Multi-Reactive Planning for Real-Time Strategy Games

Alberto Uriarte Pérez

UAB

September 8, 2011

Advisor: Santiago Ontañón Villar

Outline

1. Motivation
2. Real-time multi agent system
    - Architecture overview
    - Working Memory Information
    - Micro management
    - Macro management
3. Experimental evaluation
4. Conclusions and Future Work
    - Conclusions
    - Future Work

Motivation

AI research has been focused on turn-based games like Chess. However, RTS games offer a more complex scenario.

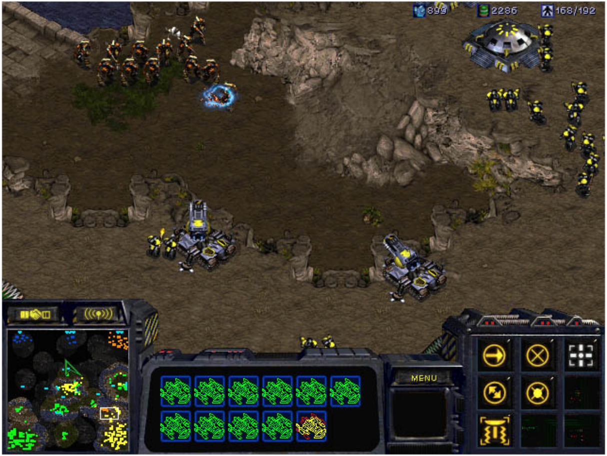
- How to build a multi agent system capable of reasoning and cooperating in a real-time environment
- Concurrent and adversarial planning under uncertainty
- Spatial and temporal reasoning

Claim

Performing an immersion in the knowledge of the domain and implementing some of the latest AI techniques, we can improve the built-in AI of the game, beat other bots and be a real challenge for a human expert player

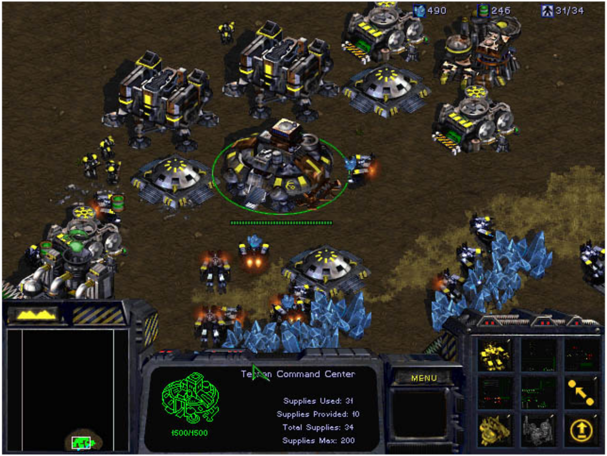
Starcraft

Micro management



Starcraft

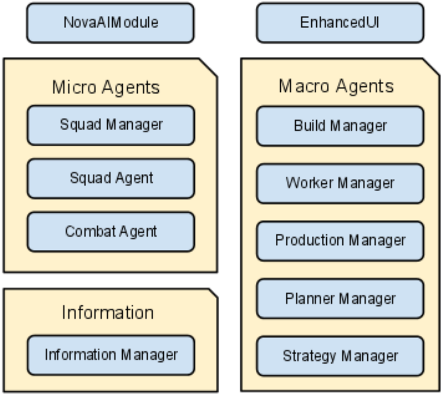
Macro management



Terran Command Center

Supplies Used: 31
Supplies Provided: 10
Total Supplies: 34
Supplies Max: 200

Architecture overview



Architecture overview

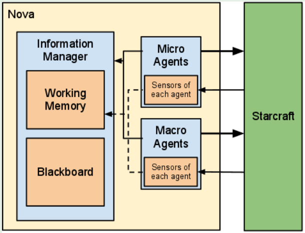
Problem

Real-time communication between agents.

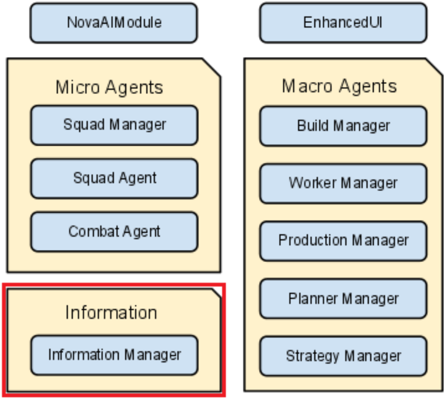
Solution

Blackboard Architecture.

Working Memory.



Working Memory Information



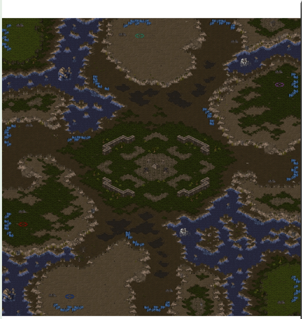
Terrain analysis

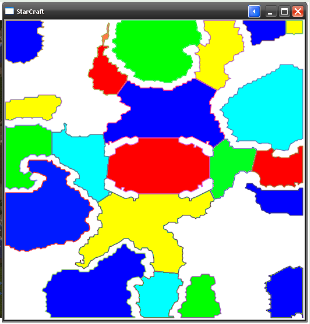
Problem

Different maps with different tactical advantatges.

Solution

Off-line terrain analysis





Opponent modelling

Problem

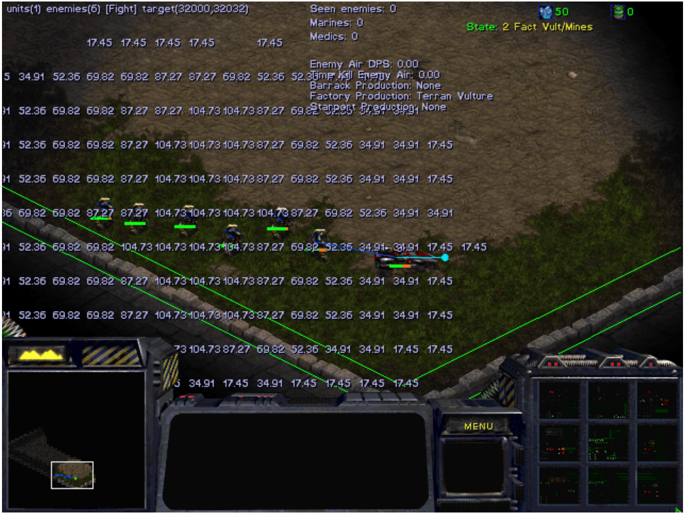
Imperfect information.

Solution

Opponent build order prediction (scouting).
Opponent's military units tracking (threat map).

Gathering task	**Usefulness**
Initial scout with a unit	Enemy start location Detect rush strategy
Scanner sweep scanning	Detect target locations
Enemy air/ground DPS	Decide to build anti-air units
Threat map	Pathfinding to save location Avoid dangerous regions

Threat map

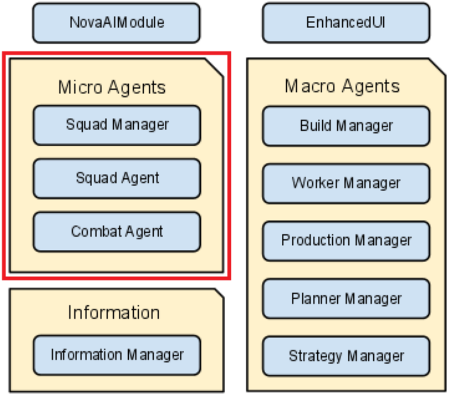


```
units(1) enemies(6) [Fight] target(32000,32032)

            17.45  17.45  17.45  17.45         17.45

5  34.91  52.36  69.82  69.82  87.27  87.27  69.82  52.36  52.3

91 52.36  69.82  69.82  87.27  87.27  104.73 104.73 87.27  69.82  52.36  34.91  34.91

91 52.36  69.82  69.82  87.27  104.73 104.73 104.73 87.27  69.82  52.36  34.91  34.91  17.45

91 52.36  69.82  69.82  87.27  104.73 104.73 104.73 87.27  69.82  52.36  34.91  34.91  17.45

36 69.82  69.82  87.27  87.27  104.73 104.73 104.73 104.73 87.27  69.82  52.36  34.91  34.91

91 52.36  69.82  69.82  104.73 104.73 104.73 104.73 87.27  69.82  52.36  34.91  34.91  17.45  17.45

91 52.36  69.82  69.82  87.27  104.73 104.73 104.73 87.27  69.82  52.36  34.91  34.91  17.45

91 52.36  69.82  69.82  87.27  104.73 104.73 104.73 87.27  69.82  52.36  34.91  34.91  17.45

                          73 104.73 87.27  69.82  52.36  34.91  34.91  17.45  17.45

                          5  34.91  17.45  34.91  17.45  17.45  17.45  17.45
```

```
Seen enemies: 0
Marines: 0
Medics: 0

Enemy Air DPS: 0.00
Time Kill Enemy Air: 0.00
Barrack Production: None
Factory Production: Terran Vulture
Starport Production: None
```

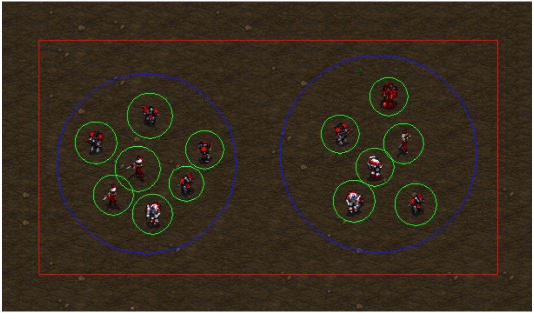
```
50    0
State: 2 Fact Vult/Mines
```

Micro management agents



Micro management agents

SquadManager, SquadAgent and CombatAgent follow a military organization.



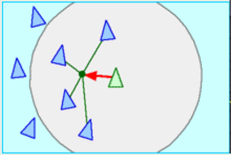
Squad Agent

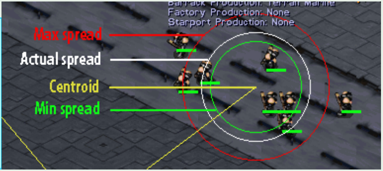
Problem

Effective squad movement.

Solution

Steering behaviours.





Combat Agent

Problem

Target selection.

Solution

Assigning a score to each target.

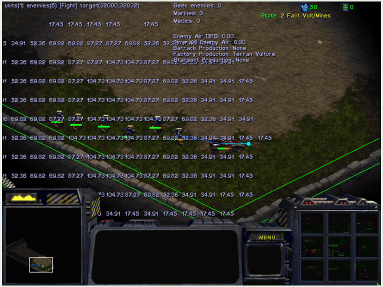
Combat Agent

Problem

Range units avoiding damage from melee units.

Solution

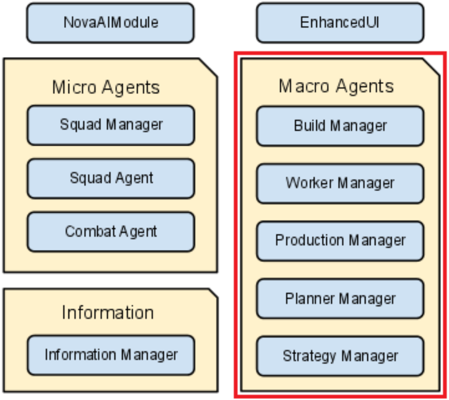
Potential fields.



```
units(1) enemies(6) [Fight] target(32000,32032)
Seen enemies: 0
Marines: 0
Medics: 0

Enemy Air DPS: 0.00
Time Kill Enemy Air: 0.00
Barrack Production: None
Factory Production: Terran Vulture
Starport Production: None

State: 2 Fact Vult/Mines
```

```
            17.45 17.45 17.45 17.45       17.45
5  34.91 52.36 69.82 69.82 87.27 87.27 69.82 52.36 52.3
1  52.36 69.82 69.82 87.27 87.27 104.73 104.73 87.27 69.82 52.36 34.91 34.91
1  52.36 69.82 69.82 87.27 104.73 104.73 104.73 87.27 69.82 52.36 34.91 34.91 17.45
1  52.36 69.82 69.82 87.27 104.73 104.73 104.73 87.27 69.82 52.36 34.91 34.91 17.45
6  69.82 69.82 87.27 87.27 104.73 104.73 104.73 104.73 87.27 69.82 52.36 34.91 34.91
1  52.36 69.82 69.82 104.73 104.73 104.73 104.73 87.27 69.82 52.36 34.91 34.91 17.45 17.45
1  52.36 69.82 69.82 87.27 104.73 104.73 104.73 87.27 69.82 52.36 34.91 34.91 17.45
1  52.36 69.82 69.82 87.27 104.73 104.73 104.73 87.27 69.82 52.36 34.91 34.91 17.45
                 73 104.73 87.27 69.82 52.36 34.91 34.91 17.45 17.45
                 5 34.91 17.45 34.91 17.45 17.45 17.45 17.45
```

Macro management agents



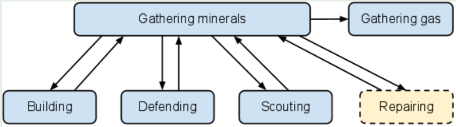
Gathering resources

Problem

Workers' tasks and income rate.

Solution

Finite State Machine.



$$2 \ workers \times mineral \ field$$

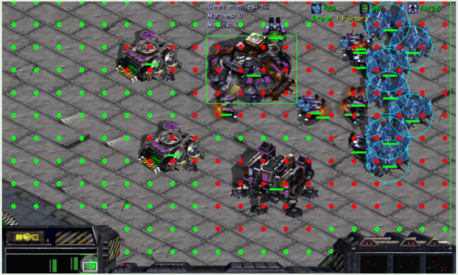
Building

Problem

Finding a build location.

Solution

Build map information with spiral-search alghoritm.



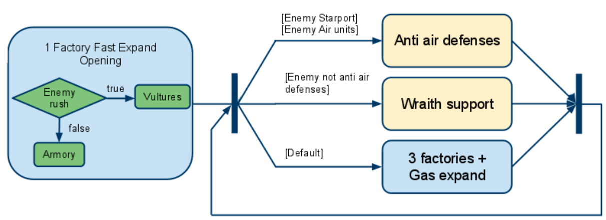
Strategies

Problem

Planning strategies and reactive behaviour.

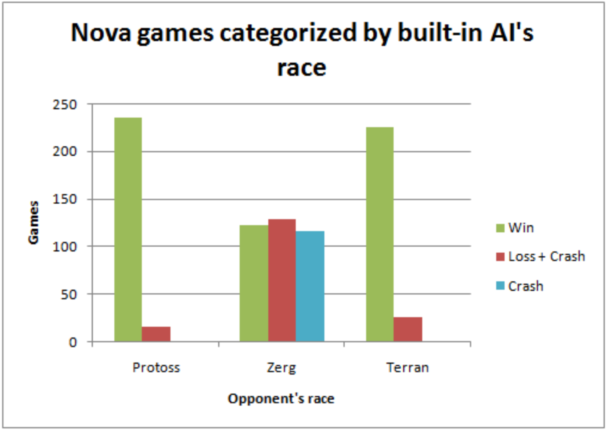
Solution

FSM with common states and trigger conditions.



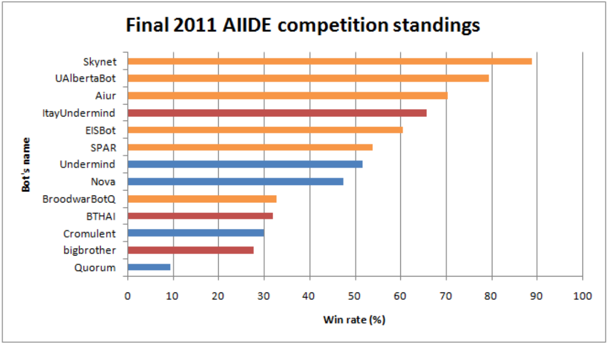
Nova Vs Built-in AI

Results after 250 games against each race

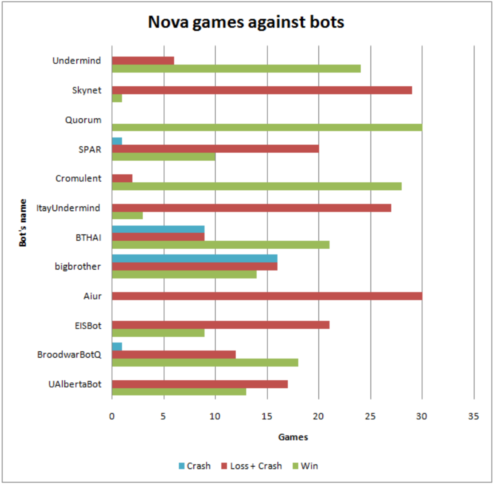


Nova Vs Bots

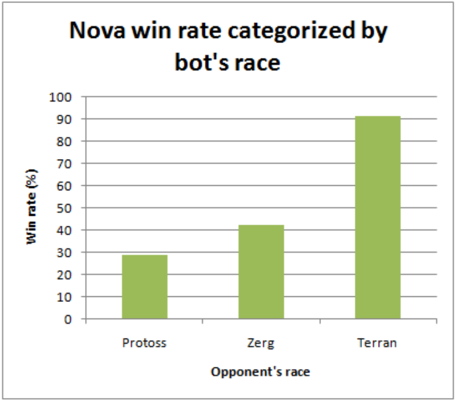
We tested our Nova bot in AIIDE Starcraft AI Competition



Nova Vs Bots


Nova games against bots
Undermind
Skynet
Quorum
SPAR
Cromulent
ItayUndermind
BTHAI
bigbrother
Aiur
EISBot
BroodwarBotQ
UAlbertaBot
Bot's name
0
5
10
15
20
25
30
35
Games
Crash
Loss + Crash
Win

Nova Vs Bots


Nova win rate categorized by bot's race
Win rate (%)
100
90
80
70
60
50
40
30
20
10
0
Protoss
Zerg
Terran
Opponent's race

Conclusions

- Working Memory and a Blackboard has given us good results on this real-time environment.
- The crashes indicate that we need better tools for debugging.
- Potential Fields raise as an effective tool for tactical decisions.
- Unit control task can improve a lot the bot's performance.
- Using FSM as our main strategy handler makes Nova easy to predict and less effective against undefined situations

Future Work

- Coordinate squads to achieve joint goals.
- Exploit tactical locations to take advantage in combats.
- Use squad formations to flank the enemy and test more squad movement behaviours.
- Improve the opponent modelling.
- Test other techniques for planning like GOAP.
- Design a learning system to emerge new AI behaviours and/or strategies.