Bayesian programming applied to Starcraft

A summer project experiment:
or why we can't come to the beach

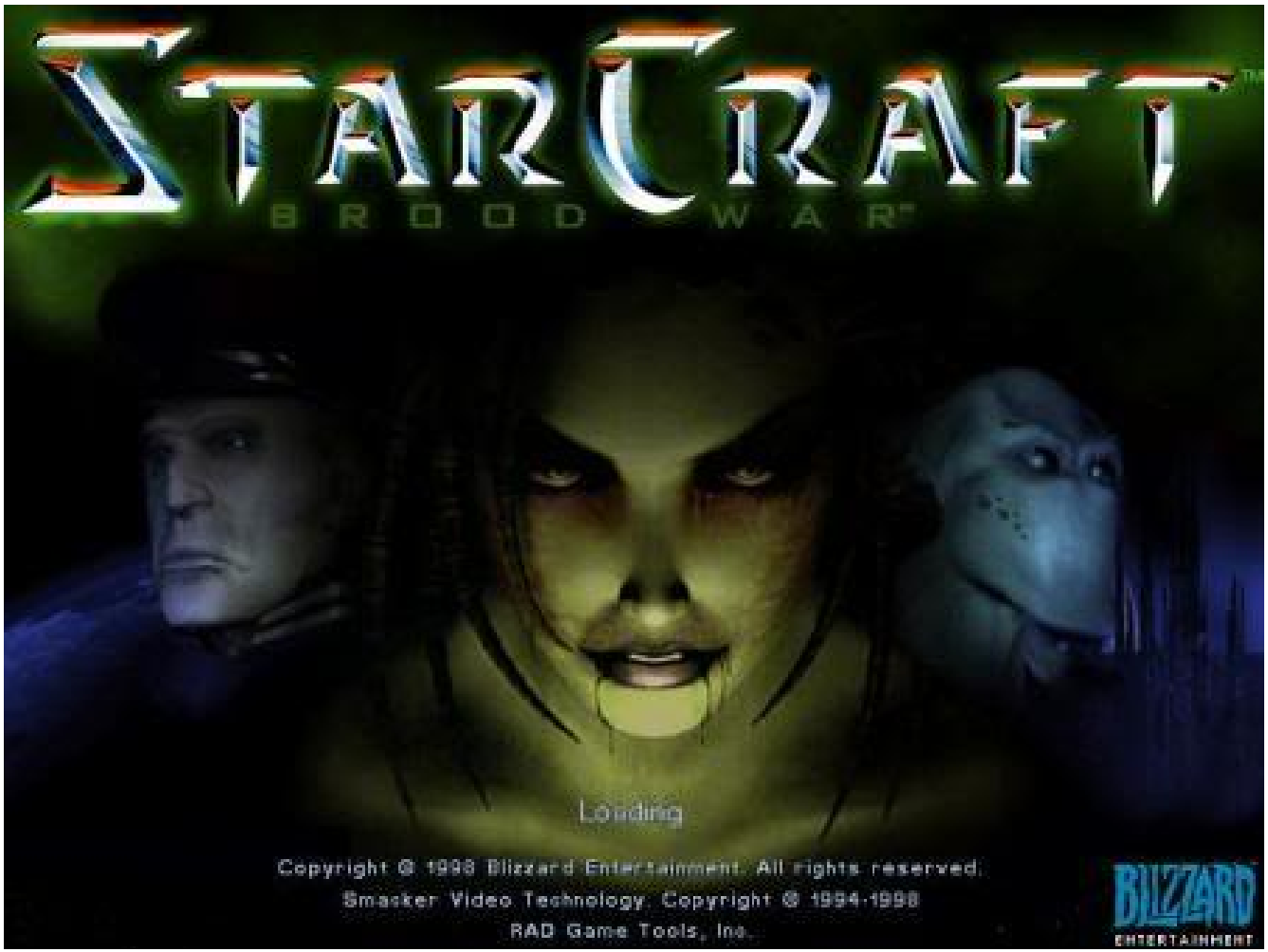
Gabriel Synnaeve

University of Grenoble / E-Motion @ INRIA

December 1, 2010

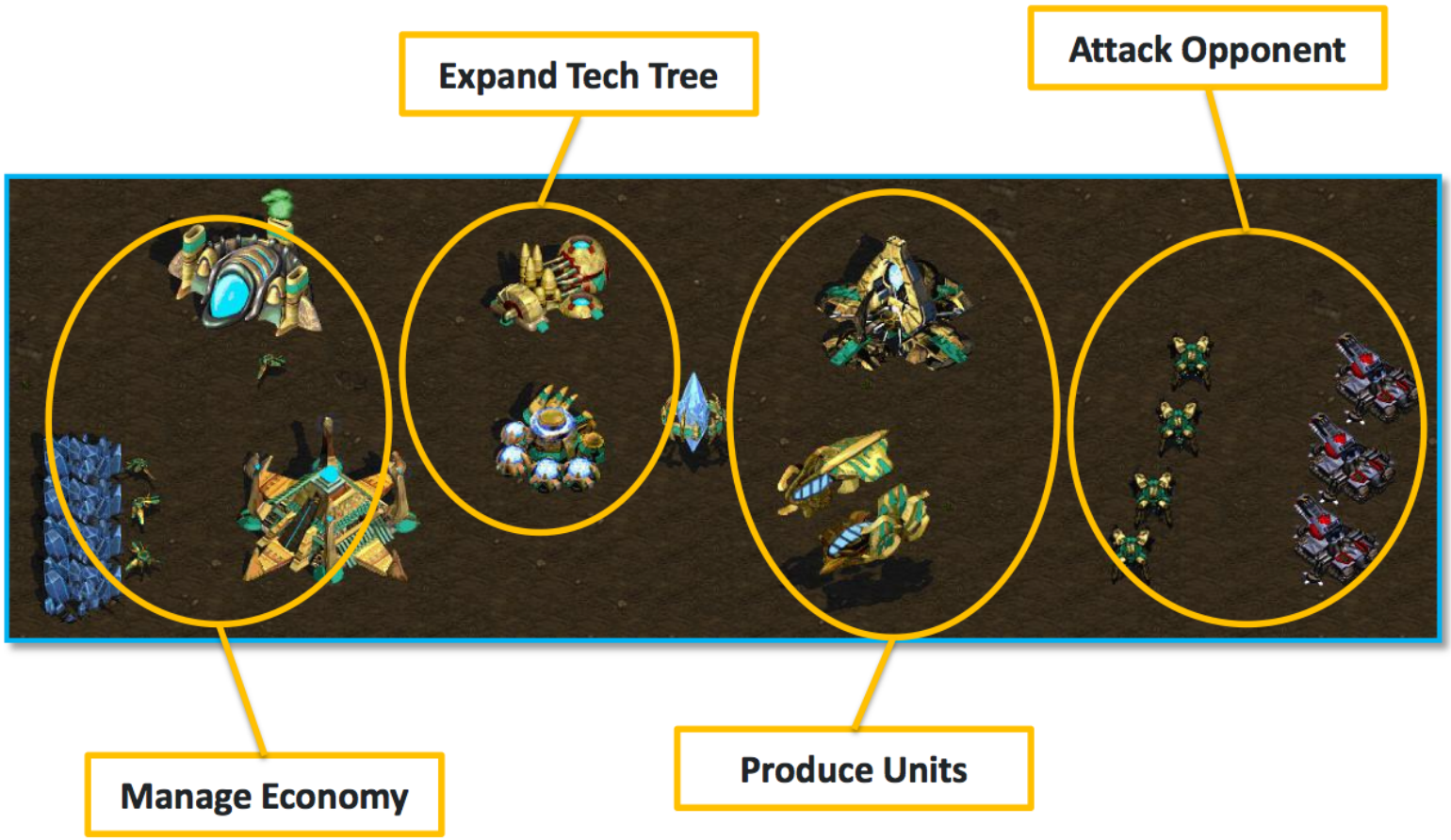
Starcraft: Broodwar

Starcraft (January 1998) + Broodwar (exp., November 1998)



RT what?

Real Time Strategy game



Pro gaming and competitions

eSports, sponsorship, tournaments' dotations, salaries (Korea)

WCG
WORLD CYBER GAMES

WORLD CYBER GAMES

The World Cyber Games is the world's first "Cyber Game Festival", designed to build a healthy cyber culture.
The best gamers around the world gather into different cities to share the excitement and fun of the game tournaments.

ESWC
ELECTRONIC SPORTS WORLD CUP

GOSU GAMERS

teamliquid
STARCRAFT PROGAMING NEWS • COMMUNITY • TEAM

Starcraft in numbers

- **11** years of competitive play
- **200** to **300** actions per minute amongst pro gamers
- **10** millions licenses sold (4.5 in South Korea)
- **$60000** average pro gamer salary (average South Korean: $16000)
- **1/4** TV channels are displaying Starcraft games in Korea
- **160** BPM: reached rates of pro gamers hearts
- **3** millions licenses sold for Starcraft II in 1 month (1 million in 24 hours, released July 27, 2010)
- **4.9** millions South Koreas still playing SC:Broodwar (August 2010)

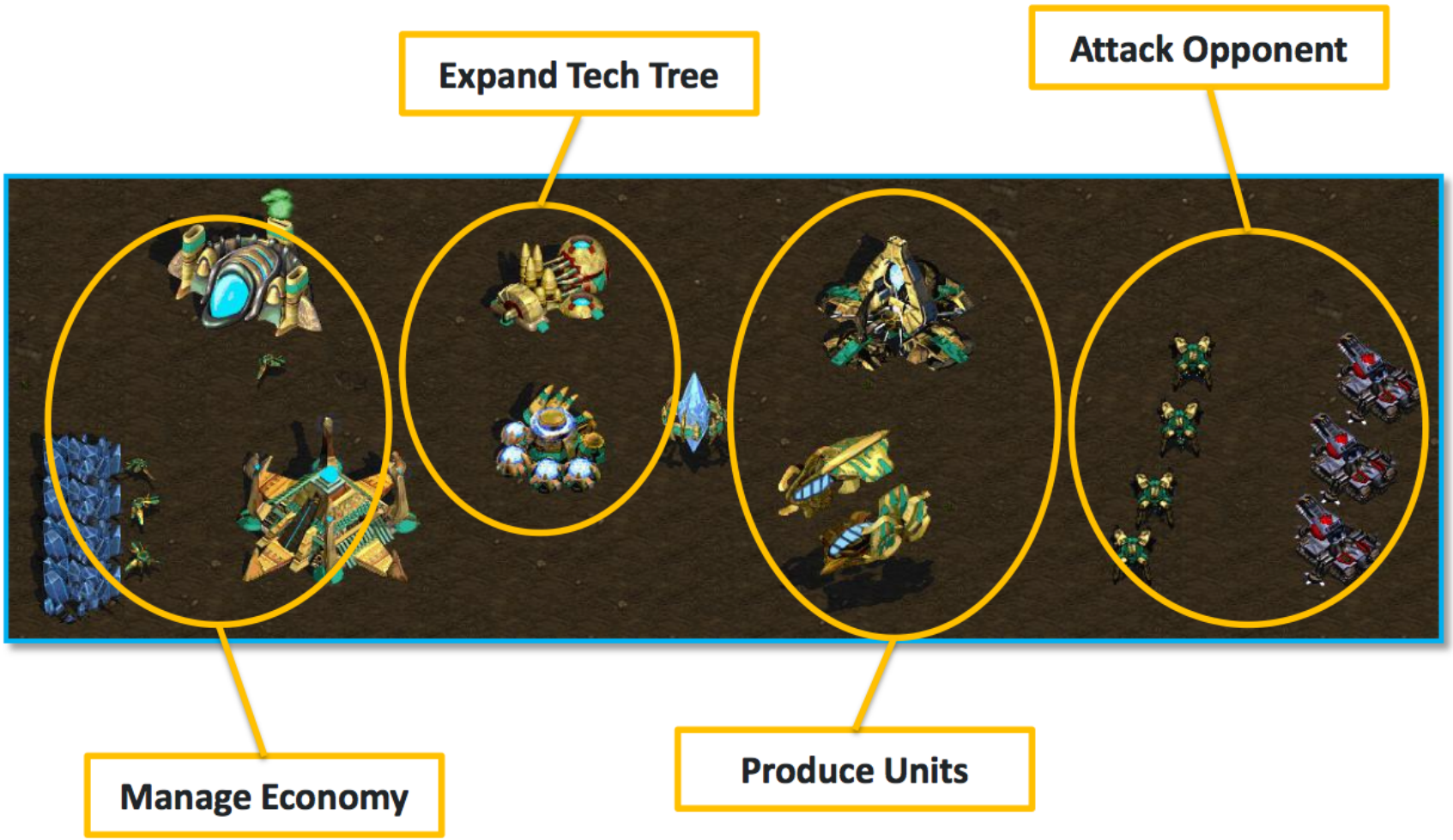
Micro/Macro

Macro - Economy and Strategy

4 resources:

- Minerals: basic resource, needed for every production
- Gas: advanced resource, needed for “high tech” building/units/research/upgrades
- Food cap: needed to increase it to produce units
- Time: everything takes time to build

$\rightarrow$ Develop tech tree
$\rightarrow$ Develop production capacity
$\rightarrow$ Produce units
$\rightarrow$ Enhance units


Expand Tech Tree
Attack Opponent
Manage Economy
Produce Units

Micro - Tactics and Unit Management

Micro-management is using your units as close as possible to the maximum outcome you can have with them.

Basically, units have:

- Hit points
- Size
- Speed
- Can fly (or not)
- Energy points
- Spells

Basics:

- Focus fire weak units / high damage units
- Retreat wounded units
- Cast spells
- Placement (short time tactics)

Toy problem?

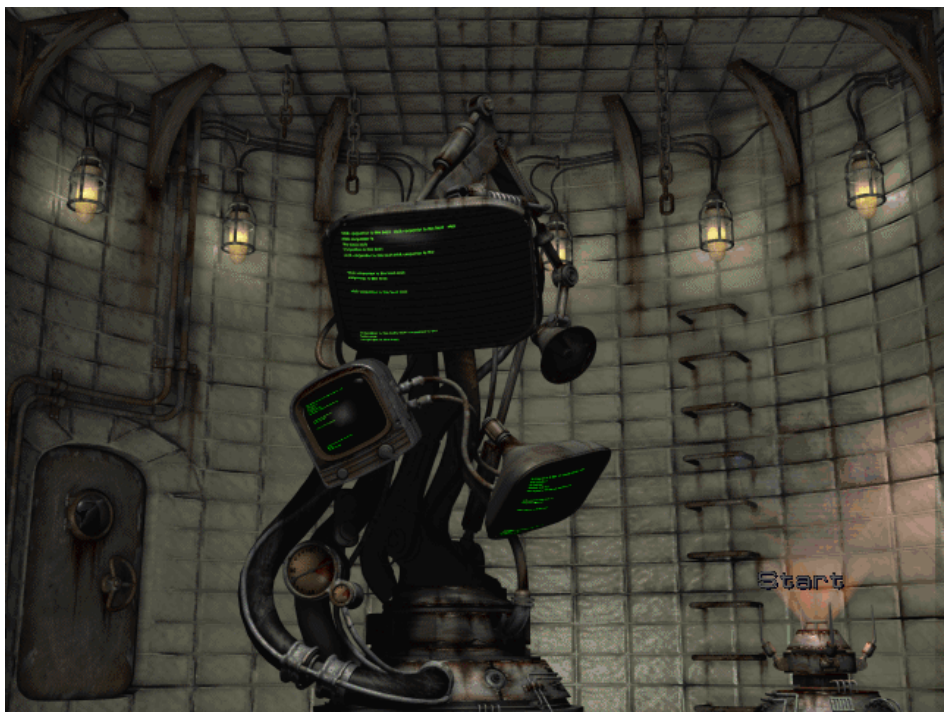
Multiplayer video games AI:

- Simulated: no sensors problems
- Multiplayer: real humans
- Closed and rich worlds

$\Rightarrow$ Good in-between real world / simulation.

Starcraft specifically

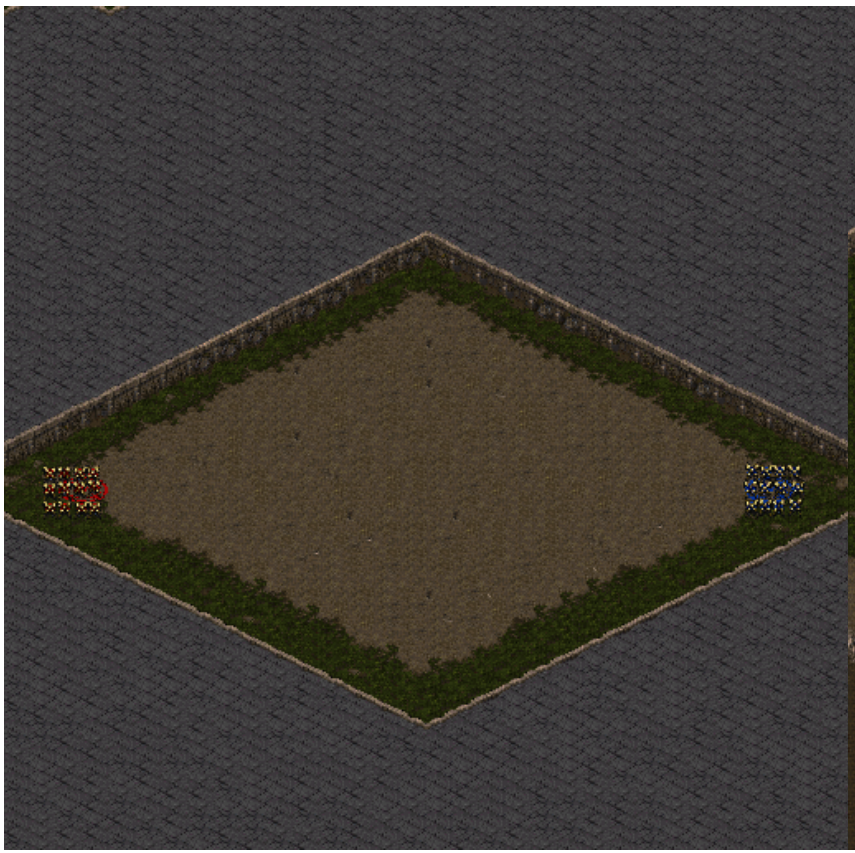
- Chess / Go / Rock-paper-scissors
- Real-time (1/24th second per micro-turn)
- Machine learning ready (replays)
- AI competitions



Open tournament

- Expressive Intelligence Studio at University of Santa Cruz (UCSC)
- Ben Weber announced the tournament in November 2009 at AIIDE
- Free participation
- Open to all
- Began work on it in March 2010 as a pet project
- Due for the 15th of September (2010)

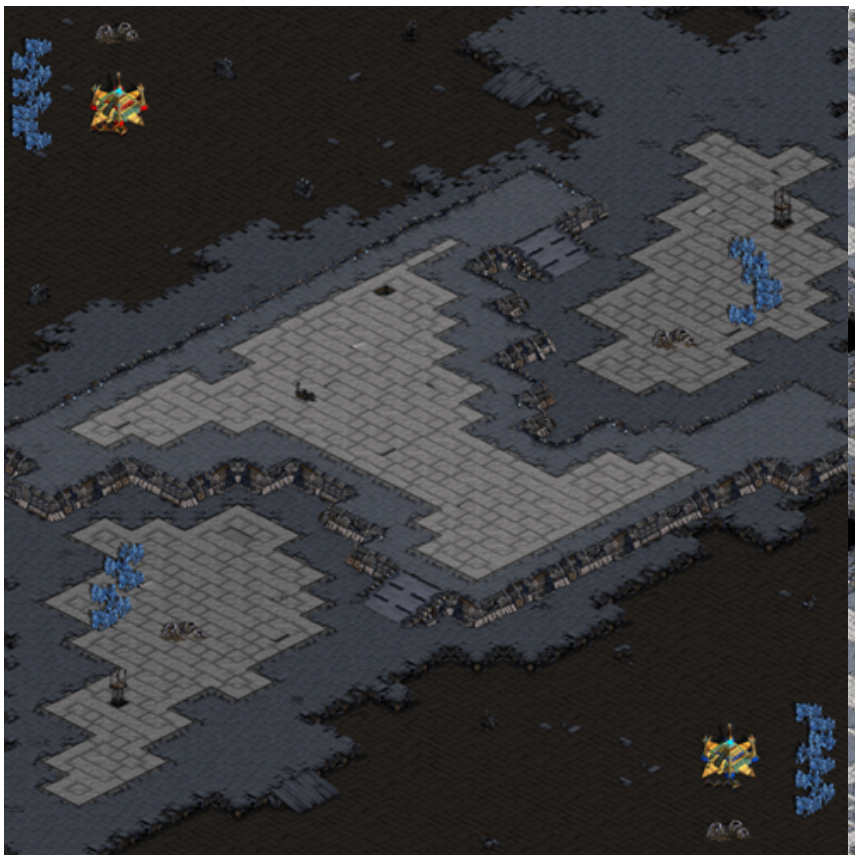
Micro-based tournaments

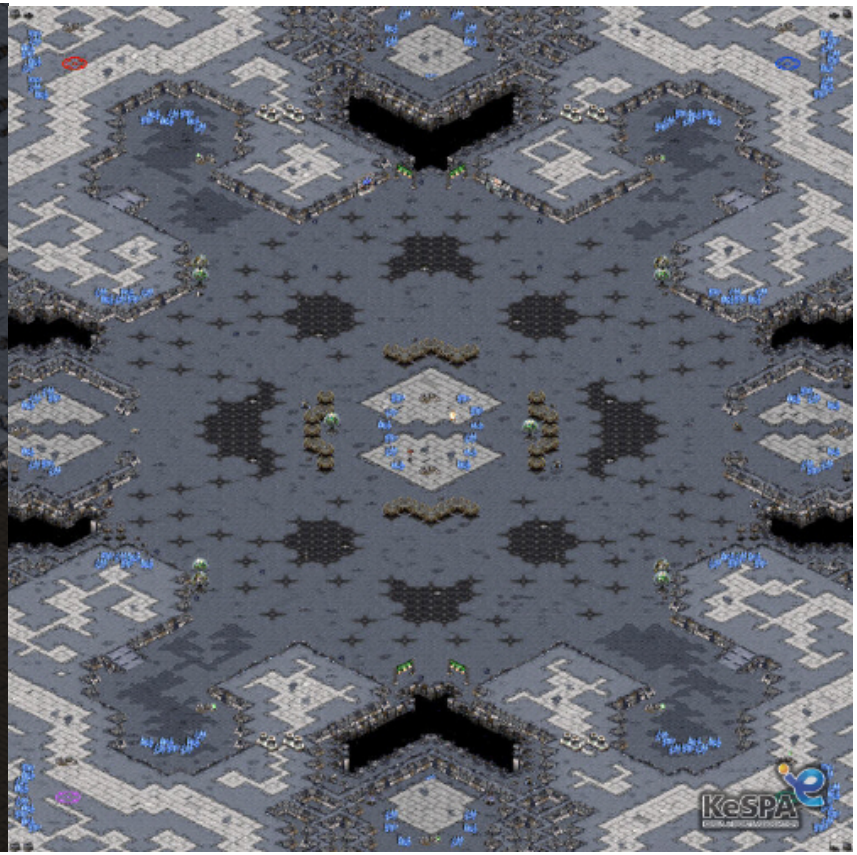


Micro-management: flat map;

Small-scale combat: cliffs

Normal games





Tech-limited: small map, only zealots/goons;

Full game: real (competitions, progaming) maps

Challenges in micro-management

Fight well, move well, think fast:

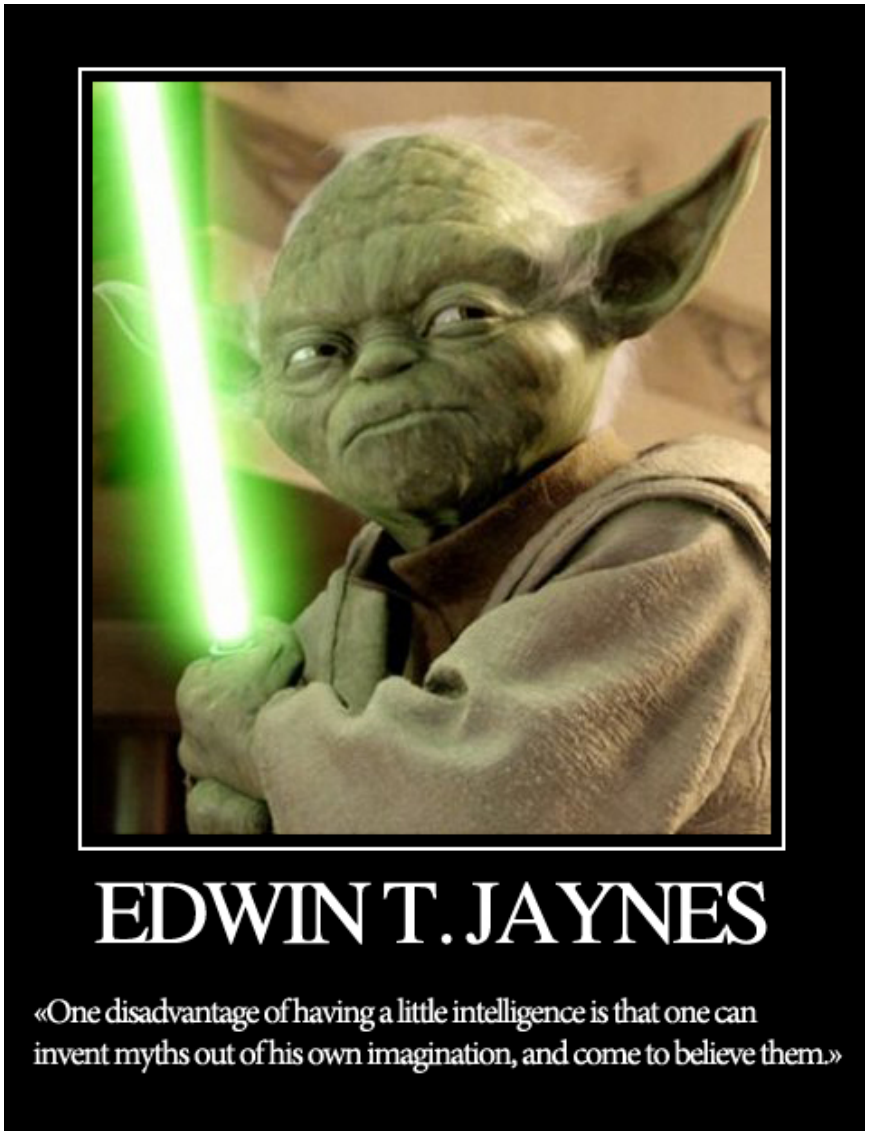
- Select targets
- Don’t stuck units (very hard!)
- Predict (close) future
- Be efficient (200x200, 1/24th of a second)

Challenges in full games

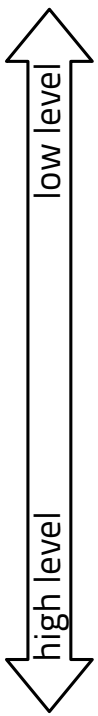
Chess in real-time:

- Do not bug/freeze/slow down
- Predict oponents strategies and tactics
- Be able to counter/handle any strategy
- Know when to attack and when not
- Micro-manage well in different environments
- A lot of other tasks
- Do not bug/freeze/slow down
- Do not bug/freeze/slow down
- ...

Bayesianism



Transmute incompleteness into uncertainty



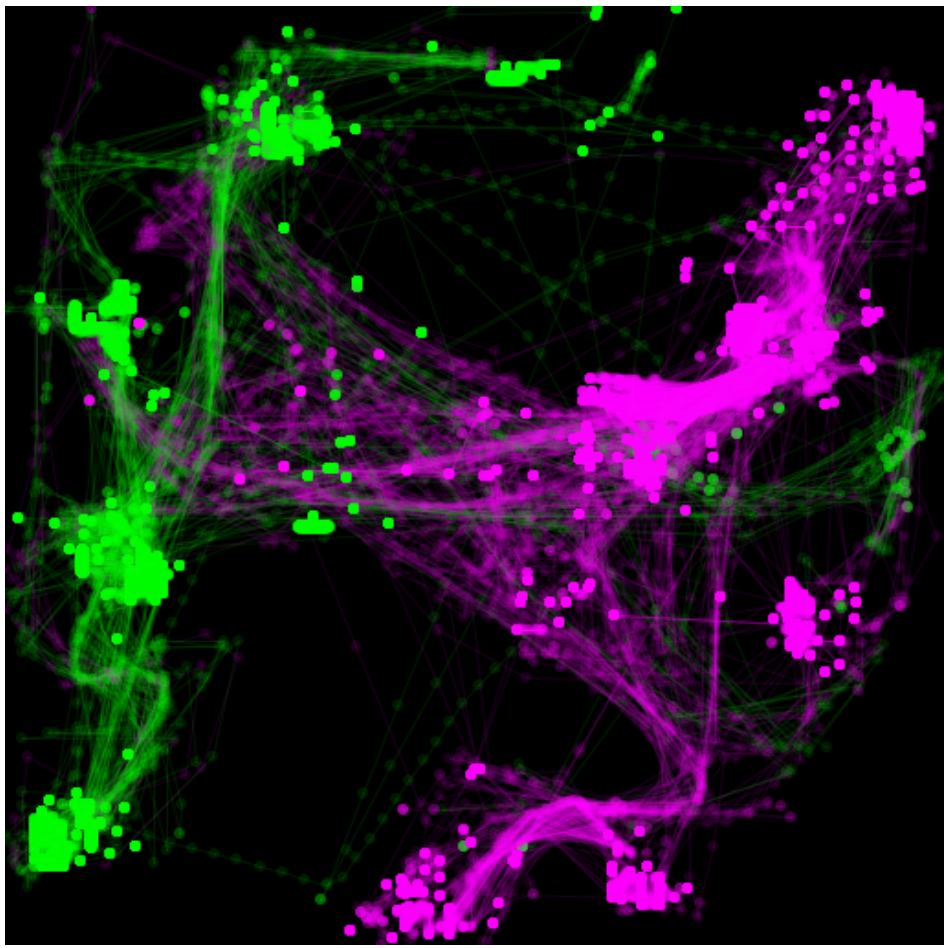
Incompleteness $\Longrightarrow$ Uncertainty

- Many low level moves achieving the same high level goal
- Fog of war (limited sight)
- Partial knowledge of opponent's force (size and composition)

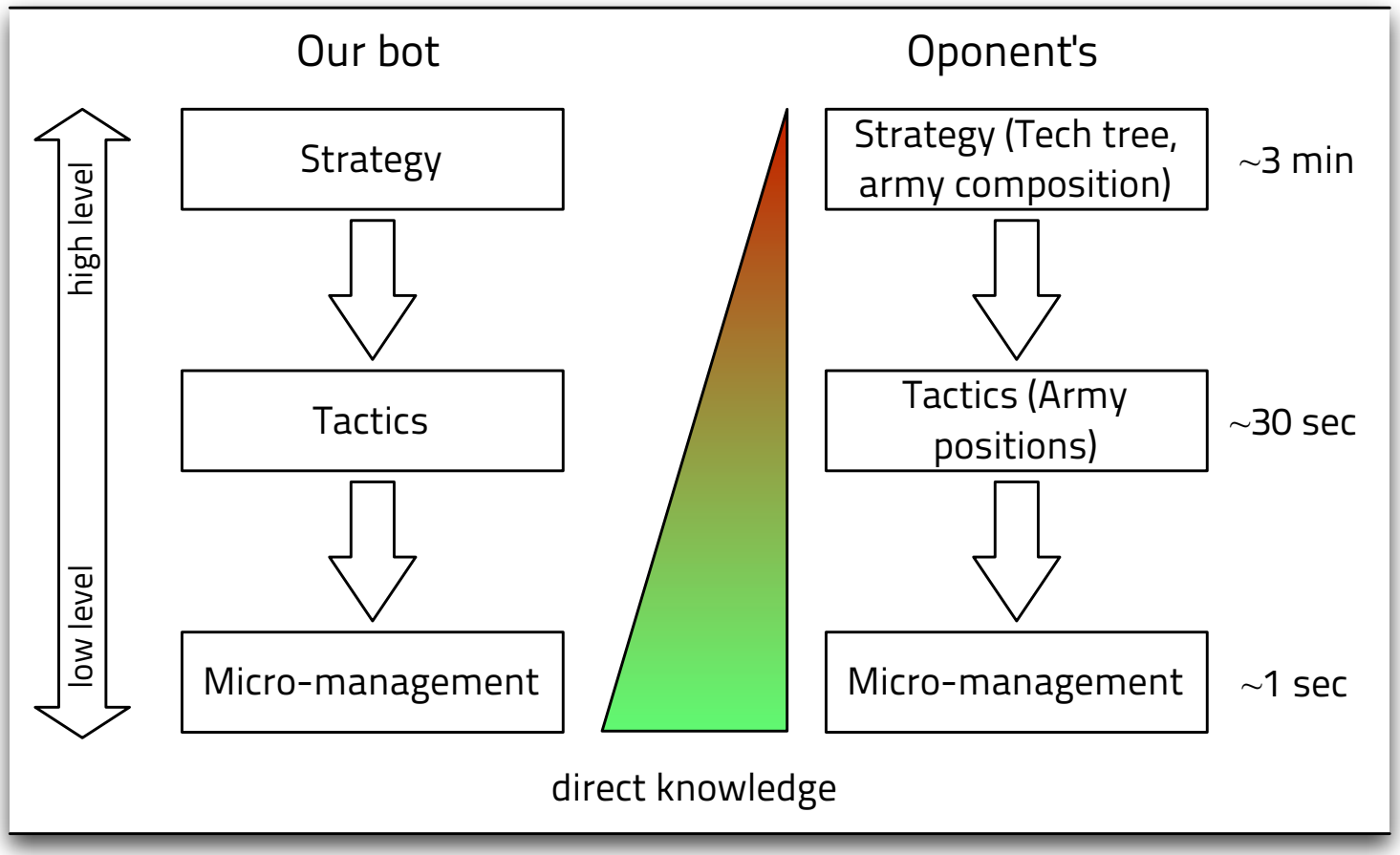
- Considering the units as individual Bayesian robots
- Seen units (viewed units filter)
- Probabilistic inference, machine learning from *replays*

Machine learnings*s*

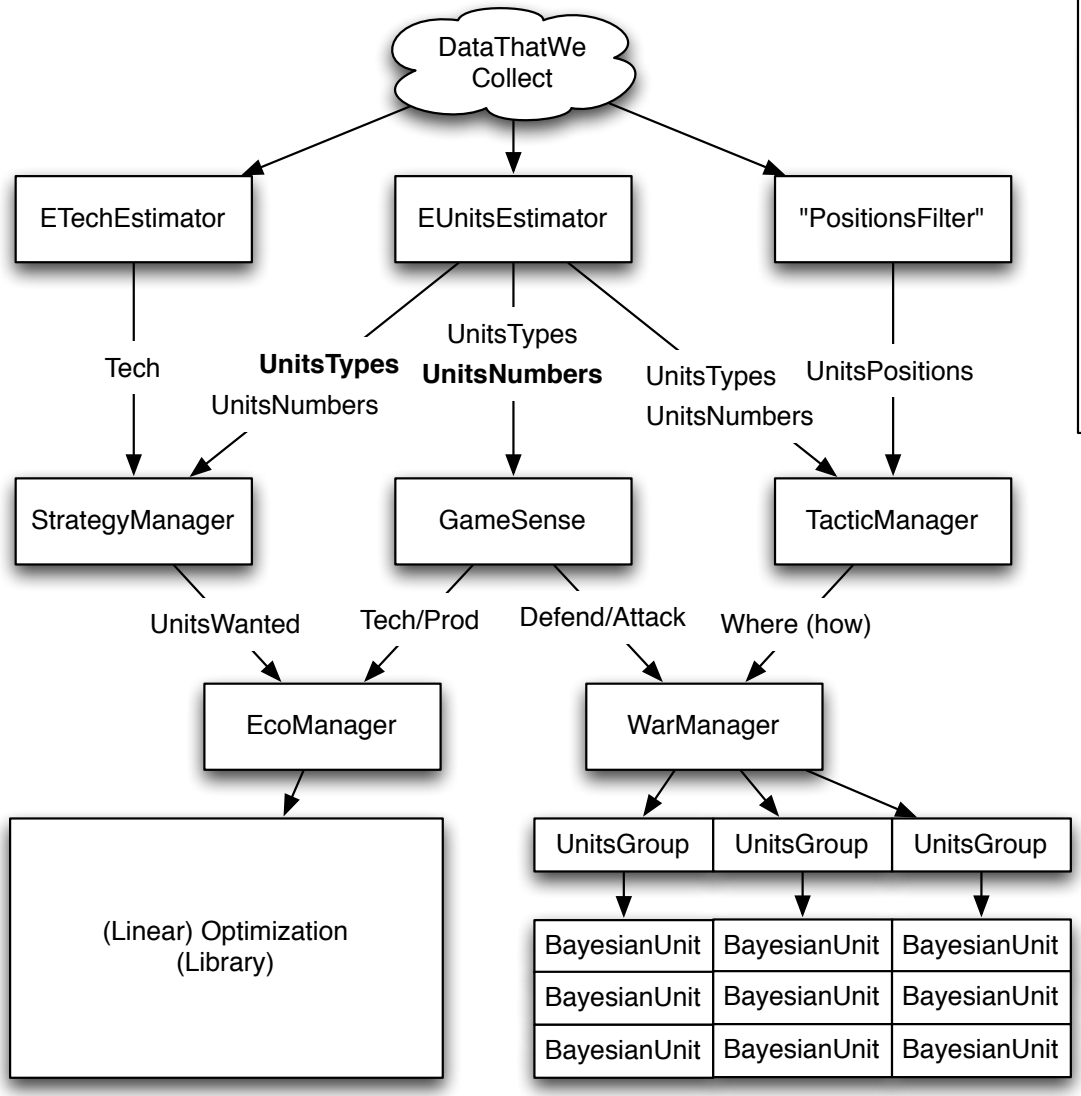
- from replays (parameters of predictive models)
- reinforcement (exploration of parameters space for the Bayesian robots)
- online (adapt to particular opponent)

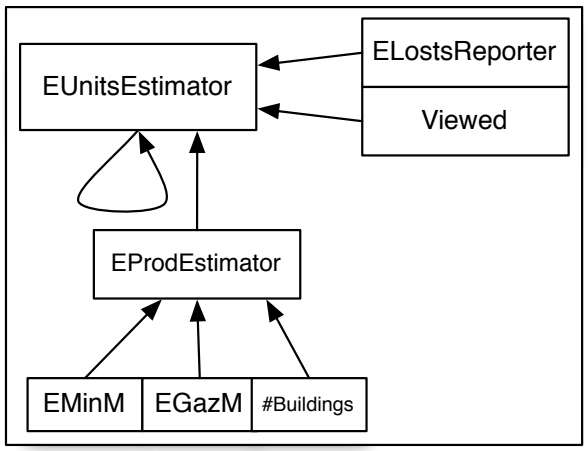


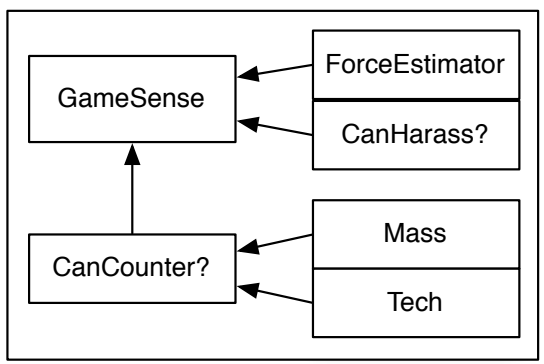
Problems to tackle



Model overview







The Bayesian unit (1)

The workers are *not* controlled as (Bayesian) sensory motor systems as their tasks are extremelly simple.

Modal unit, can be simplified as:

- Scout
- Move (small / large group)
- In position
- Fight:
  - Attack
  - Fight-move
  - Flee

} When not attacking

The Bayesian unit (2)

Mode dependant influences, as:

- “statically” occupied tiles
- “dynamically” occupied tiles
- height
- damage map (+ gradient)
- pathfinder
- ...

The Bayesian unit, Fight-move example (1)

Variables:

- $Dir_{i \in \{\text{possible directions}\}} \in \{T, F\}, n = \#\{\text{possible directions}\}$
- $Obj_i \in \{T, F\}$,
- $Dam_i \in \{no, low, med, high\}$,
- $Rep_i \in \{no, low, high\}$,
- $H_i \in \{0 \dots 3\}$,
- $Occ_i \in \{no, terrain, building\}$

Decomposition:

$$\begin{aligned} P(Dir_{1:n}, Obj_{1:n}, Dam_{1:n} Rep_{1:n}, H_{1:n}, Occ_{1:n}) &= \Pi_{i=1}^{n} P(Obj_i | Dir_i) \\ &\quad .P(Dam_i | Dir_i) \\ &\quad .P(Rep_i | Dir_i) \\ &\quad .P(H_i | Dir_i) \\ &\quad .P(Occ_i | Dir_i) \end{aligned}$$

The Bayesian unit, Fight-move example (2)

Parameters:

They are hand-specified for the moment but should be learned through task-specific maps and maximization.

Questions:

The pathfinder gives us $Obj_{1:n}$, the damage map $Dam_{1:n}$, the distance to and size of other units $Rep_{1:n}$, etc.

$$P(Dir_{1:n}|Obj_{1:n}, Dam1:n, Rep_{1:n}, H_{1:n}, Occ_{1:n})$$

Bayesian flocking (simplified)

Variables:

- Dir_i
- Obj_i
- Rep_i

Parameters:

$P(Rep_i|Dir_i)$ are learned with maze-like maps and on the objective to minimize the time-to-completion.

Obj_i is not well-defined: *We have to divise its value by the number of units flocking together*.

Bayesian repulsion (flocking applied to “in position”)

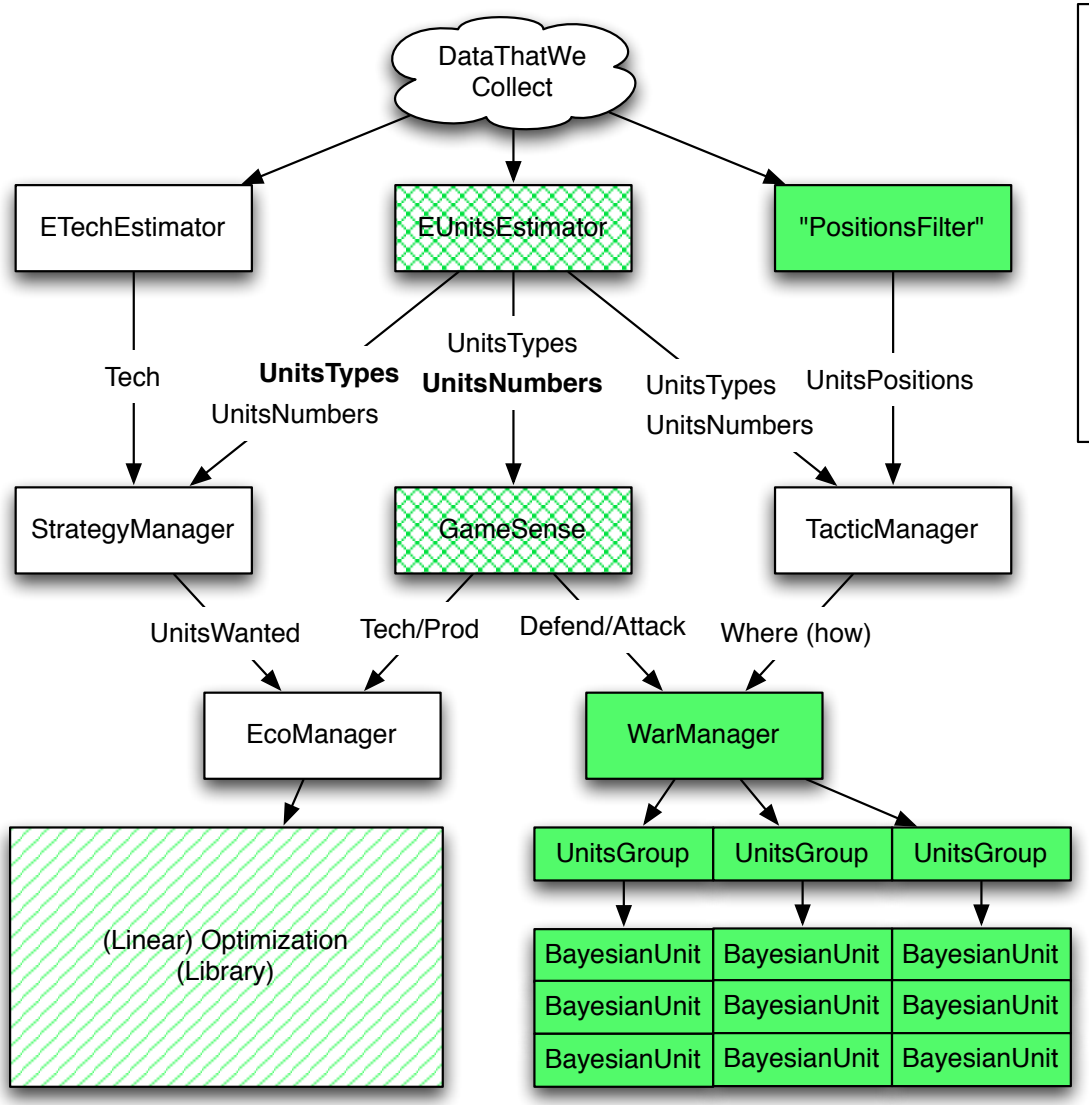
(2 youtube videos
`http://www.youtube.com/snippyhollow#p/a/u/1/sMyF_PlDqFo`
`http://www.youtube.com/snippyhollow#p/a/u/2/mvv9kUntLHU`)
Like flocking but with different parameters (and an additional attractor: the position).

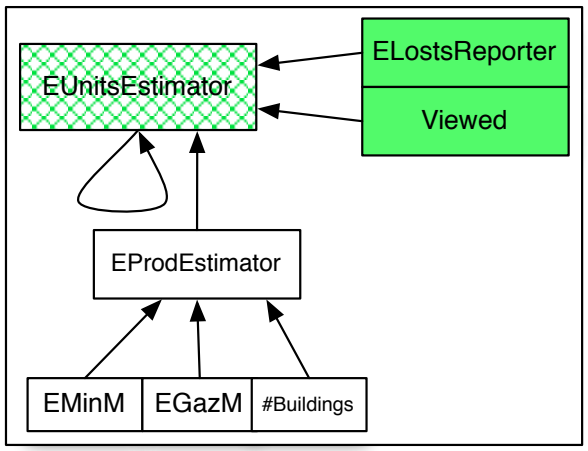
BWAPI

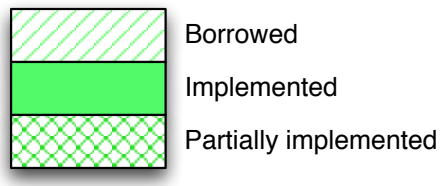
BroodWar API

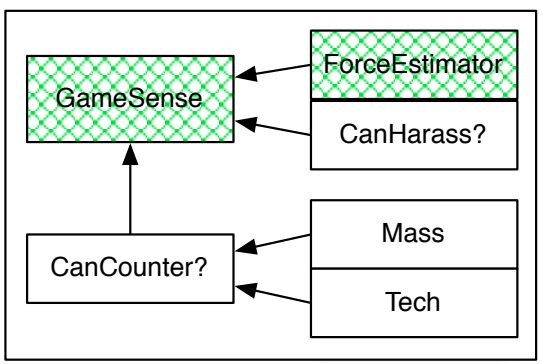
```
#include <BWAPI.h>

.onFrame()
{
    units = Broodwar->getAllUnits();
    for (set<Unit*>::const_iterator it = units.begin();
            it != units.end(); ++it)
    {
        if ((*it)->isWorker())
            continue;
        if ((*it)->getType() == UnitType::Protoss_Zealot)
            (*it)->attackMove(enemyPosition);
        else
            (*it)->attackUnit(enemyUnit);
    }
}
```

What has been implemented?









Interesting snags

- Hidden walkable pixels
- Stuck units
- Dealing with lag

To-do code work

Currently:

- Key tests (no “unit” test)
- Synchronous (1/24th second) + threads (pathfinding, storms coverage, planning)

To-do:

- Automate tests (94 test maps)
- Switch to an event-driven paradigm/framework (19498 true lines of C++)
- Add tests
- Implement remaining/new models
- Integrate learnings

Tournament 1

- 7 participants
- Took part, got left out ☹
- FreSC (Epita team) won
- Played against FreSC: draw! (0-3, 3-0, 6-8)

Tournaments 2 & 3

Tournament 2:

- 2 participants
- Didn’t took part
- FreSC won

Tournament 3:

- 8 participants
- Didn’t took part
- Mimicbot won

Tournament 4

- 17 participants
- Took part, really ☺, 0-3, 3-0, 1-3 passed one round only (“top 8”).
- Eliminated because of a (lame) strategy not handled correctly (bug)
- Overmind (UC Berkeley team) won

Team Overmind:

- 13 members (Ph.D students)
- 1 member took part in WCG 2001
- `http://overmind.cs.berkeley.edu/`

Half-satisfying results

- Not all have been implemented ☹
- What have been implemented performed well ☺
- ... after tweaks & hacks ☹
- We keep on working on Starcraft ☺

Bibliography


A Data Mining Approach to Strategy Prediction (2009) [Weber B. & Mateas M.]


Bayesian Programming and Hierarchical Learning in Robotics (2000) [Diard J. & Lebeltel O.]


It knows what you're going to do (2001) [Laird J.E.]


Bayesian Robot Programming (2004) [Lebeltel O. & al.]


Teaching Bayesian Behaviours to Video Game Characters (2004) [Le Hy R. & al.]


Probability Theory: The Logic of Science (2003) [Jaynes E.T.]

Thanks

Thank you for your attention.