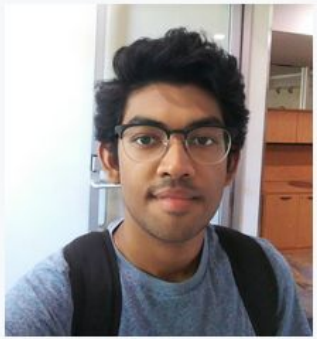
TerranFormer

Project Presentation

Hello!

We are TerranFormer

Team Introduction

Yuqian Cao (Kerry)	Gary Ng Kwong Sang	Waseem Hossain	Carter Sabadash
yuqian5@ualberta.ca	ngkwongs@ualberta.ca	whossain@ualberta.ca	sabadash@ualberta.ca
			

Motivate your work

- Build a good bot with our knowledge about sc2
- Get the bot to play the game better than we can

Goals

Effective and efficient scout

Optimal troop training

Optimize resource allocation

React dynamically to enemy actions

Bug free code, runtime efficient code

Prior Work

We decided to go for a trial and error approach, therefore, we didn’t consult any guide.

No particular build order, everything is based on the current available resources.

(Carter looked at bot_examples.cc)

Describe your approach in detail

- 4 Agents
- Easier development
- Interact with each other and the main bot
- Maximize early game survivability and aggressive expansion in middle and late game to hog resource

Base Agent Design

- Priority Queue
- Use ClientEvents
- Add Tasks to other agents
- Own units

Describe you approach cont.

- **Resource Agent**
  - mineral/gas
  - scvs
  - Command centers
  - Resource bottleneck


Minerals
Workers: 1/3
Workers: 16/16
Not enough vespene
Not enough vespene
Not enough vespene
Workers: 1/3
F1
F2
5:21
Menu
Not enough minerals
Not enough minerals
Not enough minerals
2:27


Workers: 3/3
110
69
66/70
Workers: 15/16
Queue is full
Queue is full
Not enough vespene
Workers: 3/3
0
F1
13
F2
7:02
0
Menu

Describe your approach cont.

- Scout Agent
  - Database of POIs
  - 5 SCVs
  - A task queue
  - Track viewed enemies

Describe your approach cont.

- Defence Agent
  - Choke Points
  - Air defence and cloak defence
  - Ground Defence
  - Perimeter Defence
  - Upgrades


Workers: 0/3
1295
21
46/46
ers: 16/16
ueue is full
ueue is full
ot enough vespene
0
10
F1
F2
5:12
Engineering Bay
Researching
850 / 850
2
3
4
5

Winning Strategy?

Describe your approach cont.

- **Attack Agent**
  - Troop training
  - Attack Strategies
  - Use Scout information

Goals accomplished

Effective and efficient scout

Optimal troop training

Optimize resource allocation

React dynamically to enemy actions

Bug free code, runtime efficient code

Evidence - Bug Free, Efficient Code

Playing against Hard Zerg - Time to 5min

	CactusValleyLE	ProximaStationLE	BelShirVestigeLE
BasicSC2Bot	41s	45s	38s
Our Bot	60s	64s	58s
Difference	19s	19s	20s
Time Increase	**47%**	**42%**	**53%**
AVG	**47%**		

Advantages and Disadvantages

Advantages

- High Actions Per Minute (APM)
- Modular code

Disadvantages

- Performance issues
- Competing for resources
- Complex Inter-Agent clearance structure

Future Work

- Behaviour Trees
- Use all available features
- Apply some RL techniques

  We believe that RL can be very useful when creating a engagement policy.
- Code refactoring
- Multithreading (already designed for this)

Video - Our bot winning

Any questions?