Learning to Imitate a Fighter Pilot

Shie Mannor

Department of Electrical Engineering
Technion

ICML July 2011


© copyright : Peter Steehouwer http://www.steehouwer.com

We want to imitate

The system we have

Features of combat flight

- Versatility.
- Accuracy.
- Randomness.
- Multi-agency.
- Concurrency.

Features of combat flight

- Versatility.
- Accuracy.
- Randomness.
- Multi-agency.
- Concurrency.

Features of combat flight

- Versatility.
- Accuracy.
- Randomness.
- Multi-agency.
- Concurrency.

Features of combat flight

- Versatility.
- Accuracy.
- Randomness.
- Multi-agency.
- Concurrency.

Features of combat flight

- Versatility.
- Accuracy.
- Randomness.
- Multi-agency.
- Concurrency.

RL view

- Single reward: ± 1
- State space:
  Coordinates X, Y, Z, Velocity V_X, V_Y, V_Z
  Differentials ΔX, ΔY, ΔZ, Velocity ΔV_X, ΔV_Y, ΔV_Z
  Attack angle, speed, fuel, sun location ....
  Missiles/canons
  Defensive props
- Episodic task: 30-60 seconds per fight

RL view

- Single reward: ± 1
- State space:
  Coordinates X, Y, Z, Velocity V_X, V_Y, V_Z
  Differentials ΔX, ΔY, ΔZ, Velocity ΔV_X, ΔV_Y, ΔV_Z
  Attack angle, speed, fuel, sun location ....
  Missiles/canons
  Defensive props
- Episodic task: 30-60 seconds per fight

RL view

- Single reward: ± 1
- State space:
  Coordinates X, Y, Z, Velocity V_X, V_Y, V_Z
  Differentials ΔX, ΔY, ΔZ, Velocity ΔV_X, ΔV_Y, ΔV_Z
  Attack angle, speed, fuel, sun location ....
  Missiles/canons
  Defensive props
- Episodic task: 30-60 seconds per fight

Agents

- Pilots are good!
- Very fast responses
- Established flight methodology
- Principle: always better to be behind the enemy
- A bit like chess

Agents

- Pilots are good!
- Very fast responses
- Established flight methodology
- Principle: always better to be behind the enemy
- A bit like chess

Agents

- Pilots are good!
- Very fast responses
- Established flight methodology
- Principle: always better to be behind the enemy
- A bit like chess

Agents

- Pilots are good!
- Very fast responses
- Established flight methodology
- Principle: always better to be behind the enemy
- A bit like chess

Agents

- Pilots are good!
- Very fast responses
- Established flight methodology
- Principle: always better to be behind the enemy
- A bit like chess

RL: A Naive Approach

- Train agent against itself
- Train agent against a human player
- Temporal credit assignment problem
- Never win against even a moderately good player

RL: A Naive Approach

- Train agent against itself
- Train agent against a human player
- Temporal credit assignment problem
- Never win against even a moderately good player

RL: A Naive Approach

- Train agent against itself
- Train agent against a human player
- Temporal credit assignment problem
- Never win against even a moderately good player

An AI approach

- Just hack the damn thing
- Very hard to get good at the game
- Pilots training is 3-5 years
- Many maneuvers to consider

An AI approach

- Just hack the damn thing
- Very hard to get good at the game
- Pilots training is 3-5 years
- Many maneuvers to consider

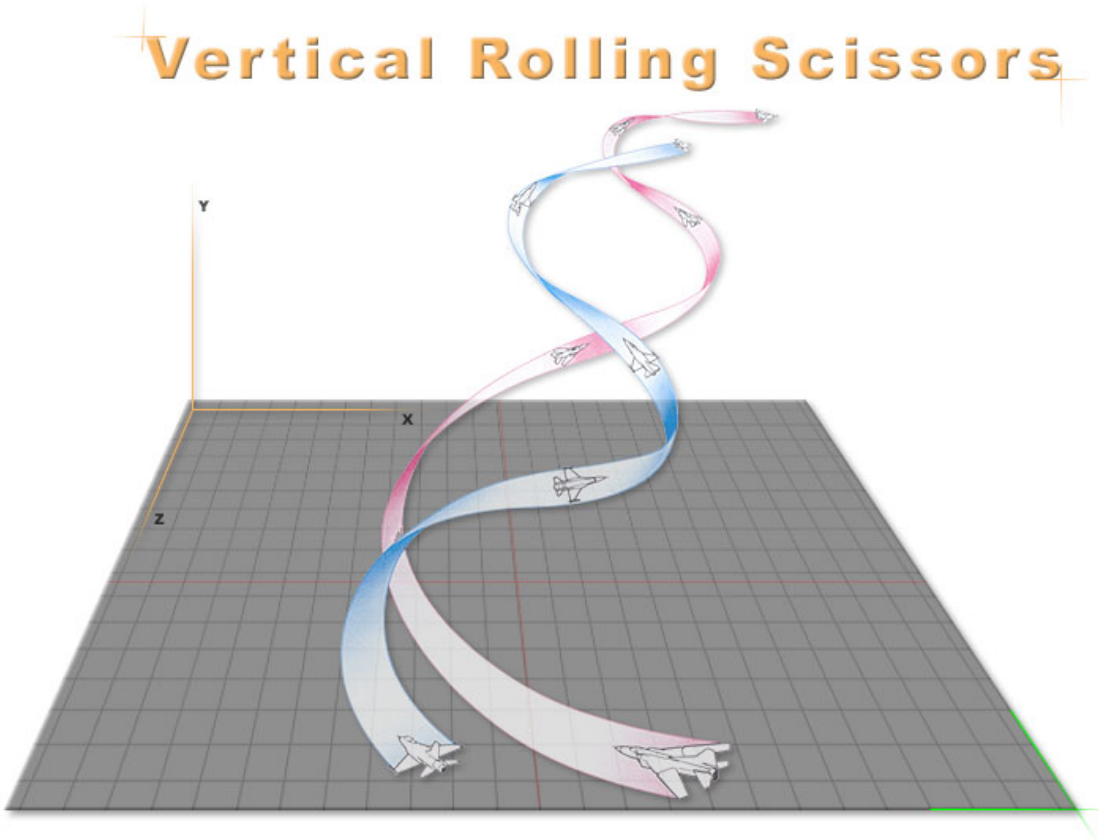
An AI approach

- Just hack the damn thing
- Very hard to get good at the game
- Pilots training is 3-5 years
- Many maneuvers to consider

An AI approach

- Just hack the damn thing
- Very hard to get good at the game
- Pilots training is 3-5 years
- Many maneuvers to consider

A sample maneuver



Our approach: Plural Reinforcement Learning

We can learn from many sources:

- Simulation
- Viewing “good" traces of human VS human
  Annotated or not
- Human VS agent
- Ask pilot what to do?
- Ask pilot what went wrong

Our approach: Plural Reinforcement Learning

We can learn from many sources:

- Simulation
- Viewing “good" traces of human VS human
  Annotated or not
- Human VS agent
- Ask pilot what to do?
- Ask pilot what went wrong

Our approach: Plural Reinforcement Learning

We can learn from many sources:

- Simulation
- Viewing “good" traces of human VS human
  Annotated or not
- Human VS agent
- Ask pilot what to do?
- Ask pilot what went wrong

Imitation: Step 1

Learn sequence of actions = options = macro-actions from traces

- Call them maneuvers
- Tried unsupervised learning on traces $\Rightarrow$ failed. Flight is too versatile.
- Got verbal description from expert:
  Expert inaccurate
  Lots of free parameters
  Articulating maneuvers is hard
- Parameterized maneuvers.

Imitation: Step 1

Learn sequence of actions = options = macro-actions from traces

- Call them maneuvers
- Tried unsupervised learning on traces $\Rightarrow$ failed. Flight is too versatile.
- Got verbal description from expert:
  Expert inaccurate
  Lots of free parameters
  Articulating maneuvers is hard
- Parameterized maneuvers.

Imitation: Step 1

Learn sequence of actions = options = macro-actions from traces

- Call them maneuvers
- Tried unsupervised learning on traces $\Rightarrow$ failed. Flight is too versatile.
- Got verbal description from expert:
  Expert inaccurate
  Lots of free parameters
  Articulating maneuvers is hard
- Parameterized maneuvers.

Imitation: Step 1

Learn sequence of actions = options = macro-actions from traces

- Call them maneuvers
- Tried unsupervised learning on traces $\Rightarrow$ failed. Flight is too versatile.
- Got verbal description from expert:
  Expert inaccurate
  Lots of free parameters
  Articulating maneuvers is hard
- Parameterized maneuvers.

Value function approach

- Illicit value from expert.
- Full advantage
- Full disadvantage
- Equality
- ....

Value function approach

- Illicit value from expert.
- Full advantage
- Full disadvantage
- Equality
- ....

Value function elicitation

- Greedy over expert value function $\Rightarrow$ trembling $\Rightarrow$ smoothing
- Features for linear approximation—unnatural
- Obtaining more constraints from traces
- Value function approach does not seem to cut it

Value function elicitation

- Greedy over expert value function $\Rightarrow$ trembling $\Rightarrow$ smoothing
- Features for linear approximation—unnatural
- Obtaining more constraints from traces
- Value function approach does not seem to cut it

Value function elicitation

- Greedy over expert value function $\Rightarrow$ trembling $\Rightarrow$ smoothing
- Features for linear approximation—unnatural
- Obtaining more constraints from traces
- Value function approach does not seem to cut it

Policy gradients

- Policy is complex
- Parametrization is tricky
- Had some success with isolated maneuvers
- Pilots do not use a pre-specified policy

Policy gradients

- Policy is complex
- Parametrization is tricky
- Had some success with isolated maneuvers
- Pilots do not use a pre-specified policy

Reward shaping

- Reward extremely rare
- Use expert for reward signals
- Hard to find a consistent reward function
- Randomization
- Risk aversion

Reward shaping

- Reward extremely rare
- Use expert for reward signals
- Hard to find a consistent reward function
- Randomization
- Risk aversion

Reward shaping

- Reward extremely rare
- Use expert for reward signals
- Hard to find a consistent reward function
- Randomization
- Risk aversion

Using traces for better features

Main challenge: when to start which maneuver and with which params.

How to combine all forms of input from pilots?

Challenges

- How to use rare feedback from expert?
- Stop and ask
- Initiating and defining maneuvers
- Better use of traces
- Reward shaping/learning maneuvers

Challenges

- How to use rare feedback from expert?
- Stop and ask
- Initiating and defining maneuvers
- Better use of traces
- Reward shaping/learning maneuvers

Challenges

- How to use rare feedback from expert?
- Stop and ask
- Initiating and defining maneuvers
- Better use of traces
- Reward shaping/learning maneuvers

Challenges

- How to use rare feedback from expert?
- Stop and ask
- Initiating and defining maneuvers
- Better use of traces
- Reward shaping/learning maneuvers

Challenges

- How to use rare feedback from expert?
- Stop and ask
- Initiating and defining maneuvers
- Better use of traces
- Reward shaping/learning maneuvers