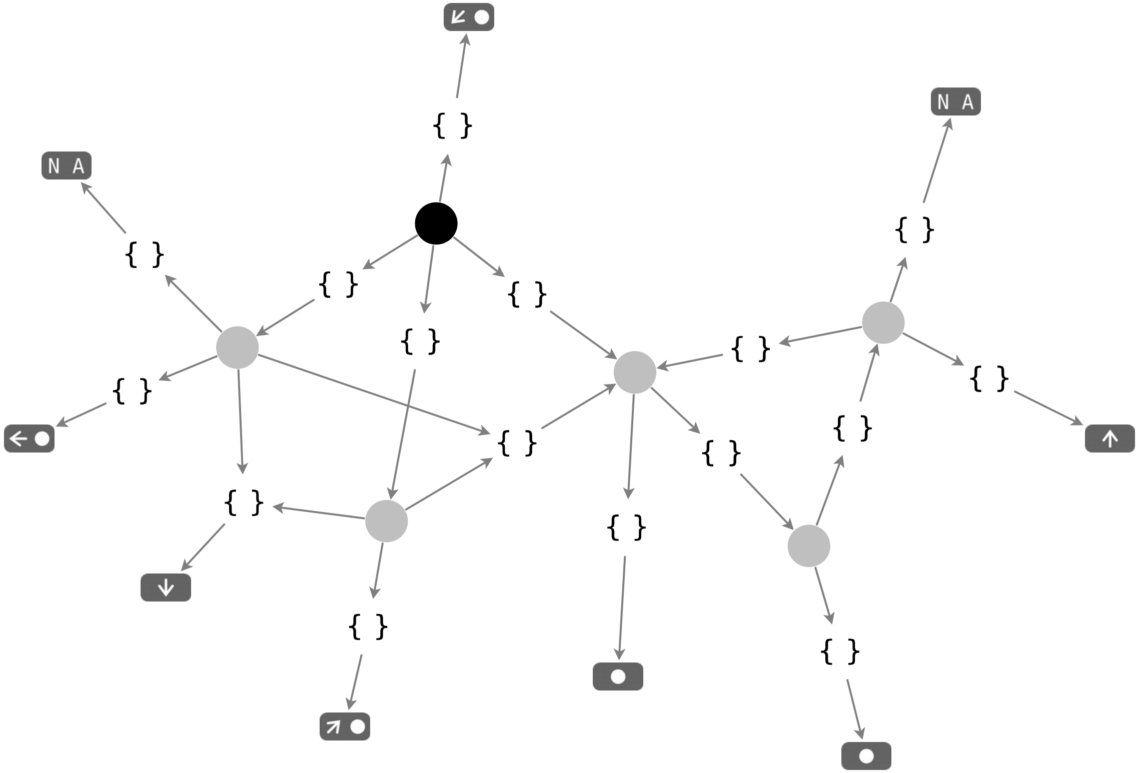


Tangled Program Graphs: Development

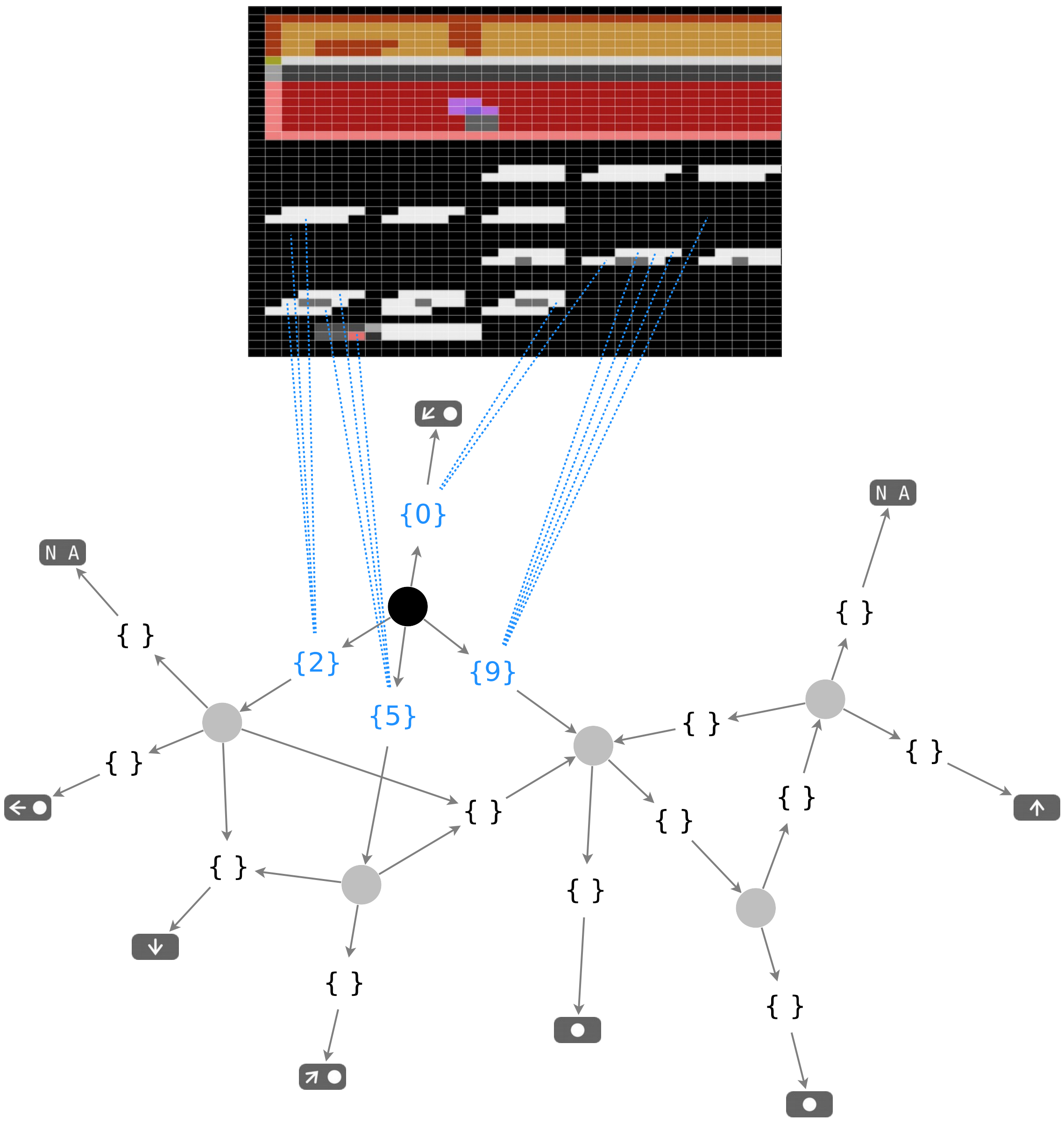


Tangled Program Graphs: Decision Making

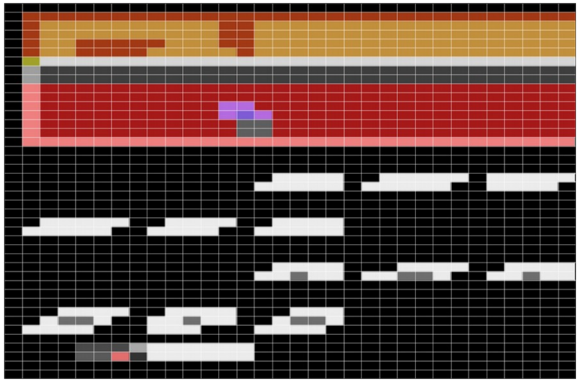


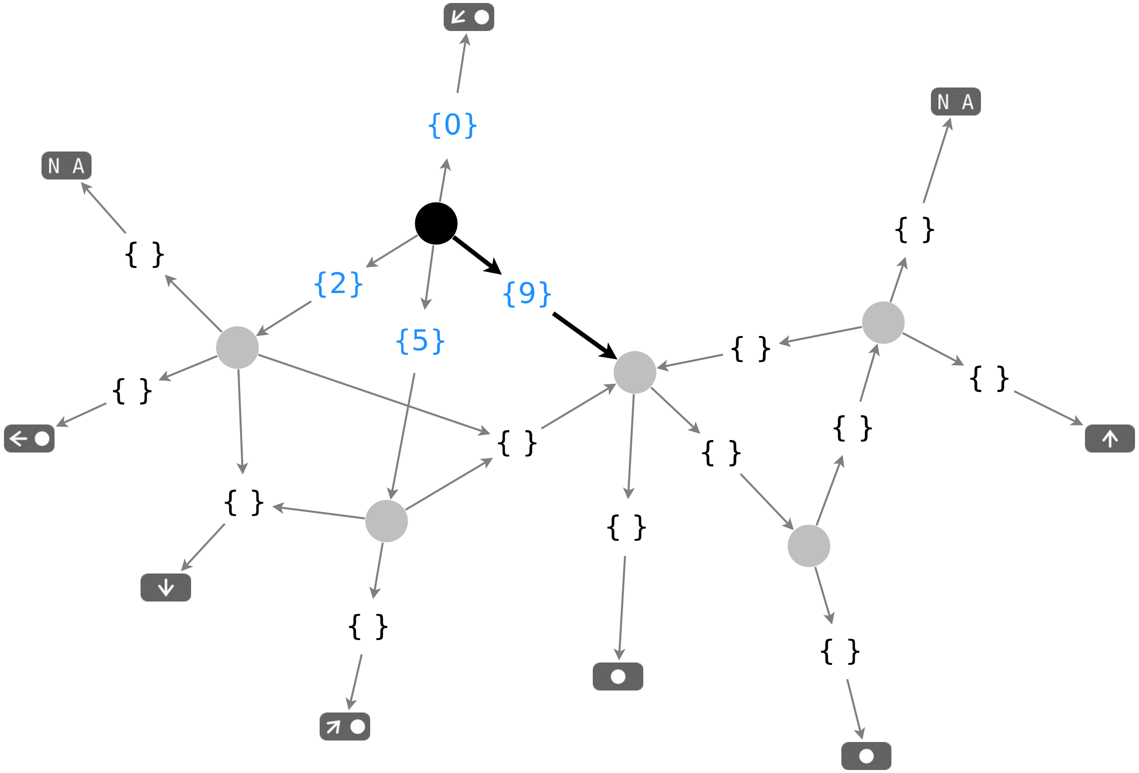


Tangled Program Graphs: Decision Making

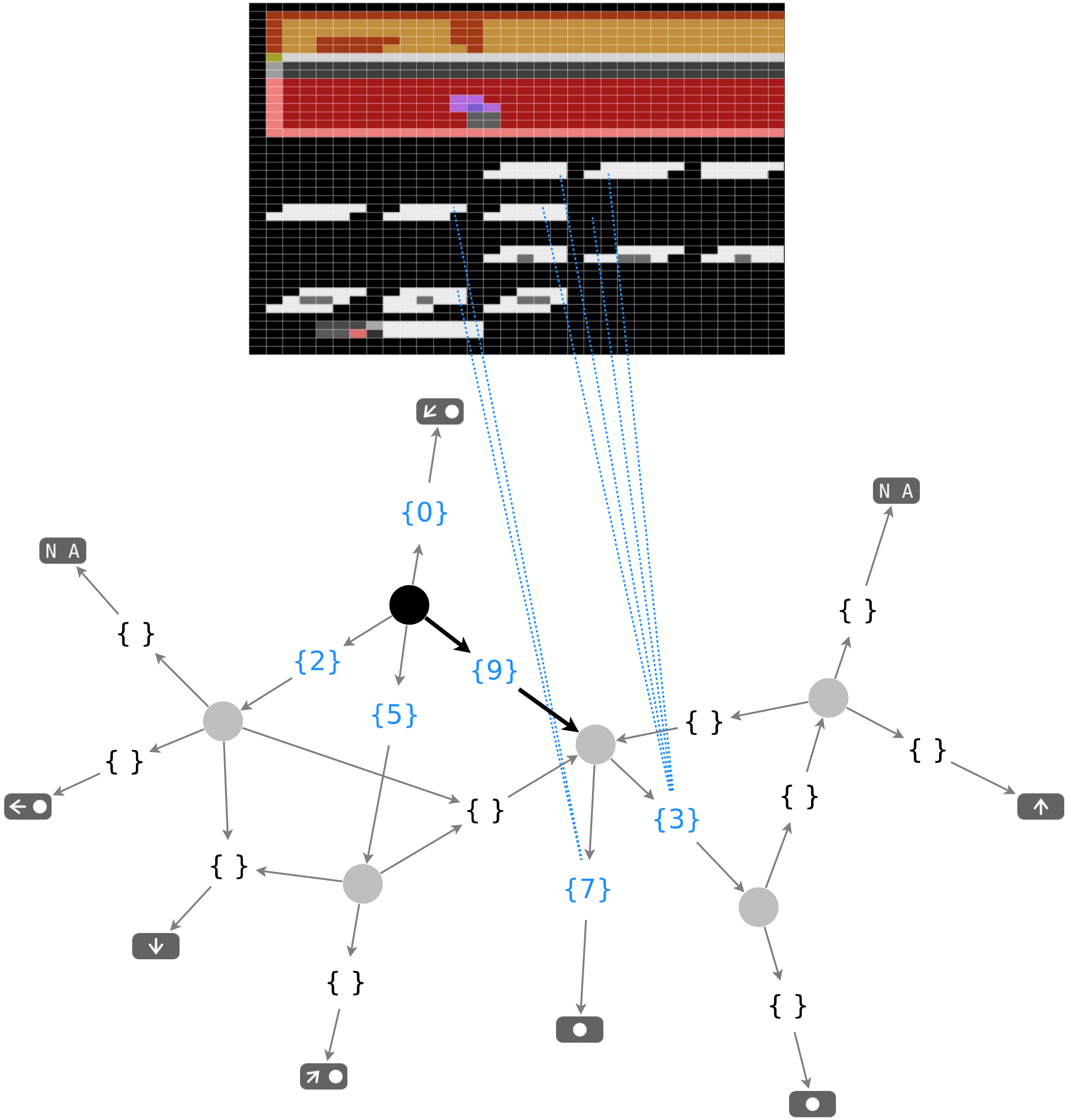


Tangled Program Graphs: Decision Making

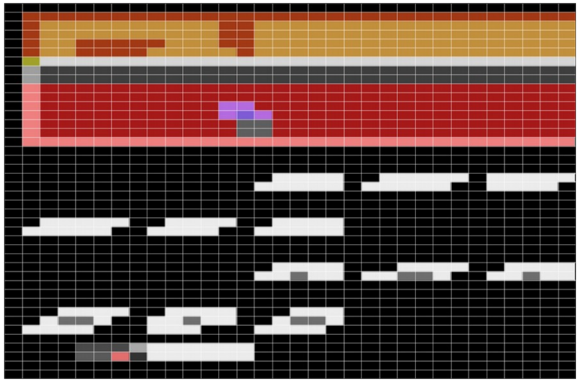




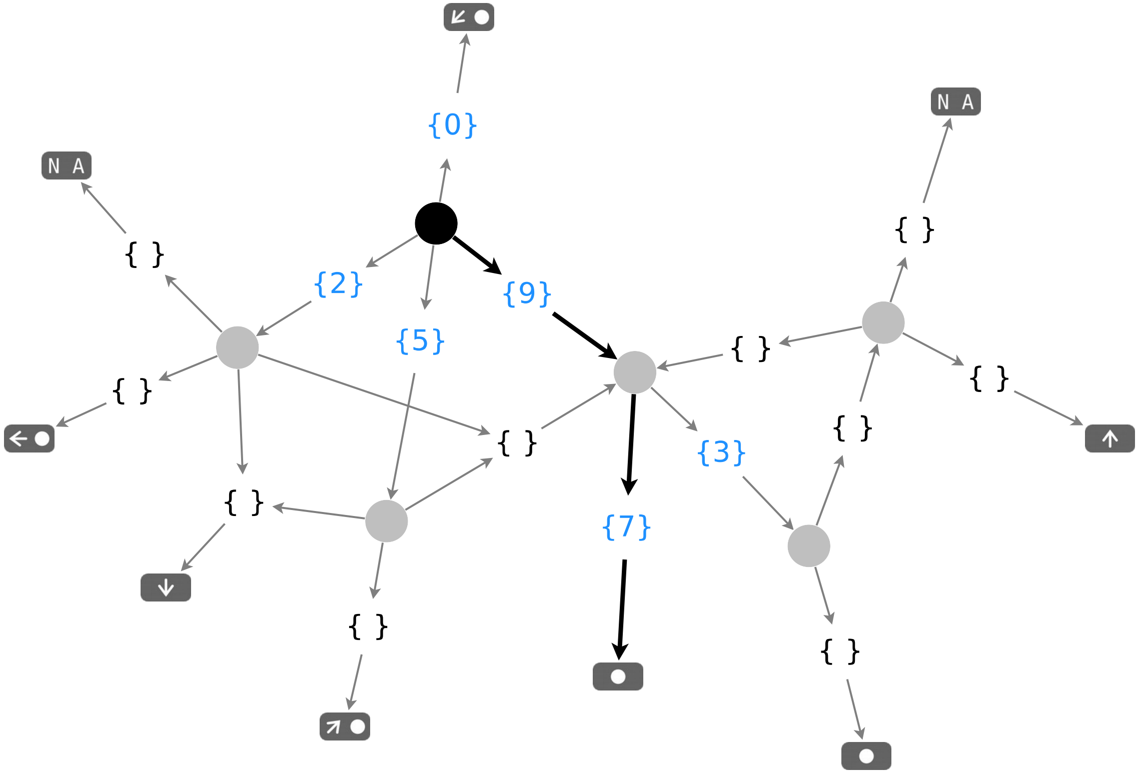
Tangled Program Graphs: Decision Making



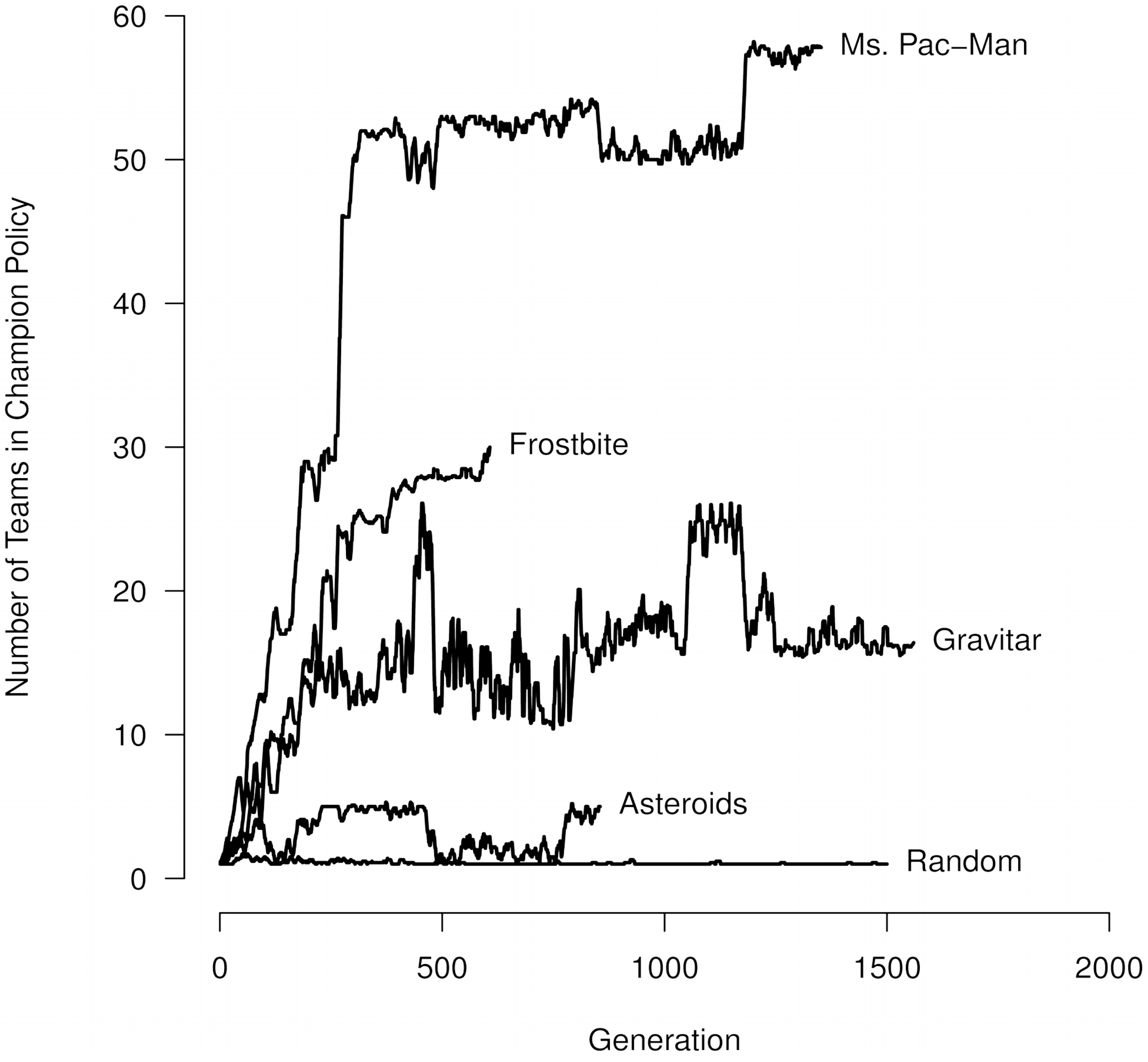
Tangled Program Graphs: Decision Making



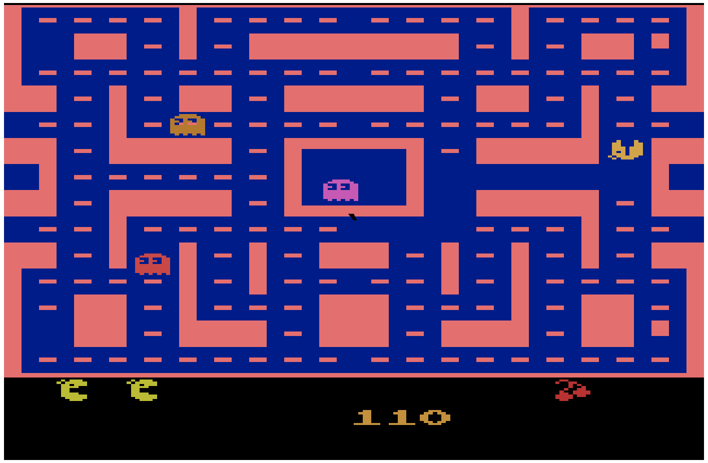
- One root→leaf path for each decision



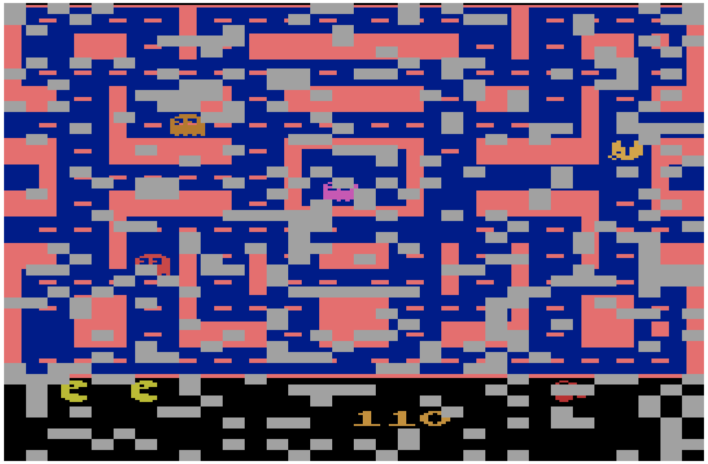
Emergent Modularity



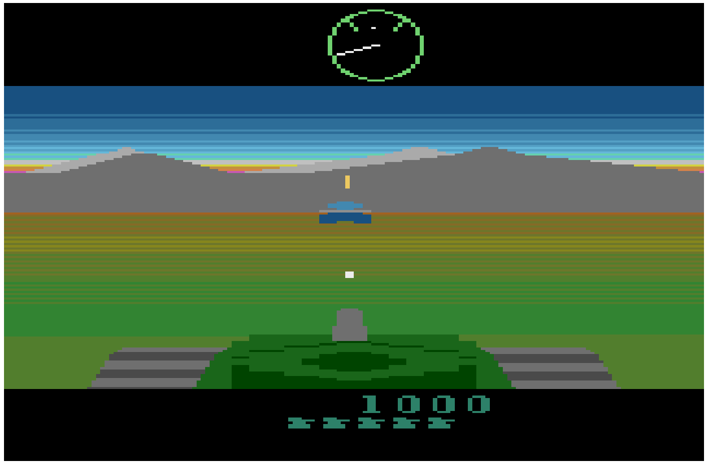
Adapted Visual Field in TPG



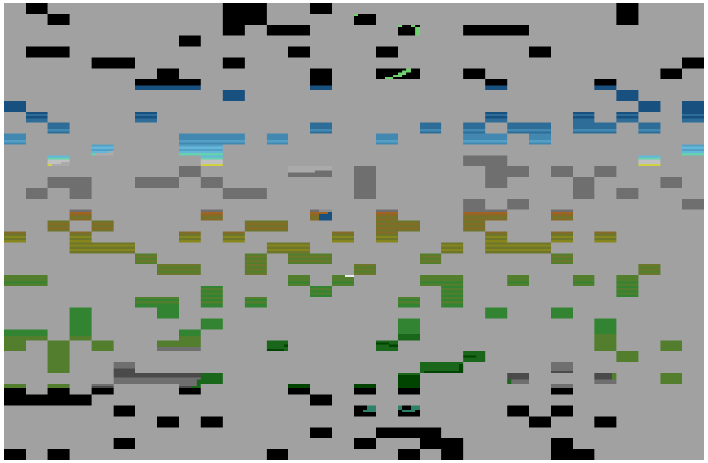
Ms. Pac-Man Screen



Ms. Pac-Man AVF

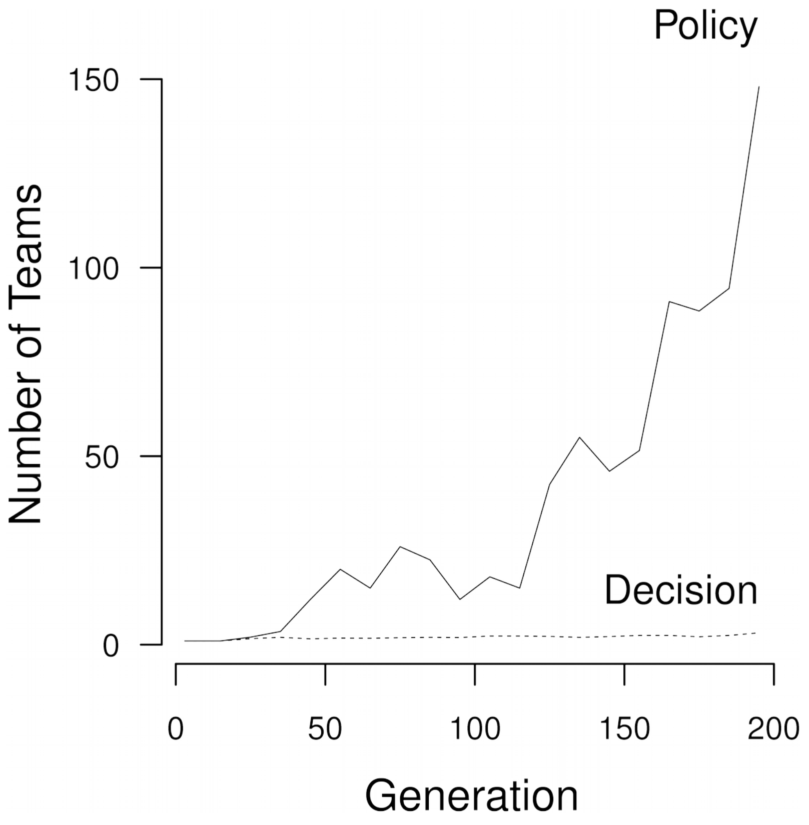


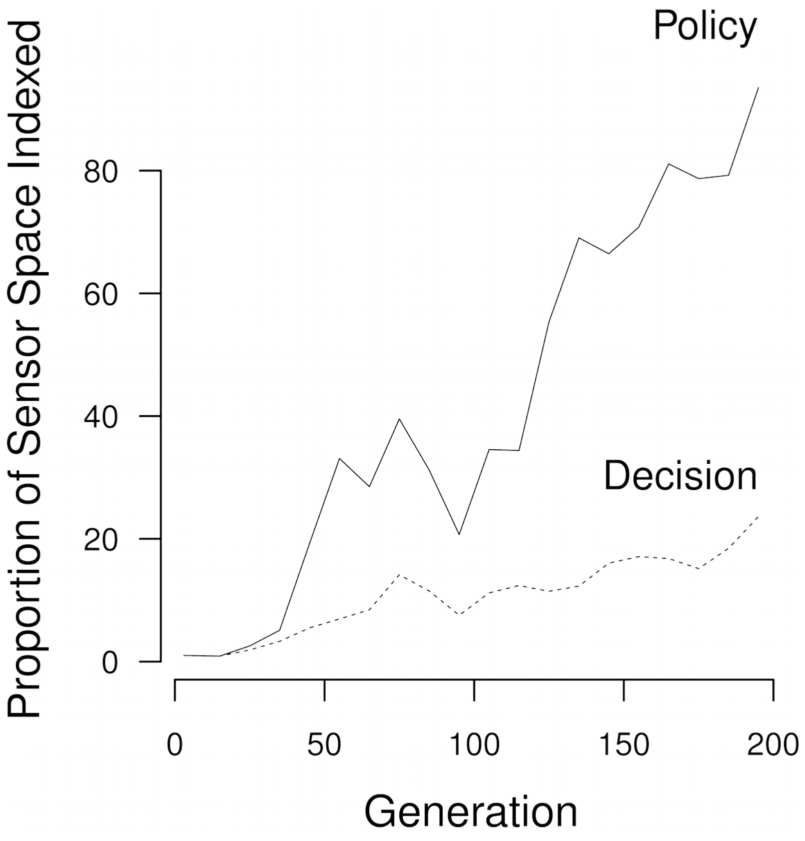
Battle Zone Screen



Battle Zone AVF

Complexity

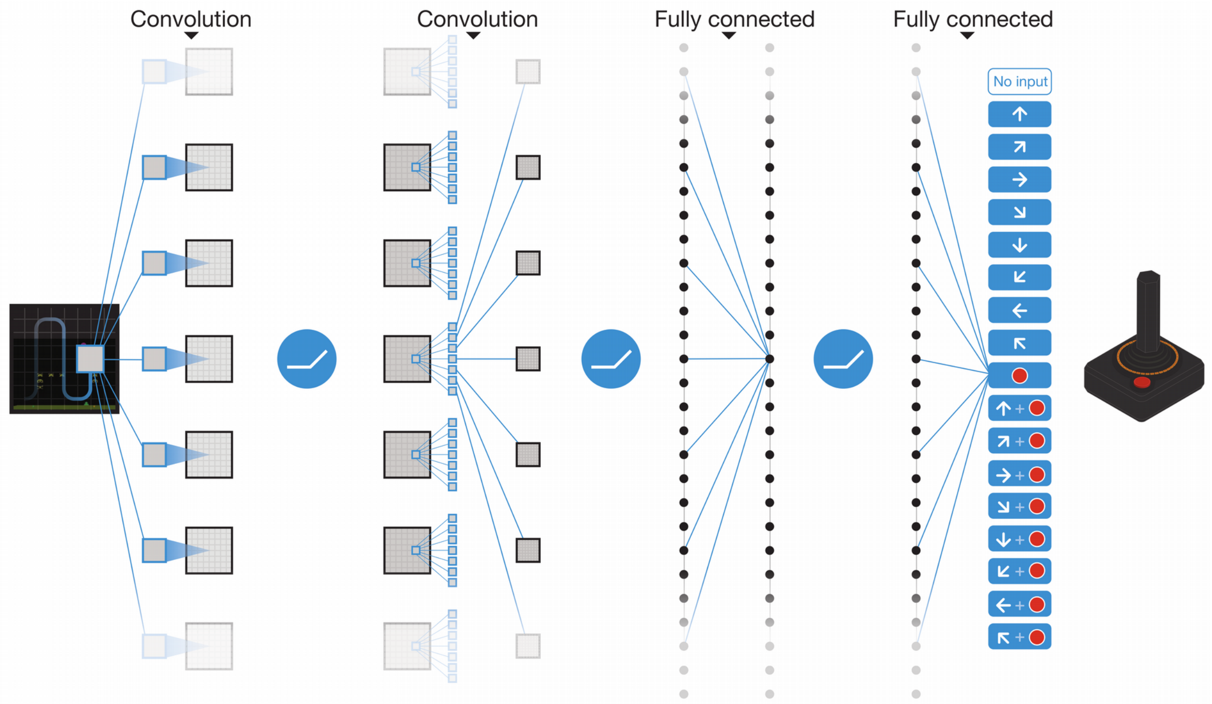




- As policies complexify, cost of decision-making remains low

Complexity

Deep Q Network (DQN)



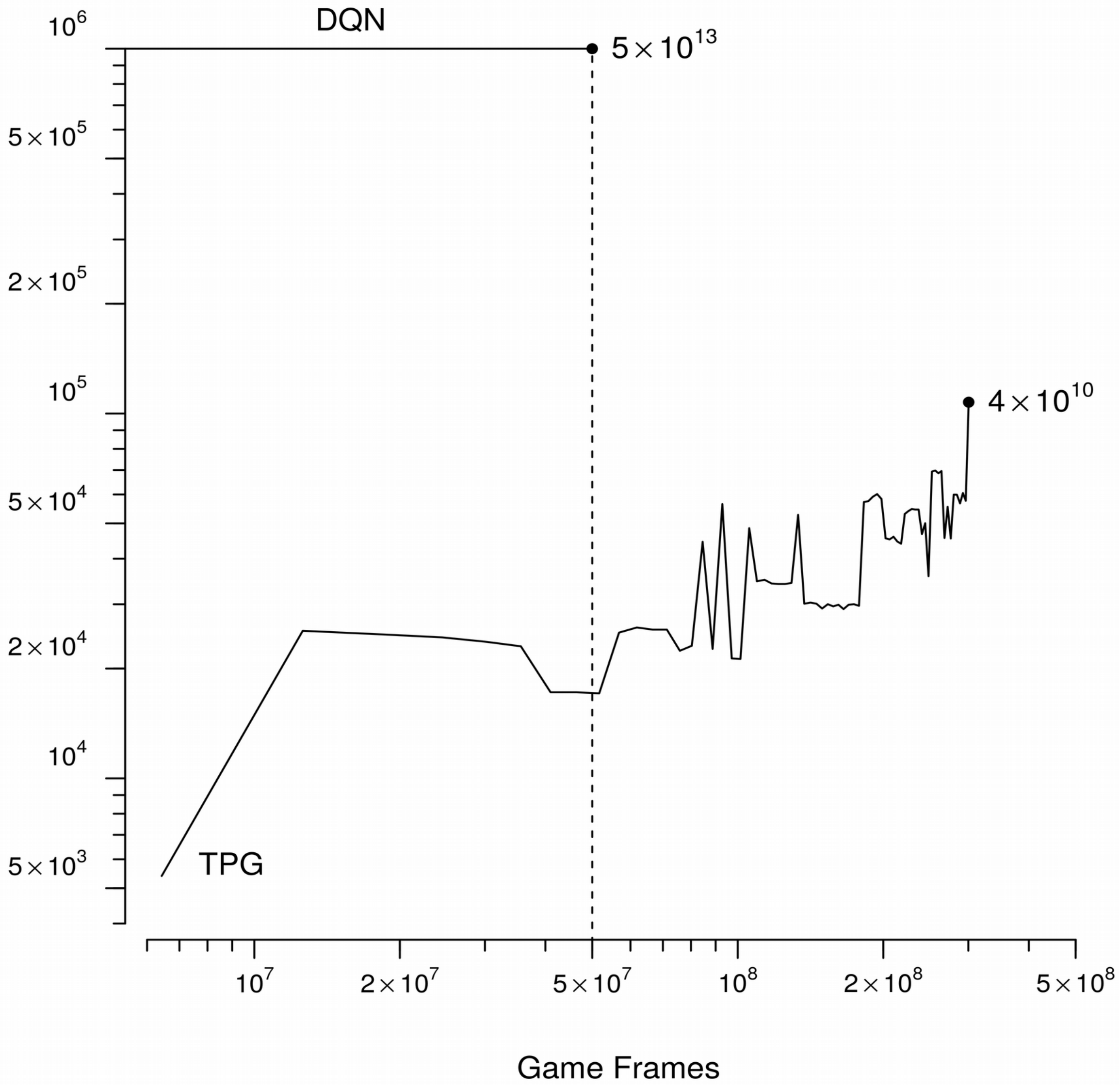
- Entire network contributes to each decision

Mnih et. al., Human-level control through deep reinforcement learning, Nature 518(7540) (2015) 529–533

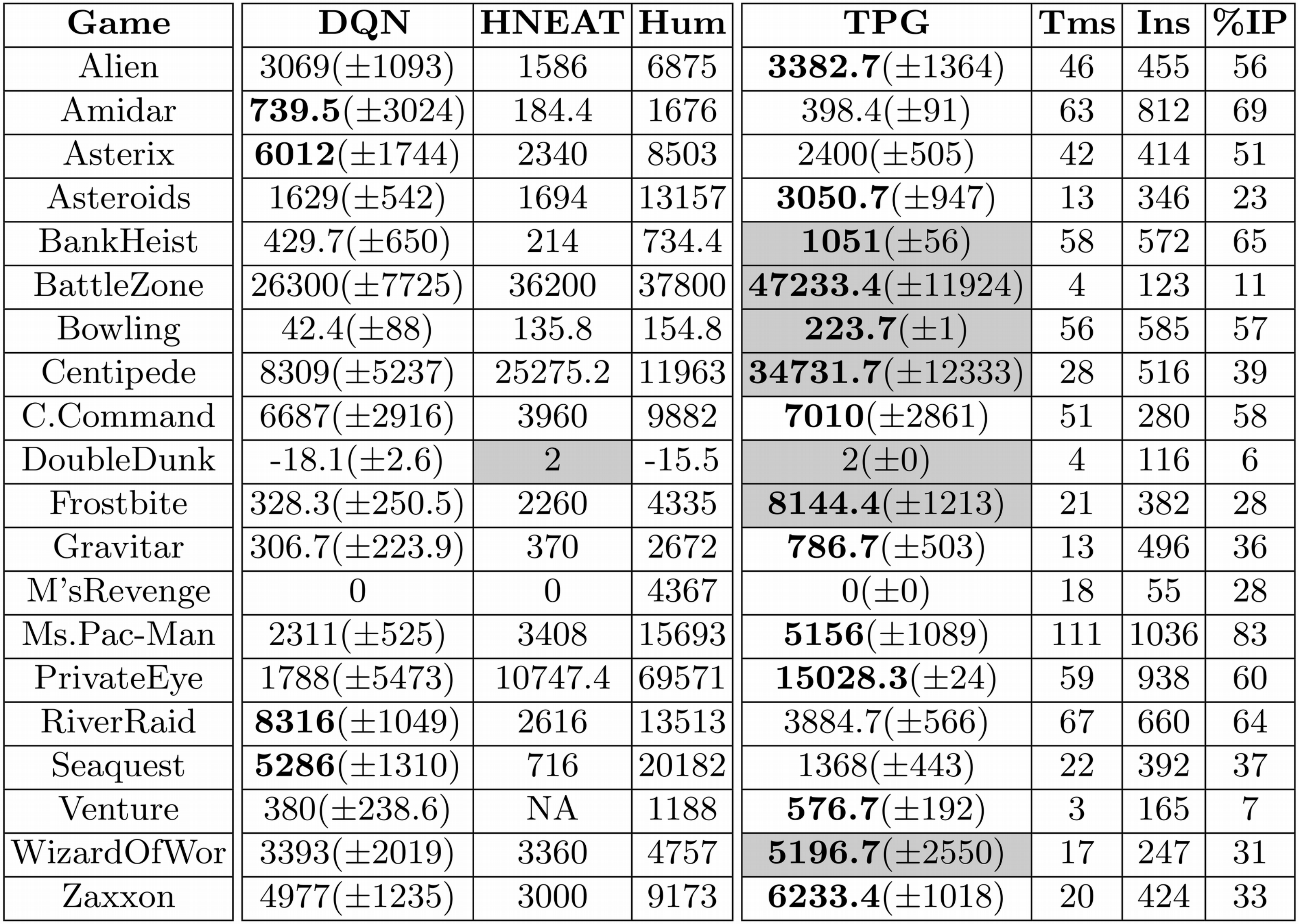
Complexity: Training Cost

Operations per Frame:

DQN - # of weights
TPG - # of instructions



Atari 2600 Results



Game	DQN	HNEAT	Hum	TPG	Tms	Ins	%IP
Alien	3069(±1093)	1586	6875	**3382.7**(±1364)	46	455	56
Amidar	**739.5**(±3024)	184.4	1676	398.4(±91)	63	812	69
Asterix	**6012**(±1744)	2340	8503	2400(±505)	42	414	51
Asteroids	1629(±542)	1694	13157	**3050.7**(±947)	13	346	23
BankHeist	429.7(±650)	214	734.4	**1051**(±56)	58	572	65
BattleZone	26300(±7725)	36200	37800	**47233.4**(±11924)	4	123	11
Bowling	42.4(±88)	135.8	154.8	**223.7**(±1)	56	585	57
Centipede	8309(±5237)	25275.2	11963	**34731.7**(±12333)	28	516	39
C.Command	6687(±2916)	3960	9882	**7010**(±2861)	51	280	58
DoubleDunk	-18.1(±2.6)	2	-15.5	2(±0)	4	116	6
Frostbite	328.3(±250.5)	2260	4335	**8144.4**(±1213)	21	382	28
Gravitar	306.7(±223.9)	370	2672	**786.7**(±503)	13	496	36
M'sRevenge	0	0	4367	0(±0)	18	55	28
Ms.Pac-Man	2311(±525)	3408	15693	**5156**(±1089)	111	1036	83
PrivateEye	1788(±5473)	10747.4	69571	**15028.3**(±24)	59	938	60
RiverRaid	**8316**(±1049)	2616	13513	3884.7(±566)	67	660	64
Seaquest	**5286**(±1310)	716	20182	1368(±443)	22	392	37
Venture	380(±238.6)	NA	1188	**576.7**(±192)	3	165	7
WizardOfWor	3393(±2019)	3360	4757	**5196.7**(±2550)	17	247	31
Zaxxon	4977(±1235)	3000	9173	**6233.4**(±1018)	20	424	33

Conclusion

- Tangled Program Graph (TPG) representation is proposed
- TPG policies are competitive with deep learning in Atari video games

- Critical benefits:

1) **Simplicity:** Policies start simple and complexify through interaction with the task (solution complexity is a learned property)
2) **State space selectivity:** Policies learn how to sub-sample from high-dimensional sensory inputs <u>and</u> hierarchically organize decisions made in each region

Future Work

Multi-Task Learning in Atari Video Games

Still working from raw screen capture, a single evolutionary run produces:

- champion policies for multiple game titles
- a single policy capable of playing multiple game titles

Stephen Kelly and Malcolm I. Heywood. Multi-Task Learning in Atari Video Games with Emergent Tangled Program Graphs. In Proceedings of the 2017 Genetic and Evolutionary Computation Conference (GECCO '17)