Multiple AI types in FreeCiv

Krzysztof Danielowski
Przemysław Syktus

Project objectives
Our project’s main goal is to add different and distinguishable AI types to FreeCiv. We wanted to create a possibility to modify some of artificially controlled opponent behaviour, while not rewriting the core algorithms. This project was a little different from others that were being developed during laboratory, since it did not focus on improving overall AI performance against players, but rather on applying some variety to its actions.

This short document will present our ideas for AI types in FreeCiv and how we implemented them.

Different approaches for modifying the FreeCiv code
Structure of FreeCiv code is in fact prepared for creating various AI opponent types, but the version we have worked on had only default one implemented. We tried two different approaches on modification on such a large application as FreeCiv, one that was designed by creators of current code and a more invasive but simpler one.

During our first attempt to modify AI we tried to follow its creators’ path for our task and, despite some difficulties, we have succeeded. Our first AI, called “dumb” was an attempt to create simplest AI type possible. We will elaborate on results of AI later on. Recommended by creators approach (which, note, has not been tested in development by them as there is only one AI type) is to substitute a structure with pointers to functions controlling whole of AI’s brain, adding new functions. We decided that most hermetic way is to create a separate module for these functions and make structure point to them. Our very simple change in AI (not to build settlers) did not require much intervention in code, but due to heavy use of static functions in code, a lot of it had to be copied into new module we added. While this approach is advised by creators we found it much too complicated to use, and requiring big amount of work. Thus, we decided to take a simpler path which resulted in much quicker progress of project.

Due to already having prepared a user’s interface for choosing our AI in first approach (more on that later) we attempted a different approach on our second try. This time, while we did substitute structure with function pointers (conveniently called “ai”), we did not replace actual functions, but modified them to include our changes based on address of “ai” structure for player. This method quickly turned out to be simple, yet it did not throw our code in such disarray as the first one. Also, time we needed to create our next simple AI type called “widespread” was much shorter.

For reasons mentioned we decided to continue with second method. If our changes are to be contributed to code they might one day be rewritten into proper form, but in our opinion that would require big reorganization of AI code structure in order to avoid pointless repetitions of static functions in different modules.
Consistent types against traits
When we started our project we read about AI traits on FreeCiv Wiki pages. Those traits would modify AI’s behaviour and different combinations of those would result in different AI types. Traits suggested by Wiki pages are listed below, with short comment:

- "Isolationist" vs "Imperialist" - the peaceful/warlike nature of the player
- "Barbaric" vs "Civilized" - the use of ‘bad’ units, governments and buildings (eg nukes); backstabbing
- "Xenophobe" vs "Diplomat" - how often the AI will do diplomacy, and chance of accepting
- "Opportunist" vs "Ideologue" - will choose enemies based on profit or personality mismatch
- "Industrialist" vs "Expansionist" - build big cities or many smaller cities

Those traits were supposed to be binary at first (either one or other type), and then get scaled values expanding possibilities of AI combinations exponentially. After giving it some thought we decided that consistent AI types (not defined by parameters, but with whole behaviour set specified), with more specific and focused behaviour will be more appropriate to show distinguishable difference in how chosen AI type works.

We have also noticed that some of suggested traits would not work together very well, or even at all. For example Isolationist and Diplomat cannot really be chosen together, so implementation of those traits would have to be wary of that. More subtle contradiction exists between Isolationist and Expansionist – AI would soon run out of place to settle for new cities (with smaller cities wanted).

Last reason we chose to abandon idea of traits were more advanced AI tweaks, like when to construct war units (more on that in “aggressive ai” description).

User’s interface for AI type choice
Because our project was meant to concentrate on AI rather than game itself we opted for simplest solution for AI type choice interface, that is server commands. Using our modifications requires user to run a FreeCiv server, and connect to it. Server command line is required to change AI type, either before game starts or while it is in progress. Server commands are simple and named after AI type.

General use of those commands is as follows

<aitypename> <playername>

Or to change all players at once

<aitypename>

Note that change will also affect human players, with obviously no result, as the AI does not control their movements. However if such player was “AI Toggled”, AI type change made while nation was controlled by a player would still be valid.

Source: [http://freeciv.wikia.com/wiki/AI_personality_traits](http://freeciv.wikia.com/wiki/AI_personality_traits)
Implementation details - commands
Commands are implemented as single command function which recognizes the command that was issued by its ID. Different AI types do not require different method to change into, that is why this simplification was done.

AI types specification

„Dumb” AI
First task we wanted to perform was to create an AI that does nothing, in order to test our coding method. However while reading the code that was already done, we decided to create an AI that does everything as the regular one, but uses strategy that is essentially flawed in terms of game mechanics and poses no threat to opponents whatsoever. Our choice was to remove ability to construct settlers (unit used to build new cities) and therefore ability to develop as civilisation.

Results were somewhat close to our predictions, yet not without surprise. Two cities have been found, and they started to produce improvements, some military units and “Wealth” (which turns city production into gold). Due to AI’s civilisation obvious weakness, a part controlling diplomatic relationships was reluctant to engage in conflict with neighbours, noticing an obvious difference in power. All of our simulations came to a point, when AI engaged in war but it was not clear whether it was due to “dumb” or regular AI action (we suspect the latter). To our astonishment our “dumb” player, which was not supposed to stay alive for more than couple turns of combat turned out to be a very good defender and survived attacks of massive waves of enemies, and in one of simulations even took the fight back to the enemy and captured a city. In the end however, technological superiority of opponents was enough to break heavy resistance of our AI.

We analysed early development of “dumb” AI cities in order to find reason for such resilient defence. Our conclusion was, that due to lack of wasted resources and population on settlers (producing a settler subtracts one population point off city) resulted in quickly developed cities with large amount of improvements and bigger garrison. Bigger garrison was a direct result of greater population for AI’s tactic to prevent riots and unhappiness by garrisoning more troops. Also default garrisoning algorithm makes number of troop a function of city’s population. Good defence was also a result of not having to move troops very far from one city to another in order to send reinforcements.

To sum up, “dumb” AI, created with very simple change to AI itself (most coding for this involved commands and things necessary to make it work at all) created a very interesting case to study and analyse. As mentioned before, this AI type was most certainly not created with intention of improving overall performance of computer controlled player and it always happened to lose the game against other AI types (however it’s resilience allowed it to outlast couple more sophisticated AIs in some simulations).
Screenshot 1.
Dumb AI having only 2 cities, note how big they are (average size of city at this point of game is bout 2, most cities being in range of 1-3 with one or 2 cities with population of 4 per civilisation.

**Dumb AI – implementation details**
Dumb AI was realised as an external AI module, as mentioned before. The changes made in order to prevent constructing settler was to reduce so called `founder_want` in function `domestic_advisor_choose_build` located in `advdomestic.c`. We just hard-coded it to 0 at the end of the function, so all calculations were made and then the need for settler was reset. There is also a parameter called `settler_want` in the very same function but despite the name it is not connected to settler construction, but rather to worker construction.

**Widespread AI**
Second AI type that we have developed was the widespread AI. The idea of this AI is to reduce number of cities in favour of stronger cities. We hoped, based on dumb AI’s success, that no longer producing settlers in such great number (producing a settler makes a city which produced it lose 1 population point) would greatly improve the city quality. Furthermore, our cities were spread further from each other, preventing them of stealing each other’s resources. As cities have 2 square radius of which they can use resources, we made them be built at least 4 squares away.

We modified AI part responsible for settler itself, in order to make it ignore fields close enough to existing city as a potential field to find new one. While initially AI achieved its goal of creating country with fewer cities that were further from each other, after reaching contact with “normal” AI civilisations it became obvious that our AI’s development as a civilisation is somehow tampered. We began to look for reasons for that and found out that, while not finding as many cities, AI is still producing huge number of settlers, and those are limiting each city’s development. After a while our island was filled with useless settlers that could not find new place to create city, but were running in circles around.
Next modification we decided to make, in order to counter the overproduction of settlers, was obviously some change that would limit number of them produced. We came up with a limit of settlers to be two third of currently created cities. This way, we would not experience crowd of settlers appearing right after the start and slowing down city development. We have also limited total number of settlers to 15, because it is very possible that our AI will have more than those 25 cities it will have to get to produce more, and then it would still produce settlers that are of no use.

This modification helped a lot, our AI’s cities grew more rapidly than those of “normal” AI’s. Their defences were also better, as AI based number of troops stationed in cities on this city population. Our widespread AI was created in order to not allow cities to interfere with each other’s production, but we quickly noticed, that even in spite of our settler overproduction limitation cities do not grow fast enough for resource stealing to actually appear. We decided to lower distance between cities to 3 (down from 4), so AI could find more cities and bloom faster as a country.

The results of mentioned changes were very satisfying and while it is hard to decide whether they are an overall improvement we can surely say that they change the style AI plays. We believe that stronger defended cities will result in human player having more fun and challenge while trying to conquer such civilisation. The drawback of this AI, smaller number of cities, is allowing players to somehow counter the stronger cities. Regular AIs have some trouble with conquering widespread one, but if their initial situation is better (for example starting on bigger island with more space to grow quickly) they still might get a fair chance to win with our modified one.

Screenshot 2.
The widespread AI. Note distances between cities as compared to next screenshot. Also cities are generally a little bigger. Further reduction and balancing number of produced settlers would lead to larger differences between this and regular AI city sizes.
Widespread AI – implementation details

As mentioned before we tampered with individual settler’s AI which is located in aisettler.c file. This time we decided not to copy the files into new module, but to create new ai in structure ai for player that is exactly the same as regular one, but we can use it as a basis of comparison to tell whether we want to use modified strategies or not. In the aisettler.c file we have modified the function called city_desirability which is used while iterating over each field in settler’s range to decide which field is available for new city creation. In fact, we tell settler to consider place as “reserved” when iteration square is too close to city.

Second change is, as with dumb AI, in file advdomestic.c, function domestic_advisor_choose_build. However, while it was copied to separate module last time, it is just changed in its original location this time, adding a conditional statement that decides on action based on AI type chosen. As before, founder_want parameter is lowered to 0, this time, when number of settlers exceeds given limitations.

Aggressive AI

Third type of our AI was designed as an aggressive player, picking up fight with everyone. This time our main focus was in relationship with another players and not on building. However, as we will mention later, we had to adjust those parameters as well.

Computer controlled player chooses its behaviour towards another player, both computer or player controlled, based on parameter called “love”. The higher the “love” toward another player is, the better relationship will given AI player have with another one. There are many things that can affect
“love”, like attacking the player, attacking allied player, giving gift through diplomacy, having military units enter player’s territory, refusing to comply with player’s wishes and many more.

Basically, we did not modify those reactions. We noticed that when AI player does not have any target to attack, he will choose one, and subtract “love” from him. We made this loss of “love” multiplied by our parameter we also used in couple different spots, that depicts aggressiveness of aggressive AI type.

We also made AI less afraid of opponents, reducing fear parameter by same factor we used to increase “love” loss at. We decided that if a player is aggressive, he should also be fearless.

Last chance concerning “love” parameter is the way “love” is forgotten. Usually, it is gravitating towards 0, with some coefficient, but the basic relation is: the further from 0 current “love” value is the faster it will be pulled towards it. We changed it in a way, that when “love” is positive it will degenerate faster (by our aggressiveness parameter) and when it is negative it raises slower (also by aggressiveness parameter). This way when our AI starts to hate someone it will hold grudge for longer than regular AI, which should lead to wars lasting longer. On the other hand if our AI likes someone it will get bored of peace quickly and initiate war.

There are also couple changes not covering “love” parameter and not connected to likelihood of war itself, but to behaviour connected with fighting. First thing we changed was number of turns after meeting player for first time that’s needed to start war with him. Regular AI gives player a benefit of doubt for first couple turns and will not attack him. Our AI does not. It usually initiates combat preparation as soon as they meet player.

We ran a couple simulations after adding those changes and while our AI was indeed aggressive and picked fights with everyone, it lacked ability to win all those fights, especially after some time when forced to fight on many fronts. We decided to modify war behaviour of this AI in order to achieve better combat ability at a cost of overall development.

We noticed, that before AI involves in a war it takes some time for preparation. During that preparation it constructs troops which will be used in an attack. We stretched this phase in order to allow our AI to create more troops before commencing an attack. We had however to not make this phase too long in order for our AI to remain aggressive and not get constantly involved in preparation instead of actual fighting.

We also changed amount of time after breaking into a war that is required for AI to be accepting treaties. Our aggressive AI is also stubborn when it starts a war.

Last change was probably the most important one. Regular AI does only produce troops that will attack enemy during preparation phase, and later only those that are supposed to defend home cities. Attack troops are also created later, but AI considers itself to not be engaged in war anymore, thus focusing on economy and defence. This tactic is based on idea of signing treaty after some time of war due to war weariness. Our AI prolongs wars and is engaged into many of those, so we had it make combat units also during the war. This behaviour, while essential to wage war successfully is very harmful for civilisation development. Without it, however, our AI tended to quickly lose due to many wars it couldn’t win decisively.
Our simulations of this AI clearly showed that its aggressiveness is indeed working very well, AI being soon involved in war with everyone. In the first phase of game it is very nice behaviour and it looks like this AI will win swiftly. However, after more turns it other AIs meet each other allying with them. This leads to our aggressive AI fighting on many fronts with allied opponents, which obviously leads to its demise at some point. Ability to produce troops while fighting gets very useful, but after a while it impacts economy and science enough to make this AI lose the wars. Its stubbornness makes it not retract from war, so it loses even more on the science and economy part.

To sum up, aggressive AI is a very interesting opponent for human player, especially if it is the first one to meet. It fights fiercer and longer during its first wars and might be a challenge to counter at first. We believe this sort of challenge would result in increased game enjoyment for a human player. On the other hand due to mentioned factors this AI becomes weaker as time progresses, usually becoming of no real threat after some time, and losing to opponents of other AI types.

**Aggressive AI – implementation details**

 Modifications to this AI are made mostly in a file `advdiplomacy.c` which contains functions responsible for relationships with other players. We added `FC_AGGR_MOD`, which serves as a modifier for “love” loss, fear and other parameters. Significant changes were also done in function `war_countdown`, responsible for countdown till engaging in a war.

For reasons mentioned above we had to modify way of choosing build, those functions are once again located in `advdomestic`, responsible for city constructions. We were considering some changes to military advisor to create units more focused on attack, but those are somewhat complicated mechanisms due to changes in technology and changes in units they produce. Effects would be small so we decided to not apply those changes.

**Peaceful AI**

Our final AI type is a peaceful AI. It was at first meant to be a simple modification of aggressive AI, with reversed action, but turned out to be a little more complicated. This AI was made in attempt of creating an AI that has a useful yet visible change that will not hurt it on long run. We started by reversing actions taken by an AI concerning “love”, but it turned out not working right. Reason for it was that our main threat generation in aggressive AI was based on selecting target to attack while there was none. This approach is obviously not very effective in peaceful AI, as it would like to limit its offensive actions. Changes we inherited from aggressive AI (in the reverse of course) were those made to “love” gravitating towards 0, in this case if the “love” is positive it stays high longer, and if its negative it will raise quicker.

The changes we made in peaceful AI was to limit negative impact on “love” parameter of hostile actions taken by opponents. For example, attacking ally or keeping military units in our territory will not lead to a war as quickly as usual. The action connected to attacking player if there is no one to attack was lowered in order to prolong state of peace. We also made the peaceful AI fear its opponents a little more.

All those parameters were once again connected by a single value similar to aggressiveness modifier in aggressive AI type.
Another change was to shorten amount of time required to pass after declaring a war before it was possible to stop waging war and make a treaty. Also number of turns after meeting a player and before considering attacking him.

We considered modifying types of units produced in city, to be more defensive, but the code for that was quite complex and we decided that an AI is not producing many offensive units anyway apart for phase of war preparation, so we just decided to shorten that period. This resulted in a peaceful AI to sometimes declare a war before aggressive AI does, due to much shorter period from deciding on a war to engaging in one. However, peaceful AI ended his wars quickly, while an aggressive AI stays in a war for a very long time, possibly till the end of the game.

We ran couple simulations with this AI as well, but it is not very distinct from regular one. We believe that the changes might not be noticed by a player immediately if he does not know exactly what to expect. Furthermore, we found this AI type to be rather boring, due its lack of activity. On the other hand, unless conquest is the only possible solution this AI is quite a good player compared to the aggressive one as it does not slow down its progress by constructing many unnecessary offensive units during wars at a cost of overall development. However, this improvements alone (being peaceful) will not make an AI win more, because conquest is also a viable way to enlarge your empire, even more so when there is no more space to develop peacefully.

**Peaceful AI – implementation details**
As with the aggressive AI most changes were made in *aidiplomacy.c* file, which is responsible for diplomacy interactions and “love”. FC_PEACE_MOD constant was created as an opposition to FC_AGGR_MOD and applied to the relevant parameters ("love", fear).

No changes in construction schedule were made, in distinction from the aggressive AI.

**Simulation results**
At the laboratory we were asked to present some simulation results, possibly in form of charts of scores. However, as mentioned before, our AI types do not really improve overall performance in game, and results are rather unsatisfying when considering it on win/loss basis.

As our guest lecturer on the module mentioned, the most important thing about AI is to give enjoyment to player, and not to win. He even noticed that human players like to win, and making it more difficult or impossible for them will lead only to their abandoning the game. That is why we believe our changes impact the game in a positive way. While our AIs are not winning, sometimes performing even worse than the regular ones (apart from widespread, which is actually pretty good), we think that adding diversity to computer controlled opponents creates more challenge for player while not making game much more difficult. If there is only one AI type then player can devise a single strategy that works against it (in this case that might be creating enormous stack of units and plundering cities as there are many but weakly defended) and keep using it every time in order to win.

With AI types single strategy is less effective, because of cases like widespread AI that forces you to attack better defended cities or aggressive AI that might attack with fury in the very beginning of
game forcing a defensive strategy in order to survive or not lose too many cities. In our opinion this diversity makes the game much more enjoyable.

To be fair there are also drawbacks of our AI types limiting fun the game creates, that is the peaceful AI which makes the game more boring at is requires player to take action or dumb AI which while in general easy to defeat and making almost no threat at all might be annoying with its resilience. Please note that dumb AI was not designed for actual play, so this is somehow justified.

**Observations of each AI type game play**

As mentioned before there is a requirement for some results of simulation, and to make up for not having the numeric values we decided to write a short paragraph about each of AI types and our observations as a player. These observations were mostly mentioned earlier in detailed description of each AI.

**Dumb AI**

Creates two cities, both of the grow quickly. Soon gets surrounded by an enemy, but not engages in a war due to high fear of larger civilisation. In war turns out to be very resilient with ability to quickly produce defensive units and deploy them to the city that is attacked at the moment. Effective strategy is to divide cities with own forces or to station units around them to occupy terrain and not allow quick defence production.

This AI type is very likely to lose in every game without having more impact than slowing down one of player’s progress.

**Widespread AI**

Creates many cities, but further from each other than the regular AI. Also makes less settlers. The result are more developed cities, with better defence. In general turns out to be at similar level of difficulty as regular AI, but forces player to change strategy to not attack random city but to choose ones that are defended worse than the others. In the regular AI there are many cities and none of them really that much more powerful than other except for maybe wonder city.

Effective strategy is similar to regular AI, but with wiser target choice. Cities that are heavily guarded should not be attacked until their defences are lowered by them helping other cities. This AI is much more able to construct good wonder city as its growth is not that much limited by surrounding cities taking its spots for production. Therefore it is able to actually construct some important wonders before other players do.

**Aggressive AI**

Usually attacks at first sight, after preparation period. Creates large amount of units in the beginning and keeps making more even during war. Is very reluctant about treaties, usually ending up in war with everyone else, which results in its eventual demise. If encountered as a first civilisation might get problematic with its fierce attack which may lead to player using some of his cities, which most players do not like.

Effective strategy involves heavily protected cities close to border to defend against attacks, and attacking the weaker defended cities after initial large wave of units is defeated. Interesting opponent at a start but gets very underpowered later with slow technology development due to lack of diplomacy and trade.
**Peaceful AI**
Not very different from regular AI. Happens to be easier to trade with (technologies, even cities) and quite difficult to annoy. Attacking his allies does not always lead to it declaring war, and may result in breaking treaties rather than helping an ally (traitor AI type). Tries to end war quickly with treaties as his hate dissipates quickly.

Effective strategy is to simply not engage in war with this one as it is unlikely to attack player even threatened. If surrounded, it will lose its ability to effectively develop, making it an easy opponent to finish off at the end of a game.

**Summary**
We consider our project a success, as we managed to create distinct AI types with different feeling to their play style. Our AIs were not the smartest on the long run but they were fun to play with and some of them created different type of challenge than the regular AI. Also we have noticed some unexpected and interesting effects of changing AI strategy while not changing many of its core algorithms.

As for releasing patch, we created one with diff tool, but due to our method of coding (using two different types of changing AI code – modules and conditional statements) we think that our code would require refining before being posted to FreeCiv developers in order to merge with the main source. If that would be a requirement of a grade we can post it anyway.