300/303COM Project Portfolio

Full Name: Martyn Ian David Patrick

SID: 6457291

Degree Title: Computer Science

Institution: Department Of Computing, Coventry University

Supervisor: Dr Simon Billings

Statement Of Originality

I Declare that This project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet etc.) has been acknowledged by citation within the main report to an item in the References or Bibliography lists. I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

Statement of copyright

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to Coventry University. Support, including funding, is available to commercialise products and services developed by staff and students. Any revenue that is generated is split with the inventor/s of the product or service. For further information please see www.coventry.ac.uk/ipr or contact ipr@coventry.ac.uk.

Statement of ethical engagement

I declare that a proposal for this project has been submitted to the Coventry University ethics monitoring website (https://ethics.coventry.ac.uk/) and that the application number is listed below (Note: Projects without an ethical application number will be rejected for marking)

Signed: Martyn Patrick Date: 30/04/2018

First Name:	Martyn
Last Name:	Patrick
Student ID number	6457291
Ethics Application Number	P68021
1st Supervisor Name	Simon Billings
2nd Supervisor Name	Reda Al Bodour

Certificate of Ethical Approval

Applicant:

Martyn Patrick

Project Title:

An examination of the Artificial Intelligence (AI) in Civilization 5 and how it could be improved

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Low Risk

Date of approval:

28 March 2018

Project Reference Number:

P68021

0. Abstract:

This project aims to improve Civilization V's AI 'expansionist' gameplay in the Ancient and Classical eras while treating the AI as a human player. In order to accomplish this modifications have be made for the game these modifications are: a modification to remove the AI bonuses, a modified version of an existing playable Civilization, two modified versions of the AI and two maps that provide a constant environment for testing the AI. To conclude if the project had been a success four versions of the AI were tested: unchanged AI, unchanged AI treated like a player, AI V1 and AI V2. Each AI was tested 5 times on each map up to turn 150. Statistical data like citizens and amount of tiles as well as non statistical data like the research the AI had completed were taken. As a result of the research conducted it was concluded that the AI in its current state, cannot effectively grow and develop an empire in the Classical and Ancient eras while being treated like a human player reliably. It was also found that the modifications made to the AI did not contribute positively to the AI's performance and that more extensive improvements would be required in order for this to change.

Keywords: AI; Civilization 5; Mod; Game Design; Video Games

0.1 Acknowledgements:

I would like to thank my supervisor Dr Simon Billings, for their valuable feedback and guidance.

1. Introduction:

Unlike traditional physical games, computer games (henceforth referred to as games) can better support individual play by providing an artificial opponent to play against often referred to as Artificial Intelligence (henceforth referred to as AI.) Games that use AI heavily are often made and broken on how 'intelligent' the AI appears to be. This project looks at how the AI is used as well as looking at some ways to improve the AI in a game called Civilization V, a 4X strategy game where players are tasked with growing an empire throughout human history. Civilization V is a 4X game because 4X describes some of the main themes of the game: eXplore, eXpand, eXploit and eXterminate (Emrich 1993). Civilization V is a very complicated game under the surface as it has a lot of different components and mechanics to learn that all interact with each other. The AI in this game is regarded as one of the weakest components of the game. The AI is often criticized for irrational decision making, in addition on higher difficulties the AI receives large bonuses to compensate for the lack of 'intelligence.' Which causes players to think the AI is 'cheating' by creating units faster than a player would be able to normally.

Research Question: Can an AI effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players? (See 1.1 for a clear breakdown of project aims.)

This research project looks at how the AI could be improved as if it was treated as a normal 'human player' and was given no special bonuses. The author ran an experiment on two maps. Running each of the following versions of AI:

- Unchanged AI - The game's default AI
- Unchanged AI treated as a human player - The game's default AI but with all AI bonuses removed
- AI V1 - The AI with some modifications that should increase the priority of workers
- AI V2 - As above with V1, but with more modifications that should increase the priority of settlers and increase the change that the AI will to choose to expand.

Each AI was run 5 times on each map, up to turn 150 which is around where a human player would begin to reach the end of the early game. Notes about what the AI was doing were taken and metrics that contribute heavily towards empire growth and expansion where of particular interest.

The early game was chosen as a focus because the game is very large, can have a lot of different strategies applied to it and the early game appears to be the area where the AI struggles most (See Literature Review). Empire growth and expansion were of particular interest as the game has too many aspects and trying to focus on them all would likely lead to nothing useful being gained. In the author's experience the AI appears inconsistent in this area as it either expands or doesn't.

This report, does use a lot of the jargon relevant to Civilization V and Strategy games as well as some terms within the realm of Computer Science. Readers of a technical background as

well as those who are familiar with Strategy games in particular turn based 4x strategy games will have an easier time understanding this report. However this is not necessary as readers may use the Glossary to identify and define words and terms. Any definition given in the glossary is the actual definition of that word or term that is considered true by this report.

This project might be of interest to those who are exploring the application of Artificial Intelligence applied to games. In addition to anyone developing a '4x' strategy game, as many of the same elements can be transferred over. This project will be of special interest to Firaxis Games, the studio that created Civilization 5, as they may be able to take some of the results of this project and use them for future games.

1.1. Project Aims:

The following need to be completed in order for the project to be successful:

- Create two maps where the experiment will be conduct.
- Create a modification to remove all bonuses given to the AI
- Create a modified version of an existing playable Civilization and removing all special traits or units the Civilization has access too.
- Iteratively create and test a modified version of the AI
- Run experiments to prove either Hypothesis:

Hypothesis:
An AI can effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players.
Null Hypothesis:
An AI cannot effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players.

The following was the research question the author initially set out to answer, By the end of the project the research question should be effectively answered:
Research Question: Can an AI effectively develop an empire during the ancient and classical era in civilization V while playing under the same rules as human players?

Ultimate Project Aim:
To produce an Artificial Intelligence, that is capable of growing and expanding an empire in the early game of Civilization V while being treated like a human player.

2. Literature Review:

AI in games has its foundation in chess and in the case of chess is considered a 'solved' problem however research is still ongoing in this area. This is because while we have chess programs capable of outperforming world champions these programs have a lack of learning capability and act like 'brute force searchers' (David et al 2014). A report which Omid David

co-authored discussed some of the challenges that chess programs faced (David et al 2014):

1. The ‘search space’ the total number of different combinations of positions is estimated to be 10^{46}this means that any method based on a exhaustive search is ineffective.
2. The search space is not unimodal and because of this problem they argue that many top performing chess programs evaluation function’s parameters are codependent on each other i.e. changing any number of the parameters produce wildly unpredictable results. As a result chess programs using ‘conventional’ means of learning are ‘hand tuned’ by the programmers themselves. The authors suggest that Genetic algorithms are a better fit for this type of problem.
3. The authors argue that the problem of tuning and learning is not well understood this is because all top chess programs are hand tuned by programmers which based their tuning on educated guessing and intuition. This suggests there are few if any other practical alternatives.

It is clear that there is still a great deal of work to be done in order to make ‘truly intelligent’ chess ‘AI.’ Not only do we need to make chess programs capable of learning more than moves that would result in a loss on the first turn but also chess programs that are tuned automatically. It is true that we have ‘solved’ AI chess as in we have solutions that work but that does not mean we have the most efficient solution nor do we have the most ‘perfect’ one. Like Civilisation 5 the AI works and can provide some challenge to the player especially when playing on higher difficulties but to provide a challenge it doesn’t ‘play by the rules.’ It is clear that there is still room for improvement for Civilizations 5 AI.

Interestingly, as part of an experiment run by Omid David et al, where footage of a specific grandmaster was used to train a chess program. They found that the chess program would only mimic the general ‘grandmaster’ style and not the style of a specific player if 1-ply searches are used (The program only looked for one move at a time and didn’t plan ahead) (David et al 2014).

Starcraft 2 has recently become one of the hubs of AI and machine learning research this is because it represents some very interesting challenges not only from a ‘strategic’ decision making element but also how base mechanics function. A report by Oriol Vinyals et al looked into some of the challenges that Starcraft 2 presented (Vinyals et al Unknown):

1. Starcraft 2 often features multiple players competing for multiple different types of resources all of which the AI has to take into account. In addition the AI has to manage sometimes hundreds of units as well as take into account each individual unit abilities.
2. Imperfect information is often the only information the AI has access to this is because the AI has to move a camera in order to observe the map. There is also a fog of war which prevents the AI from seeing areas of the map where it has no units present.
3. The number of legal actions varies as the AI progresses through a ‘research tree’ in addition to this the things the AI builds and researches early have an effect later in the game.

Unlike in Starcraft, where there are a limited number of starting positions but a vast number of potential actions, Civilization 5's starting positions are randomly generated. This means there is a immediate challenge in determining standardised effective early moves.
This is the reason that the author is focusing on the early game, up to the classical era, as early advantages snowball leading to greater advantages in the mid and late game. Also diplomacy, the management of military and religious units will be ignored in an effort to reduce the complexity to a manageable level for this project. If successful the result should be an AI that, approaches the early game of Civilization in a way comparable to that of a human player. While the AI may still need advantages to make it competitive against good human players it should feel more natural for a human player to play against.

Jon Shafer, The lead designer of Civilization 5, revisited the decisions he had made during the development of the game. He described the AI as "not as strong as it could be, shall we say." (Shafer 2013) going on to detail what he sees as one of the major flaws of Civ 5's AI, "The computer opponents were weighted towards a variety of possibilities, with a healthy serving of RNG (random number generator) on the side. This meant they floated from one "strategy" to another without any real cohesion behind those decisions" (Shafer 2013). He goes on to propose a possible solution to the issue "if you want a strong AI there are times when you need to force it to behave in very specific manner" (Shafer 2013).
It appears that the reason the AI has this flaw is because during development Shafer became distracted with the design of the AI rather than how it 'feels' in gameplay. "Unfortunately, my enjoyment of building caused me to fall in love with the design rather than its actual impact. I was very proud of my code. But it really wasn't very good" (Shafer 2013). This shows that building an AI that is designed well in theory might not make an AI that is an effective opponent and/or fun to play against. In the author's opinion the reason the AI is ineffective is mostly because of the AI's heavy reliance on RNG to determine what strategy to purse. The AI is not consistent which not only reduces how effective it is but also sometimes makes it frustrating to play against.

This inconsistency with the AI's behaviour is also due to the different AI component's not communicating with each other effectively. The following is a situation that occured to a player while playing against the AI. The player gets dragged into a war with Egypt due to a defensive pact with The Byzantines. After a few turns Egypt realises it's going to lose so it gives the player's civ a city in return for peace. The player accepts and attempts to trade the new city to other civilizations if they would declare war on india. None of the other civilizations want to declare war on india so the player trades the city to india in exchange for some resources. Immediately after receiving the city, India, burns the city to the ground, producing a lot of unhappiness for the civilization for a few turns and effectively wasting the resources India traded for the city (Hemulihemmo 2016).
What the author believes happened, is the Diplomacy AI component decided to make the trade, but to the CityManagementAI/EconomicsAI India had just conquered a new city. The way India works is they want to keep the number of cities they have low, so because the AI just 'conquered' this city those AI components want to get rid of the new city so it does.

Although admittedly it's not just the way the AI is programmed that is at fault, it is also the way the game plays. In an interview with Rock-Paper-Shotgun, Jon shafer described a problem that he believes the Civilization series of games has always had. "The early game is the interesting bit: what's the map like, what are my opportunities, what shall I focus on. And then it gets into more of a formulaic place from there" (Shafter 2017). Because of this the early game is significantly more complex this is because:

- The AI has an incomplete view of the map
- The AI doesn't know where it's opponents cities are.
- Decisions early on in the game like: what to build, what to research, what social policies to pick. Affect the amount of choice available and wrong decisions can have an affect on performance late game.
- The amount of decisions that have an impact are higher in the early game than in the late game.

These issues make it harder to program an AI for the game that is both effective and fun without directly telling it how it should behave. The AI is more vulnerable in the early game as it is more likely to make an incorrect choice that will have a late game impact on later performance. If the AI was told exactly how it should behave in a given situation this wouldn't be as much of an issue. However it would make the AI way more predictable and would ultimately mean more development time as an AI would likely have to be coded for each Civilization or atleast segments would have to be. Another solution is to perhaps have the AI 'build' a strategy as it plays reacting to what's happening around it with perhaps some sort of 'offset' or partial strategy applied based on what Civilization is being played.

It is possible that Machine Learning could hold the answer, advances in Reinforcement Learning have seen some level of success with starcraft 2. Where the AI agents was able to successfully complete several 'minigames' that focused on different areas of the game such as exploration and camera movement. However none of the agents developed a viable strategy for a full 1v1 game, although interestingly one of the agents managed to avoid consecutive losses by using the terran, a starcraft playable faction, ability to pick up and move buildings out of attack range (Vinyals et al unknown). Although Reinforcement Learning likely will not produce an solution to the problem on its own. As both Glen Robertson and Ian Watson note, that in complex environments Reinforcement Learning requires "clever state abstraction mechanisms in order to learn effectively" and that Reinforcement learning is really suited for long term strategic decision making "because of the huge problem space and delayed reward inherent in strategic decisions" (Robertson et al 2014).

Kittisak Potisartra proposed an interesting solution, where the AI constantly evaluated it's opponent's skill level and then played 'worse' or 'better' moves depending on perceived skill level of the player (Potisartra et al 2009). However, this approach doesn't entirely fit with Civilization's gameplay as the game is a lot less about unit movement and deciding what to attack and more about deciding what to build, what to research, what tiles to improve and when to expand. Which all fit into more of the 'strategic' long game rather than the tactical short game. However it is possible to incorporate this if the AI was limited to certain 'strategies' at different difficulties, for example a normal difficulty AI is 'less intelligent' than a hard Difficulty AI. Which would take a lot more work but is certainly an interesting idea to

entertain that a higher difficulty is not challenging because the game rules have changed, but the opponents are literally ‘better.’

AI’s have always had trouble understanding context which is why they tend to suffer when it comes to opportunities and exploiting weaknesses. For example in AI Craft E3 which was a AI VS AI starcraft 2 game that was cast by Life’s A Glitch TV. After destroying all AI 2’s units and carving a path into AI 2’s base, AI 1 decided to retreat even though it had a clear opportunity to win the game (LAGTV 2013). It is clear in this situation a human player would have just finished the game unless the human player wanted to ‘toy’ with their opponent. But this kind of strategy is only effective if you’re playing multiple games against the same opponent in the same sitting and it’s clear this is not something the AI considered when choosing to retreat.

As Ed Welch a game developer with 10 years of experience points out, the point of a AI in a game, is not necessarily to beat the player. It is there to give the illusion of intelligence and to provide a challenge (Welch 2007). Does the Civilization 5 AI accomplish this? Not really, for the first couple games maybe, it is likely the AI is going to do something at least questionable even to a novice player every game. Arguably, AI is a core component of the Civilization series. Each game can often take quite a long time to complete, taking anywhere from between 1 - 100 hour(s) to complete (depending on the speed.) Because of this, it can be quite tricky to get multiple players together for a game, meaning it is likely some of the slots are going to be filled with AI. It is an understandable argument that the AI should be at least capable of posing a threat to player, which the AI is only capable of, on the highest difficulty due to the massive bonuses they get.

3. Project Execution:

This section will explore the Execution of my project and my findings as a result.

3.1 Project Methodology:

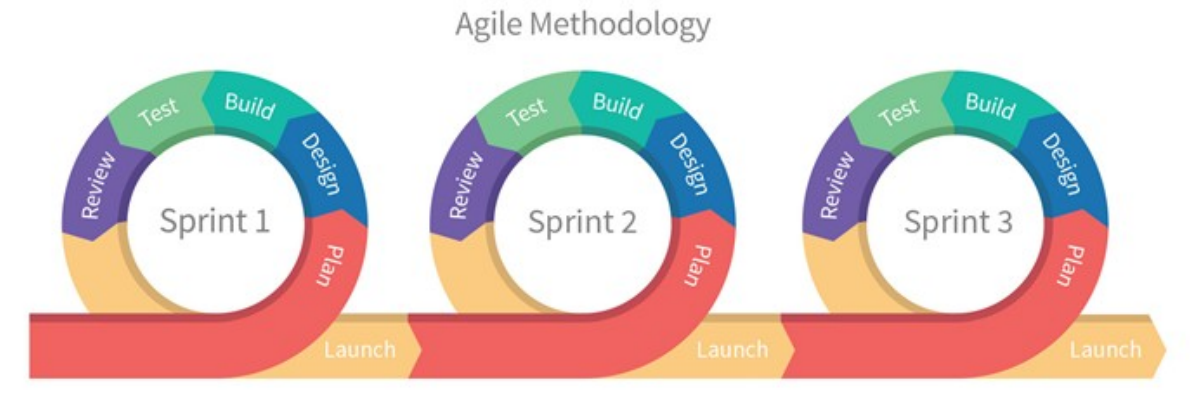


Figure A: An example of the agile methodology (Everaerts 2018)

During the project's development there were several areas the author was concerned about namely limited time, lack of the author's personal experience, the project being a single person project and finally how complex the project actually was. Because of this it was necessary for a project management methodology. For the project, the author used an Agile Methodology as it would allow the author to quickly adapt and change the project due to unexpected issues. The Agile Methodology that was picked was, Agile Kanban, because Kanban is continuous flow allowing the author to plan and react 'in the moment' based on results and not ahead of time (Dan Radigan Unknown). This aided greatly in the iterative development of the different AIs as the author always knew what they needed to do by when in order to meet the deadline. Both a project backlog and a Kanban board were produced which can be seen in appendix…....

3.2 Research Method:

Hypothesis:
An AI can effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players.
Null Hypothesis:
An AI cannot effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players.

In order to prove that an AI could effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players. The author ran two experiments against 4 different versions of the AI.

Unchanged AI:
The plain AI without any changes running whatsoever.

Unchanged AI treated as a human player:
The Default AI but with any bonuses they get due to being AI removed. All the difficulties were set to Settler (the civ equivalent of normal difficulty.) This was done because the AI always plays on Chieftain difficulty and then the AI receives bonuses based on the difficulty the player is playing on (which were removed.)

AI V1:
Has the same changes to Difficulty as the above but is running slightly modified code. The changes to the code aren't very big but they deal with increasing the priority of building worker units under certain conditions. *Changes to code are marked with //New code -

```
/// "Need Tile Improvers" City Strategy: Do we REALLY need to train some Workers?
bool CityStrategyAIHelpers::IsTestCityStrategy_NeedTileImprovers(AICityStrategyTypes
eStrategy, CvCity *pCity)
{
	CvPlayer &kPlayer = GET_PLAYER(pCity->getOwner());
	int iCurrentNumCities = kPlayer.getNumCities();
```

```
	int iLastTurnWorkerDisbanded =
kPlayer.GetEconomicAI()->GetLastTurnWorkerDisbanded();
	if (iLastTurnWorkerDisbanded >= 0 && GC.getGame().getGameTurn() -
iLastTurnWorkerDisbanded <= 25)
	{
		return false;
	}

	int iNumWorkers = kPlayer.GetNumUnitsWithUnitAI(UNITAI_WORKER, true, false);
	// If it's a minor with at least 1 worker per city, always return false
	if (kPlayer.isMinorCiv())
	{
		if (iNumWorkers >= iCurrentNumCities)
			return false;
	}
	else{
		int iNumCities = max(1, (iCurrentNumCities * 3) / 4);
		if (iNumWorkers >= iNumCities)
			return false;
		// If we're losing at war, return false
		if (kPlayer.GetDiplomacyAI()->GetStateAllWars() == STATE_ALL_WARS_LOSING)
			return false;
	}

	// If we're under attack from Barbs and have 1 or fewer Cities and no credible
defense then training more Workers will only hurt us
	if (iCurrentNumCities <= 1)
	{
		CvMilitaryAI* pMilitaryAI =kPlayer.GetMilitaryAI();
		MilitaryAIStrategyTypes eStrategyKillBarbs = (MilitaryAIStrategyTypes)
GC.getInfoTypeForString("MILITARYAISTRATEGY_ERADICATE_BARBARIANS");
		if (eStrategyKillBarbs != NO_MILITARYAISTRATEGY)
		{
			if (pMilitaryAI->IsUsingStrategy(eStrategyKillBarbs))
			{
				MilitaryAIStrategyTypes eStrategyCriticalDefense =
(MilitaryAIStrategyTypes)
GC.getInfoTypeForString("MILITARYAISTRATEGY_EMPIRE_DEFENSE_CRITICAL");
				if (eStrategyCriticalDefense != NO_MILITARYAISTRATEGY)
				{
					if (pMilitaryAI->IsUsingStrategy(eStrategyCriticalDefense))
					{
						return false;
					}
```

```
                        }
                  }
            }
      }

      CvAICityStrategyEntry* pCityStrategy =
pCity->GetCityStrategyAI()->GetAICityStrategies()->GetEntry(eStrategy);

      //New code - If we have less workers than cities priorities building workers.
      if (iNumWorkers < iCurrentNumCities){
            return true;
      }

      int iModdedNumWorkers = iNumWorkers * /*67*/ pCityStrategy->GetWeightThreshold() /
100;
      int iModdedNumCities = iCurrentNumCities + kPlayer.countCitiesFeatureSurrounded();

      // We have fewer than we think we should, or we have none at all
      if (iModdedNumWorkers <= iModdedNumCities || iModdedNumWorkers == 0)
      {
            // If we don't have any Workers by turn 30 we really need to get moving
            int iDesperateTurn = /*30*/
GC.getAI_CITYSTRATEGY_NEED_TILE_IMPROVERS_DESPERATE_TURN();

            iDesperateTurn *= GC.getGame().getGameSpeedInfo().getTrainPercent();
            iDesperateTurn /= 100;

            if (GC.getGame().getElapsedGameTurns() > iDesperateTurn)
                  return true;
      }

      return false;
}
```

This is a small change but it should up the priority of workers if the AI every has less workers than cities.

```
/// "Want Tile Improvers" City Strategy: Looks at how many Builders are available
empire-wide.  This is not a Player Strategy because it's only worried about training new
Builders in Cities, and not Techs, Policies, etc.
bool CityStrategyAIHelpers::IsTestCityStrategy_WantTileImprovers(AICityStrategyTypes
eStrategy, CvCity *pCity)
{
      CvPlayer &kPlayer = GET_PLAYER(pCity->getOwner());
      int iLastTurnWorkerDisbanded = kPlayer.GetEconomicAI()->GetLastTurnWorkerDisbanded();
      if (iLastTurnWorkerDisbanded >= 0 && GC.getGame().getGameTurn() -
iLastTurnWorkerDisbanded <= 10)
```

```
	{
		return false;
	}

	if (!GET_PLAYER(pCity->getOwner()).isMinorCiv())
	{
		// If we're losing at war, return false
		if (GET_PLAYER(pCity->getOwner()).GetDiplomacyAI()->GetStateAllWars() ==
STATE_ALL_WARS_LOSING)
			return false;
	}

	int iNumWorkers = kPlayer.GetNumUnitsWithUnitAI(UNITAI_WORKER, true, false);

	if (iNumWorkers >= ((kPlayer.getNumCities() *  4) / 3) + 1)
		return false;

	// If we're under attack from Barbs and have 1 or fewer cities then training more
Workers will only hurt us
	//if (kPlayer.getNumCities() <= 1)
	//{
	//	MilitaryAIStrategyTypes eStrategyKillBarbs = (MilitaryAIStrategyTypes)
GC.getInfoTypeForString("MILITARYAISTRATEGY_ERADICATE_BARBARIANS");
	//	if (eStrategyKillBarbs != NO_MILITARYAISTRATEGY)
	//	{
	//		if
(GET_PLAYER(pCity->getOwner()).GetMilitaryAI()->IsUsingStrategy(eStrategyKillBarbs))
	//			return false;
	//	}
	//}

	// Don't get desperate for training a Builder here unless the City is at least of a
certain size
	if (pCity->getPopulation() >= /*2*/
GC.getAI_CITYSTRATEGY_WANT_TILE_IMPROVERS_MINIMUM_SIZE())
	{
		int iNumBuilders = kPlayer.GetNumUnitsWithUnitAI(UNITAI_WORKER, true, false);

		// If we don't even have 1 builder on map or in a queue, turn this on
immediately
		if (iNumBuilders < 1)
		{
			return true;
		}
```

```
            //New code - If we don't have 1 worker per city piroirtize workers.
            if (iNumBuilders < kPlayer.getNumCities()){
                  return true;
            }

            //New code - If we have more than 2 cities and don't have a extra worker, Build
one (hopefully used for building roads.)
            if (kPlayer.getNumCities() > 2 && iNumBuilders < kPlayer.getNumCities() + 1){
                  return true;
            }

            CvAICityStrategyEntry* pCityStrategy =
pCity->GetCityStrategyAI()->GetAICityStrategies()->GetEntry(eStrategy);

            int iWeightThresholdModifier =
CityStrategyAIHelpers::GetWeightThresholdModifier(eStrategy, pCity);    // 2 Extra Weight
per TILE_IMPROVEMENT Flavor
            int iPerCityThreshold = pCityStrategy->GetWeightThreshold() +
iWeightThresholdModifier;     // 40
//More code that has cut because it hasn't been changed
```

These changes should mean that if the AI ever has less workers than cities or if the AI has more than 2 cities the priority of workers should be increased. Overall both these and other changes should mean that the chance of the AI choosing to build a worker has increased as a result the AI should in theory have more improved tiles.

AI V2:

As above with V1, but has several clauses in the expansion code that check for 'cultural victory' removed.

```
bool EconomicAIHelpers::IsTestStrategy_ExpandLikeCrazy(EconomicAIStrategyTypes eStrategy,
CvPlayer *pPlayer)
{
      if (GC.getGame().isOption(GAMEOPTION_ONE_CITY_CHALLENGE) && pPlayer->isHuman())
      {
            return false;
      }

      // Never run this if we are going for a cultural victory since it will derail that
      //AIGrandStrategyTypes eGrandStrategy = (AIGrandStrategyTypes)
GC.getInfoTypeForString("AIGRANDSTRATEGY_CULTURE");
      //if (eGrandStrategy != NO_AIGRANDSTRATEGY)
      //{
      //    if (pPlayer->GetGrandStrategyAI()->GetActiveGrandStrategy() == eGrandStrategy)
      //    {
                  // Does number of cities matter?
      //          if (GC.getMap().getWorldInfo().GetNumCitiesPolicyCostMod() > 0)
```

```
	//		{
	//			return false;
	//		}
	//	}
	//}

	//Verision 2 change: If we can expand do so! Don't worry if the AI decides to do a culture victory.
	int iFlavorExpansion = pPlayer->GetGrandStrategyAI()->GetPersonalityAndGrandStrategy((FlavorTypes)GC.getInfoTypeForString("FLAVOR_EXPANSION"));
	CvEconomicAIStrategyXMLEntry* pStrategy = pPlayer->GetEconomicAI()->GetEconomicAIStrategies()->GetEntry(eStrategy);
	if (iFlavorExpansion >= pStrategy->GetWeightThreshold())
	{
		return true;
	}

	return false;
}
```

The code above 'Version 2 change' checks if the AI is going for a cultural victory. This code has been removed so that if the AI was going for a cultural victory it wouldn't impact on expansion.

```
/// Are we running out of room on our current landmass?
bool EconomicAIHelpers::IsTestStrategy_ExpandToOtherContinents(CvPlayer *pPlayer)
{
	if (GC.getGame().isOption(GAMEOPTION_ONE_CITY_CHALLENGE) && pPlayer->isHuman())
	{
		return false;
	}

	// Never run this at the same time as island start
	EconomicAIStrategyTypes eStrategyIslandStart = (EconomicAIStrategyTypes)GC.getInfoTypeForString("ECONOMICAISTRATEGY_ISLAND_START");
	if (eStrategyIslandStart != NO_ECONOMICAISTRATEGY)
	{
		if (pPlayer->GetEconomicAI()->IsUsingStrategy(eStrategyIslandStart))
		{
			return false;
		}
	}

	// we should settle our island first
	EconomicAIStrategyTypes eEarlyExpansion = (EconomicAIStrategyTypes)
```

```
GC.getInfoTypeForString("ECONOMICAISTRATEGY_EARLY_EXPANSION");
	if (eEarlyExpansion != NO_ECONOMICAISTRATEGY)
	{
		if (pPlayer->GetEconomicAI()->IsUsingStrategy(eEarlyExpansion))
		{
			return false;
		}
	}

	// Never desperate to settle distant lands if we are at war (unless we are doing okay at the war)
	MilitaryAIStrategyTypes eStrategyAtWar = (MilitaryAIStrategyTypes) GC.getInfoTypeForString("MILITARYAISTRATEGY_LOSING_WARS");
	if (eStrategyAtWar != NO_MILITARYAISTRATEGY)
	{
		if (pPlayer->GetMilitaryAI()->IsUsingStrategy(eStrategyAtWar))
		{
			return false;
		}
	}

	// Never run this if we are in serious need of defending the stuff we already have
	// ??? How about at extreme difficulties?
	//MilitaryAIStrategyTypes eBuildCriticalDefenses = (MilitaryAIStrategyTypes) GC.getInfoTypeForString("MILITARYAISTRATEGY_EMPIRE_DEFENSE_CRITICAL");
	//if (eBuildCriticalDefenses != NO_MILITARYAISTRATEGY)
	//{
	//	if (pPlayer->GetMilitaryAI()->IsUsingStrategy(eBuildCriticalDefenses))
	//	{
	//		return false;
	//	}
	//}

	// Never run this if we are going for a cultural victory since it will likely derail that
	//AIGrandStrategyTypes eGrandStrategy = (AIGrandStrategyTypes) GC.getInfoTypeForString("AIGRANDSTRATEGY_CULTURE");
	//if (eGrandStrategy != NO_AIGRANDSTRATEGY)
	//{
	//	if (pPlayer->GetGrandStrategyAI()->GetActiveGrandStrategy() == eGrandStrategy)
	//	{
			// Does number of cities matter?
	//		if (GC.getMap().getWorldInfo().GetNumCitiesPolicyCostMod() > 0)
	//		{
	//			int iMaxCultureCities = GC.getAI_GS_CULTURE_MAX_CITIES();
```

```
                        // scale this based on world size
      //                const int iDefaultNumTiles = 80*52;

      //                iMaxCultureCities = (iMaxCultureCities * GC.getMap().numPlots()) /
iDefaultNumTiles;

      //                if (pPlayer->getNumCities() >= iMaxCultureCities)
      //                {
      //                      return false;
      //                }
      //          }
      //    }
      //}
      //Verision 2 change: We don't care about culture! if we need to expand to a another
culture do so!

      if (pPlayer->getCapitalCity() != NULL)
      {
            CvArea* pArea = GC.getMap().getArea(pPlayer->getCapitalCity()->getArea());

            // Do we have another area to settle (either first or second choice)?
            int iBestArea, iSecondBestArea;

pPlayer->GetBestSettleAreas(pPlayer->GetEconomicAI()->GetMinimumSettleFertility(),
iBestArea, iSecondBestArea);

            if ( (iBestArea != pArea->GetID() && iBestArea != -1) || (iSecondBestArea !=
pArea->GetID() && iSecondBestArea != -1))
            {
                  return true;
            }
      }

      return false;
}
```

Again this code has the segment that checks for a cultural victory removed. This is because the author wanted to make sure that culture didn't impact the AI's empire growth. As it wasn't clear if it was a factor. The author expects that these changes combined with others, increase the chances of the AI attempting to expand both on the same continent and different continents. As well as eliminating a factor that could decrease the effectiveness of the AI's empire growth.

The Civilization the AI was playing:

In Civilization V each Civilization has a different playstyle as well as unique units and traits that impact gameplay in order to get the most ‘honest’ results possible the author needed to remove these aspects. In addition to this each Civilization is a series of numbers that decide what approach / types of victories the Civilization prefer to take. The author developed a modified version of England with the special traits and units removed which all of the AI ran.

The following is the ‘Flavour’ Array the Civilization had once modified (The values shown are important values and the rest have been left at their ‘default’):

- Offence: 3 - How strongly the Civilization will focus on building units to attack with
- Defence: 6 - How strongly the Civilization will focus on building units to defend with
- Growth: 9 - How strongly the Civilization will focus on increasing the Population of cities.
- Expansion: 8 - How strongly the Civilization will focus on settling new cities.
- Tile Improvement: 9 - How strongly the Civilization will focus on improving tiles within its territory
- Production: 8 - How strongly the Civilization will focus on increasing the production output of its cities.
- Infrastructure: 7 - How strongly the Civilization will focus on building roads between cities.
- Naval Growth: 6 - How strongly the Civilization will focus on increasing the Population of coastal cities.
- Naval Tile Improvement: 7 - How strongly the Civilization will focus on improving water tiles within its territory.

Game Settings:

Every experiment ran with the following game settings:

Victories:

- Domination - Win by taking every other Civilization’s Capital City
- Science - Win by building a rocket to send into space (only happens in the end game.)
- Culture - Win by increasing ‘tourism’ to a point where it is higher than the culture stat of all other Civilizations still in the game (tends to only happen late middle to late game.)
- Diplomacy - Win by being elected ‘democratically’ as world leader (tends to only happen late game when Venice can buy everyone’s votes.)

Speed: Standard

Difficulty: Settler (Normal)

Complete Kills On - This is so the author could play the game with only on unit otherwise the game would just close this means the author doesn’t have to have any cities and actually play the game.

No barbarians - Barbarians are random ‘camps’ of military units hostile to everyone they add a bit of randomness to the game because they can spawn anywhere that is in the ‘fog of war’ and can end up destroying improved tiles if they get close enough to a Civ’s city. The Barbarians were disabled to prevent them from impacting the results.

Testing Methodology:

Figure B The Civilization end game screen (Firaxis Games 2010).

Originally the author had planned to record data from the end game screen. However during testing it emerged that the majority of data from the end game screen wasn't useful, only showing who was the 'best' in a certain area. As well as the score listed was fairly arbitrary and didn't exactly translate into the actual game. Because of this, a mod called 'In Game Editor' made by steam user donquiche was used to manually record data (donquiche 2012). This was accomplished by using the mods ability to see data like: what buildings has this city built, what research has this civilization completed etc.. and then recording said data manually.

The author decided to run each test to turn 150, this is because prior testing revealed that turn 150 is around about the point where the AI just about reached the end of the classical era / the beginning of the medieval era. Each experiment was run 5 times for each version of the AI. This is because research into the subject revealed that 5 was around the number that would, reveal if further testing was needed. It was eventually decided that the data collected after 5 runs was sufficient. Data was recorded about the AIs current situation and what it had done on previous turns every 15 turns. This is because in Civilization, something doesn't always happen on every turn. During testing, it was determined that it would be best to record data every 30 turns for the first time a individual version of the AI was running one of the experiments. And then to record text based observations for remaining 'runs', recording data only on turn 150. This is because the amount of useful data collected was the same, in addition text based observations provided valuable insight into what the AI did and when. As well as providing context for special cases which numerical data alone could not provide.

Experiment 1 (Map 1):

Figure C Experiment Map 1 (Firaxis Games 2010).

Experiment 1 is a small island that can in-fact support two cities. The cows and sheep on the right of the island will be out of London's improvement range. This is because you can only 'work' (have that tile give you a 'yield' benefit) a maximum of 3 tiles away from you're city. In order to make maximum use of the resources on this island the AI will need to build a second city on the right of a island. Underneath the 'Warrior' unit is a gold resource which provides a nice amount of production and gold per turn when improved. The wheat resource provides a nice amount of food when improved, the cows and the sheep provide both food and production when improved. The AI is provided with a city (London in this case) and a warrior to recreate a typical turn 1 start that any player would have. The reason the AI's city has been placed already is because otherwise they would be placed randomly which is another factor that can drastically affect the result.

There is another AI in the bottom left hand corner, this is just so the author could turn on domination victory as a viable victory without having to have a city. The domination victory was enabled because its a more realistic interpretation of a normal game and the AI should

be able to grow and expand while building appropriate defences even if the goal of the project wasn't to deal with military units. The submarine in the bottom left hand corner is just so the author could observe the game without the game ending and without the AI being able to see the unit.

Experiment 2 (Map 2):

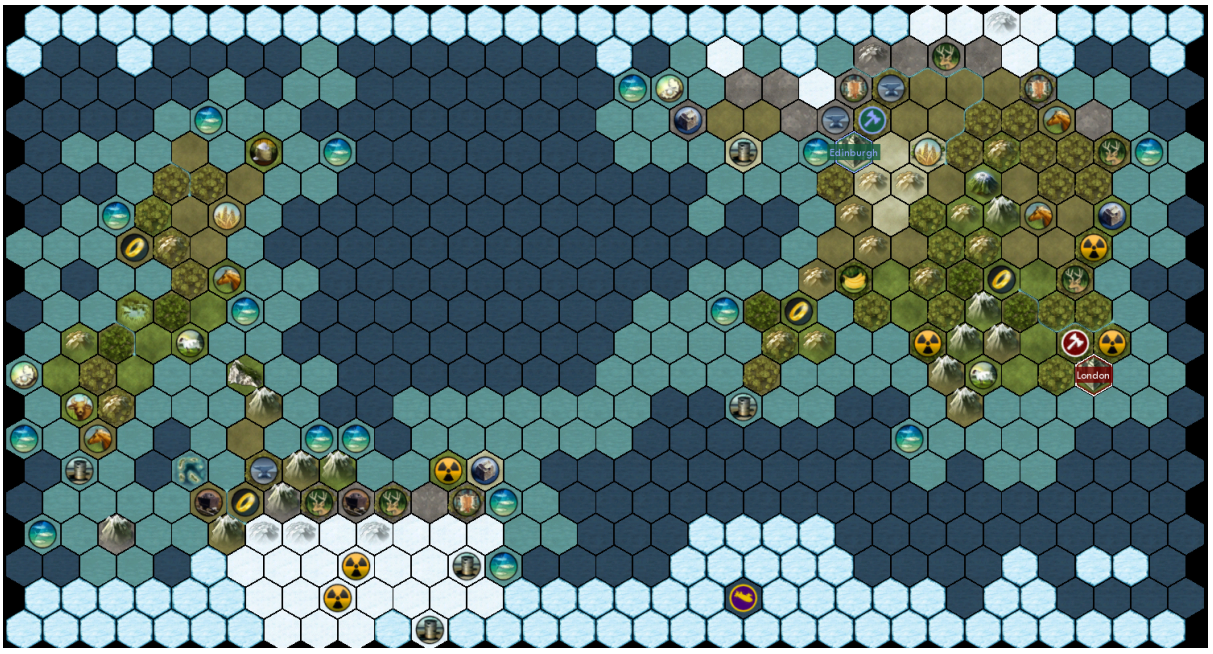


Figure D: Experiment Map 2 (Firaxis Games 2010).

Experiment 2 is a much bigger map with multiple continents the AI has been set up in a semi decent starting location with the ability to expand easily. In addition there is a second AI present on the map the two will compete for resources. The second AI are The Celts which were picked because they are a fairly aggressive AI that turn aggressive if an opponent's military is weak. The second AI was added to see how the AI would respond to the new 'threat' as well as to add a little bit of chaos and realism as in a real game the AI would be playing against both other AI and human players.

It's worth noting on this map that from every continent you can get to every other continent with only the 'optics' technology (A technology that allows you to travel through the coastal 'Light blue/cyan' tiles with any land unit a requirement for recon early game.)

3.3 Results And Evaluation:

Expectation:
Going into the project the author had the expectation that the AI would perform worse in cases where the AI bonuses were removed.
Going into the project the author had the expectation that: unchanged AI treated like a human player, AI V1, AI V2. Would perform worse, with a slight improvement on the end of V2 as it should expand more.

Actual result:
According to the results the bonuses the AI receives does not really make much of a difference. Below shows the average score of each version of the AI by turn 150. ‘Score’ isn’t exactly a metric that the project was really looking to test but its a decent measure of the progress that Civilization has made.

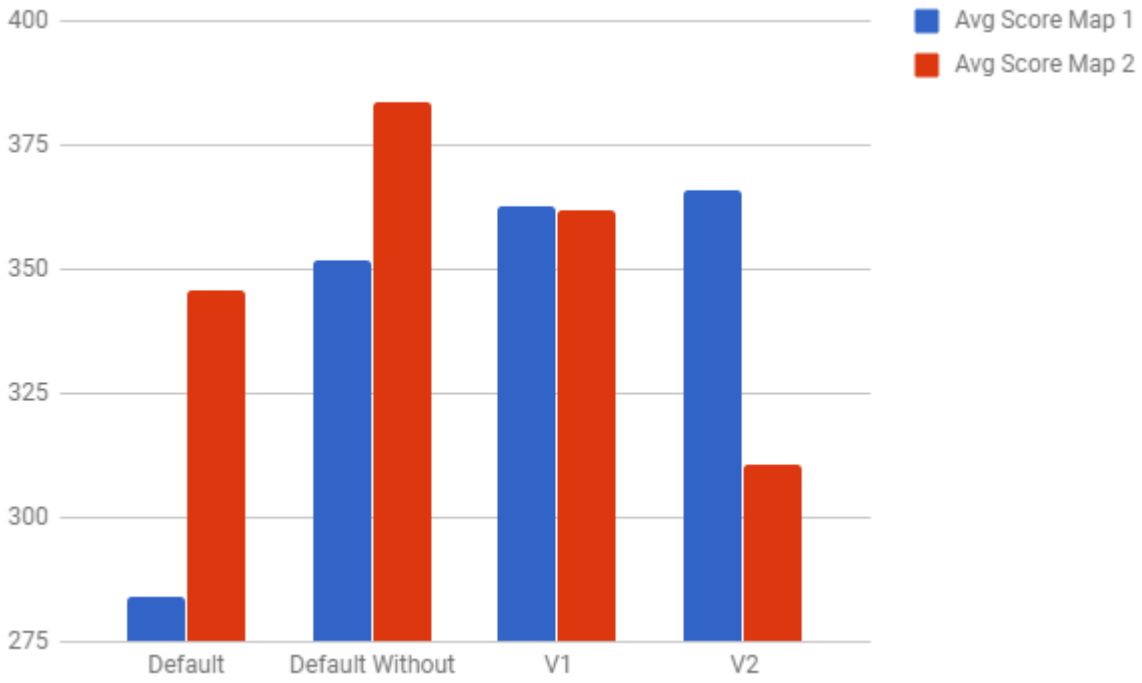
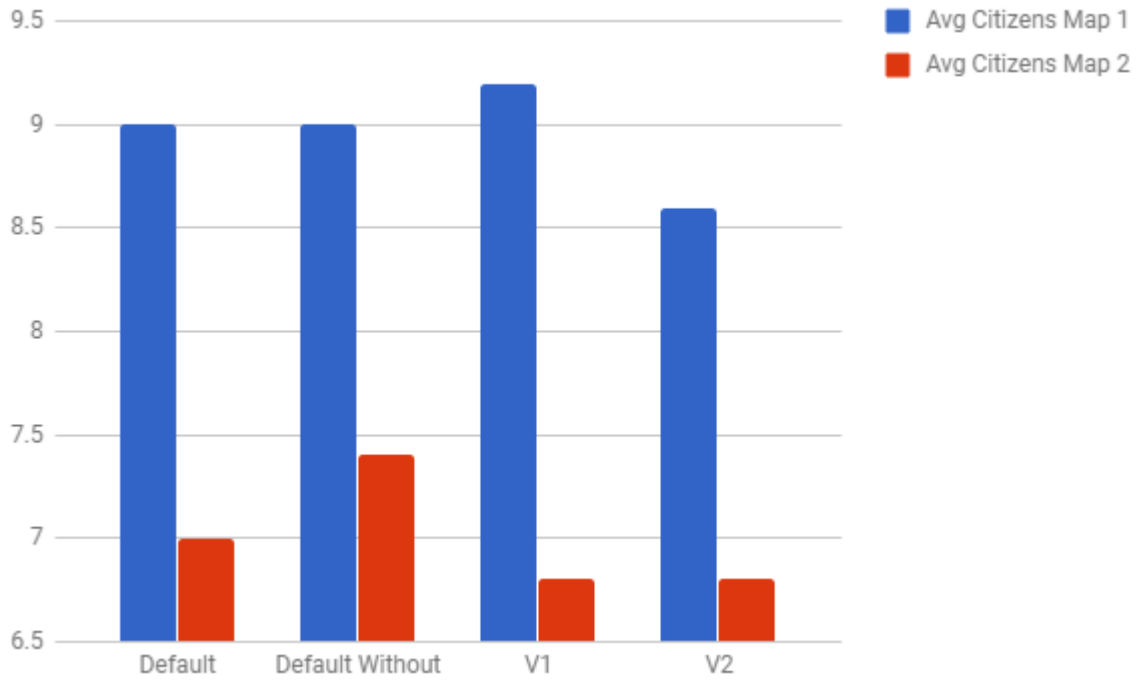


Figure E: Average score of all AI versions by turn 150.

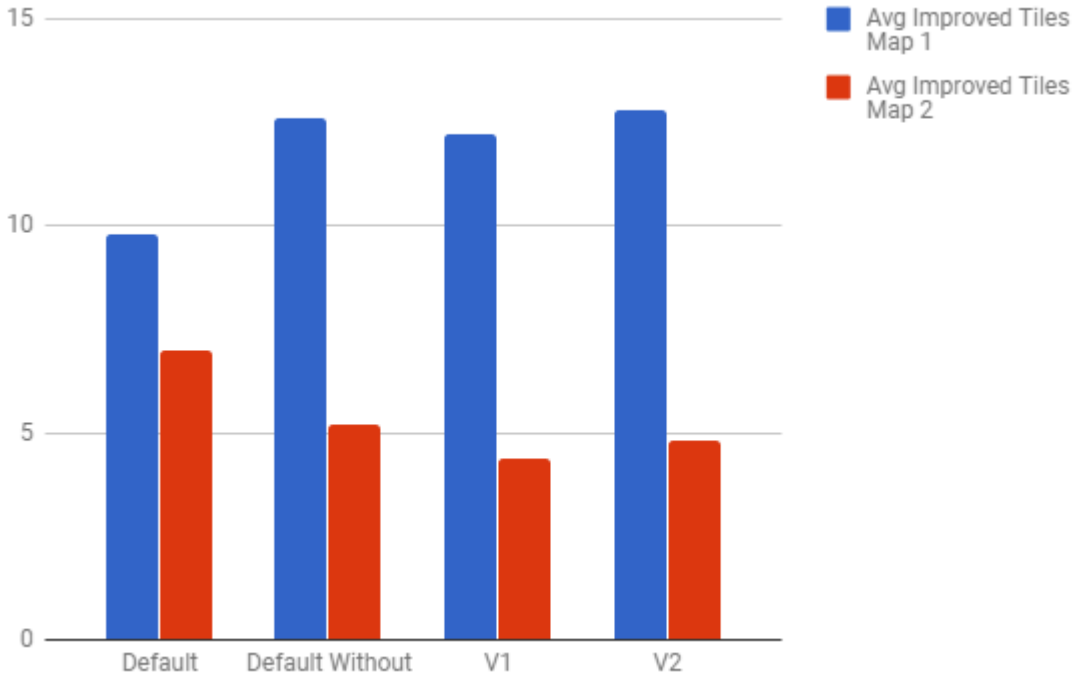
Immediately, there are some odd results, the expectation that the AI versions that have the AI bonuses removed would score lower, is false. In fact we see the opposite, the AI with the bonuses removed performed better (In terms of score at least.) This is because of several reasons:

1. The AI’s approach to the game ‘I.e. what victory it wants to try and achieve’ is chosen semi randomly at the start of the match with the Civilization’s ‘Flavours’ adding an offset to that (See ‘An unexpected result’ for more on this.) This ultimately means that what the AI does has some random bias to it. Which can lead to some ‘unpredictable’ results.
2. The bonuses the AI gets don’t really come into play until late game, this is because the majority like unhappiness generation and unit production speed are percentile modifiers which have a massively reduced impact early game. Adding to this the ‘Settler’ bonuses the AI gets aren’t very big and the AI actually plays on Chieftain meaning they are in some areas handicapped when compared to a player.
3. On ‘Map 2’ there is much more random ‘external’ stimulus (coming from the other civilization) however there is also a lot more room to expand. Because of this the Civilization’s choice of technologies are very important. Without the optics tech both Civilizations will run out of room on the starting continent very quickly. Because of this if the Civilization doesn’t expand quick enough and doesn’t choose the right techs it can end up with a really low score.



	Average Citizens Map 1	Average Citizens Map 2
Default	9	7
Default Without	9	7.4
V1	9.2	6.8
V2	8.6	6.8

Figure F: Average citizens in capital city by turn 150 for every AI version



	Average Improved Tiles Map 1	Average Improved Tiles Map 2
Default	9.8	7
Default Without	12.6	5.2
V1	12.2	4.4
V2	12.8	4.8

Figure G: Average tiles improved in capital city by turn 150 for every AI version.

Above are two ‘growth’ statistics that are related to each other, Citizens and Improved tiles. These are two very interesting statistics because a Civ can only gain benefit from as many improved tiles as the Civ has Citizens. However, each improvement costs a significant investment of time to build. If a Civilization has multiple cities micro-management skills are needed to effectively run all cities. It is possible to ‘over build’ as generally it will take longer for a city to ‘grow’ (increase amount of Citizens) than it will complete a tile improvement. Looking at the results on Map 1 the Default AI did a reasonable job of not ‘over building’ and not ‘under building’ with an average improved over just under 10 and an average Citizen ‘population’ of 9. However in this case, the AI decided to never settle another city so the one worker it built was just enough to keep up.

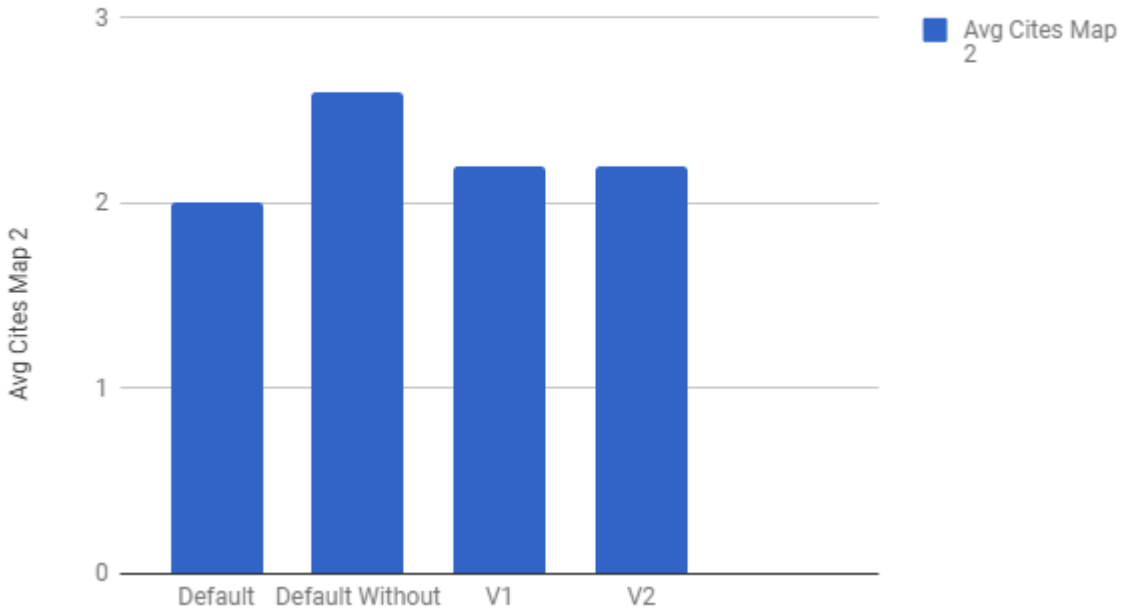


Figure H: Average Cities by turn 150 for every AI version.

Looking at the Map 2 results is another story, in cases all cases apart from the default AI the AI under built. Why is this? Simple, the AIs decided to build more cities on average. Because of this the worker unit was unable to keep up with demand as it spent many turns moving across to the map to get to another city. However there is a simple fix to this problem, build a second worker. In the author’s experience in the early game it is generally best to have as

many workers as you have cities or at least 2 at a minimum. In cases where the AI has at least 3 cities it needs at least 2 workers. This is because the AI will not just need tile improvements for the cities but also roads to increase unit movement speed. This simple mechanic creates both a micromanagement and a long term strategy problem for the AI solve. The author observed the AI gaining a second worker very rarely it mostly happened as a result of the AI choosing the 'Citizenship' social policy in the liberty tree which grants a worker immediately and decreases the turns it takes to build an improvement by 25%. And there was a case where the AI built the 'pyramids', a wonder that grants 2 workers and decreases the number of turns it takes to build an improvement by 25%. Which stacks with Citizenship however this only happened once.

The author plotted all the spots the AI decided settle a city for every AI version and Experiment run on Experiment Map 2 (See Appendix) The AI picked okay spots to place cities most of the time. There were however a few exceptions like AI V2 on run 5 picked an area of snow and tundra in the top right hand corner of the map to settle a city and the unchanged AI treated like a player on run 2 settled a city in that same area. The reason snow and tundra are so bad is they provide the worst tile bonuses in the game unless a special set of circumstances are met.

Special Cases and Other Observations:
Below are some special cases that occurred as well as some observations the author made during the experiment.

The Settler Problem:
Throughout all of the author's runs on all maps the AI always seemed to run into this problem at least once except in the case for map 1 where it had the problem every run. This is a problem where the AI builds a settler unit and does nothing with it wasting at least 10 turns or in the worse case never settling the city at all. This issue's root cause is the AI not building a scout unit to 'recon' the map. If it doesn't build a scout it doesn't know what's around it's city so it doesn't know where to place the city. So it builds a settler and has to wait until the tiles are revealed by either random chance or it builds a scout which reveals the tiles. This problem is easily solved in two ways:

1. Before the AI builds a Settler, Check if there is an active scout under the AI's control.
2. Use the Warrior gained at the start of the game to scout for a few turns. Most human players will use their warrior to recon the tiles around their starting city as well as search for 'Ancient ruins' which randomly grant a boon to whoever moves a unit onto the tile they occupy.
3. Only build a settler once we've identified either a single area to settle or several.

Why doesn't the AI use the Warrior to 'scout' early game? The AI assigns military units to the 'taticalAI' which controls what units are assigned to what roles and what those roles do. Unfortunately it seems like the AI doesn't see the Warrior as a scout and assigns it immediately to a military role which means all it does is patrol the borders of the city. This is fairly easily fixed by adding a few clauses that state if it is the early game and no scout exists, assign the warrior to the scouting role and then up the priority of a scout. When the scout is produced assign the warrior back to the military.

Map 1 and the Settler Problem:
As stated previously Map 1 suffered from the settler problem on every run and every version. Not once did the AI settle a second city despite making a settler. The island can technically support two cities however it's clear the AI didn't want to settle a city that near to it's capital city despite there being two resources to grab. Secondly the AI quite often didn't build a scout so it didn't always know it was on an island. And it's reasonable for the AI to assume that there is other land beyond its borders that is settleable after all there is at least one other Civ playing. In this case the 'don't build a settler until we've got somewhere we want to settle' would be a good strategy to employ.

Same Stimulus. Same Choices:
While recording data for map one the author reloaded the save on turn 0, three times and was surprised to find the AI made the exact same decisions every single turn milestone. There would be some minor differences in areas relating to pathfinding however the AI always ended up with the same score, same techs, same everything. Surprising but explainable, the way Civ's AI works is it initializes what approach to the game it's going to take at the start of the game. Things like what victories it's going to go for etc the Civilization the AI is playing gives a priority to a kind of victory type because of the Civilization plays. As the same save was being loaded each time that data was the same and Map 1 doesn't contain any outside stimulus it made the same choices what to build, what to research etc.

3.4 Results Summary

After the research the author has gained significant insight into how the AI performs and how that correlates to its design.
The author has identified the following areas the AI struggles in:

- Reconnaissance, The AI really should use it warrior to scout for new terrority. (See Settler Problem)
- Choosing what to build, from the author's observations the AI didn't always prioritize buildings that would provide larger benefit or increase the speed of growth. (I.e. Building a shrine before a granary.)
- Policy choices, The AI seemed oddly fasicated with both the honor and the piety tree. It's possible this was left over 'code' from the Civilization that was used as a template that wasn't removed.
- The AI's performance wasn't consistent because it relies on randomness to make decisions too much.
- Micromanagement of worker units and prioritizing them when more are needed.

Most of these problems boil down the AI relying too much on randomness. However if the randomness was removed, the result would be an AI that doesn't fit every Civilization's playstyle. For example, India likes to keep its empire small at around 2 cities so the author can see why it is constructed in the way it is. However, a more consistent approach might be to give each Civilization either a single or multiple baseline strategy that they follow which gets tweaked as the game goes on.

The results also show, that the modifications made to the AI in V1 and V2 impacted the performance of the AI in regard to empire expansion and growth negatively.

4. Project Evaluation:

4.1 Discussion:

Achievements:
In the author's opinion the project has narrowed down some of the reasons why the AI is not quite as capable as it should be, and why it isn't able to reliably grow an empire, at least, with the modifications made to it. It has also expanded the author's personal knowledge on not only how Civilization V's AI is put together but how those types of systems are put together in the first place.

Deficiencies:
For one, because of the nature of the AI and the game itself the results aren't always guaranteed to be repeatable, even on the same map. This is why some of the results the author received were vastly different (i.e. AI never building a worker vs building a worker on turn 21.) There are simply too many variables, which the author tried as much as possible to reduce even going so far as to remove humans from the equation entirely. It's almost a certainty that the results would be influenced heavily by the actions of a human player. Although given that the reason Civilization V exists is to be played to by a human as a game, perhaps the results would be more useful if a human was included.

Could've Would've:
The author would've like to have done more extensive improvements to the AI but due to the time constant and issues during development this was nearly impossible. If the author had the time, they would've done a complete rewrite of the AI system removing the AI's dependency on randomness. The author would also have liked to do more extensive testing, with different maps perhaps also including a human player. To ensure the results would actually transfer over into an actual game of civilization and not just a simulated environment.

4.2 Management Of Social, Legal Or Ethical Issues:

Because this project was focused on a automated system, the author was not dealing with any human data. Which is why the project was classified as 'low risk' and the reason the author was allowed to carry out the research. In addition, this project focused on the development of a Civilization V mod using the SDK and modding tools that Firaxis released for free with the game. The game itself supports modding openly even allowing the user to download and enable mods in the main menu. Because of this, the author did not encounter

any Social, Legal Or Ethical issues. Besides from the possibility of creating skynet, which based on the results, did not happen.

4.3 Incorporating feedback from supervisor meetings:

Every week the author attended a project meeting to discuss the project's progress as well as to ask for guidance. A summary 'log' can be seen in the Appendix. During the project's development the author had some difficulty learning the mod develop tools as well as debugging issues with the .dll that caused crashing. Meaning the author did not always have anything significant to show.

5. Conclusion:

To summarise, the author has:

- Created two maps where the experiment was conducted.
- Created a modification to remove all bonuses given to the AI
- Created a modified version of an existing playable Civilization and removing all special traits or units the Civilization has access too.
- Iteratively created and tested a modified version of the AI

The two experiments the author ran, were run 5 times on the following versions of the AI: Unchanged AI, Unchanged AI treated like a player, AI V1, AI V2. Each experiment was on a custom made map and ran until turn 150.
The two experiments were performed in an attempt to prove either hypothesis:
Hypothesis:
An AI can effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players.
Null Hypothesis:
An AI cannot effectively develop an empire during the ancient and classical era in Civilization V while playing under the same rules as human players.

In conclusion, the author was able to demonstrate the null hypothesis, that in its current state the AI cannot effectively develop an empire while being treated like a human player. Or rather, the AI cannot effectively develop an empire while being treated like a human player **reliably**. Because of this it is clear the ultimate project aim hasn't been fully met. However, this does not mean that with more extensive improvements the same result would occur. Infact the author believes it reasonable that an AI could be developed to solve the problem. But the results show the improvements to the AI were not effective. The research provides some insight as to how the AI functions and some of the causes to the issues that the AI has as a result. Because of this research the author believes that the AI relies on randomness to much and that the default AI could possibly develop an empire effectively in the given eras but could not reliably.

There are also quite a lot of grey areas that the research does not give us any insight into like: How the AI would compare with a different 'flavour' set, How the AI would compare on a

different map, How the AI would deal with other opponents (Human/AI) on the same maps. In addition the results are only relevant to Civilization 5 and no other game in the Civilization series.

In the future the following could be explored: Other ‘stages’ of the game, Different Civilization or Flavour sets, Different maps, Different ‘improvements’ to the AI. The author would personally be interested in how a complete rewrite of the AI system to more of a ‘rule based’ approach would affect the game, and if unpredictability would need to introduced in order to prevent players from exploiting the AI.

References:

Emrich, A. (1993) "Sneak Preview: Masters Of Orion". Computer Gaming World [online] 93. available from <http://www.cgwmuseum.org/galleries/issues/cgw_110.pdf> [29 April 2018]

Sandy Everaerts (2018) “The Agile Invasion” Agile Methodlogy Diagram [Online] Available From <https://www.initio.eu/blog/2018/1/18/the-agile-invasion> [30/04/2018]

Dan Radigan (n.d) Agile Kanban [Online] Available From <https://www.atlassian.com/agile/kanban> [07/02/18]

Donquiche (2012) Civilization V Ingame Editor Mod [Online] Available from: <https://steamcommunity.com/sharedfiles/filedetails/?id=77002777> [30/04/2018]

Firaxis Games (2010) Civilization V [Online] Available From: <https://store.steampowered.com/app/8930/Sid_Meiers_Civilization_V/> [29/04/2018]

David, O., van den Herik, H., Koppel, M. and Netanyahu, N. (2014) "Genetic Algorithms For Evolving Computer Chess Programs". *IEEE Transactions On Evolutionary Computation* [online] 18 (5), 779-789. available from <https://arxiv.org/pdf/1711.08337.pdf> [3 February 2018]

Vinyals, O., Ewalds, T., Bartunov, S., Georgiev, P., Sasha Vezhnevets, A., Yeo, M., Makhzani, A., Kuttler, H., Agapiou, J., Schrittwieser, J., Quan, J., Gaffney, S., Petersen, S., Simonyan, K., Schaul, T., van Hasselt, H., Silver, D., Lillicrap, T., Calderone, K., Keet, P., Brunasso, A., Lawrence, D., Ekermo, A., Repp, J. and Tsing, R. (2018) *Starcraft II: A New Challenge For Reinforcement Learning*. [online] 1-2. available from <https://arxiv.org/pdf/1708.04782.pdf> [1 February 2018]

Jon Shafer (2013) Revisiting the Design of Civ 5 [Online] Avilable from: <https://www.kickstarter.com/projects/jonshafer/jon-shafers-at-the-gates/posts/404789> [27/04/2018]

Jon Shafer (2017) ‘Jon Shafer on design Civilization 5, joining paradox and making strategy games better’ [Interview by Adam Smith] [Online] Available from:

<https://www.rockpapershotgun.com/2017/05/17/jon-shafer-on-designing-civilization-5-joining-paradox-and-making-strategy-games-better/> [28/04/2018]

Hemulihemmo (2016) Civilization 5 AI in a nutshell [Online] Available from <https://www.youtube.com/watch?v=gb9OvAolJ80> [28/04/2018]

LAGTV (2013) AI Craft E3 [Online] Available From: <https://www.youtube.com/watch?v=DUTWRreaH-M> [28/04/2018]

Ed Welch (2007) Designing AI algorithms for turn based strategy games [online] available from: <https://www.gamasutra.com/view/feature/129959/designing_ai_algorithms_for_.php?page=1> [28/04/2018]

Glen Robertson, Ian Watson (2014) AI Magazine: A Review of Real-Time Strategy Game AI [Online] 76 (PDF:4) Available from: <https://pdfs.semanticscholar.org/0359/b687cccb4139f204a716dfd98e3652c67f67.pdf> [28/04/2018]

Kittisak Potisartra, Vishnu Kotrajaras (2009) Towards an evenly match opponent AI in turn based Strategy games [Online] Available from: <https://www.cp.eng.chula.ac.th/~vishnu/gameProg/papers/CGAT_TowardsEvenlyMatchTBS-FINAL.pdf> [29/04/2018]

Bibliography:

The following sources have been read but not directly referenced:

Lee, C., Wang, M., Yen, S., Wei, T., Wu, I., Chou, P., Chou, C., Wang, M. and Yan, T. (2016) "Human Vs. Computer Go: Review And Prospect [Discussion Forum]". *IEEE Computational Intelligence Magazine* [online] 11 (3), 67-72. available from <https://arxiv.org/ftp/arxiv/papers/1606/1606.02032.pdf > [3 February 2018]

Rahul Aralikatte,G Srinivasaraghavan (2015) “Phoenix: A self optimizing Chess engine” IEEE CICN 2015 [online] available from <https://arxiv.org/pdf/1603.09051.pdf> [3 February 2018]

Siddhesh Chaubal, Aditya Ayyarm, Chinmay Chauhan (Feb 6 2012) “AI in Chess” [online] available from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.361.6044&rep=rep1&type=pdf> [3 February 2018]

Ponsen, M. (2004) *IMPROVING ADAPTIVE GAME AI WITH EVOLUTIONARY LEARNING*. B.Sc. Delft University of Technology [Online] Available from

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.109.6055&rep=rep1&type=pdf > [3 February 2018]

Patrick Klek (2017) The Modders who decided to overhaul the AI in 'Civilization V' [Online] Availalbe from: <https://waypoint.vice.com/en_us/article/ez34jp/the-modders-who-decided-to-overhaul-the-ai-in-civilization-v> [29/04/2018]

Troy Goodfellow (2006) Civilization Chronicles Interview with Brian Reynolds [Online] Availalbe From: <https://www.civfanatics.com/misc/interview-with-brian-reynolds/> [29/04/2018]

James Vincent (2018) An Institute studying existential risk has made a Civilization mod about superintelligent AI [Online] Available from: <https://www.theverge.com/2018/1/5/16853628/civilization-v-ai-risk-mod-cser> [29/04/2018]

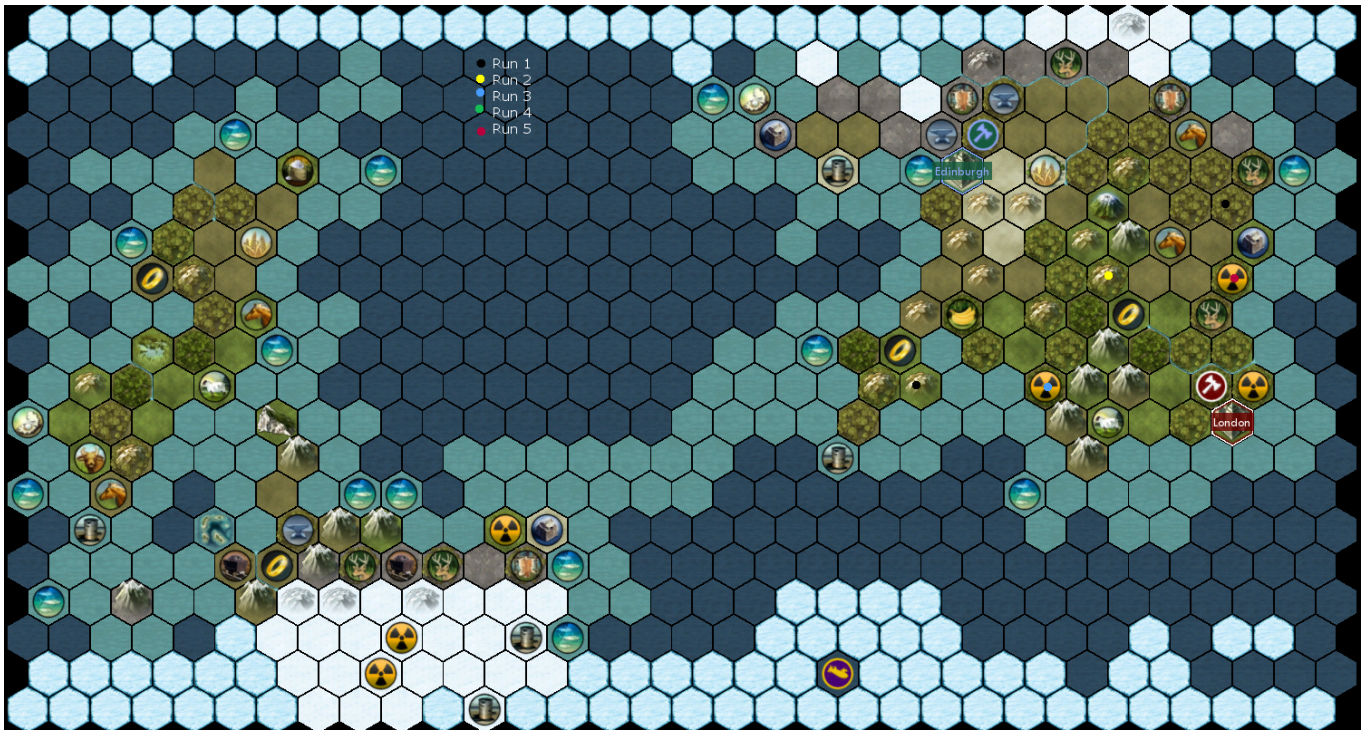
Glossary:

Civilization: Can refer to: the Civilization series of games, Civilization V, a playable 'Civilization' in game which the players control.
Civ: Refers to: Civilization V or a playable Civilization ingame which the players control.
Civilization 5: Another term for Civilization V.
Civ 5, Civ V: Another term for Civilization V.
Player: Something that interacts 'plays' with the game can be either human or artificial.
AI: Artificial Intelligence, in this case it means an artificial agent that plays the game.
Artificial Intelligence: Intelligence distracted by something artificial. Is often the illusion of intelligence.
Machine Learning: A field of Computer Science which attempts to get machines to perform a task without being explicitly programmed how to do that task. The machine is expected to learn from the experience.
Reinforcement Learning: A Machine Learning technique to 'teach' the machine the correct desired outcome by 'rewarding' it when it does something correct and 'punishing' it when it does something wrong.
4X: A term used to describe the general themes of a strategy game. EXplore, eXpand, eXploit, and eXterminate (Emrich 1993).
Worker: A 'civilian' unit a Civilization's city can produce, the worker unit can improve tiles and build roads.
Settler: A 'civilian' unit a Civilization's city can produce, the settler can settle (create) a new city for the Civilization. While being produced by a city, the city produced no food and thus cannot grow.
City: Cities allow Civilizations to produce a unit or building. Cities have buildings that can grant bonuses to both the Civilization and the city itself. Cities have a border around them that grows according to the 'culture' of the Civilization.

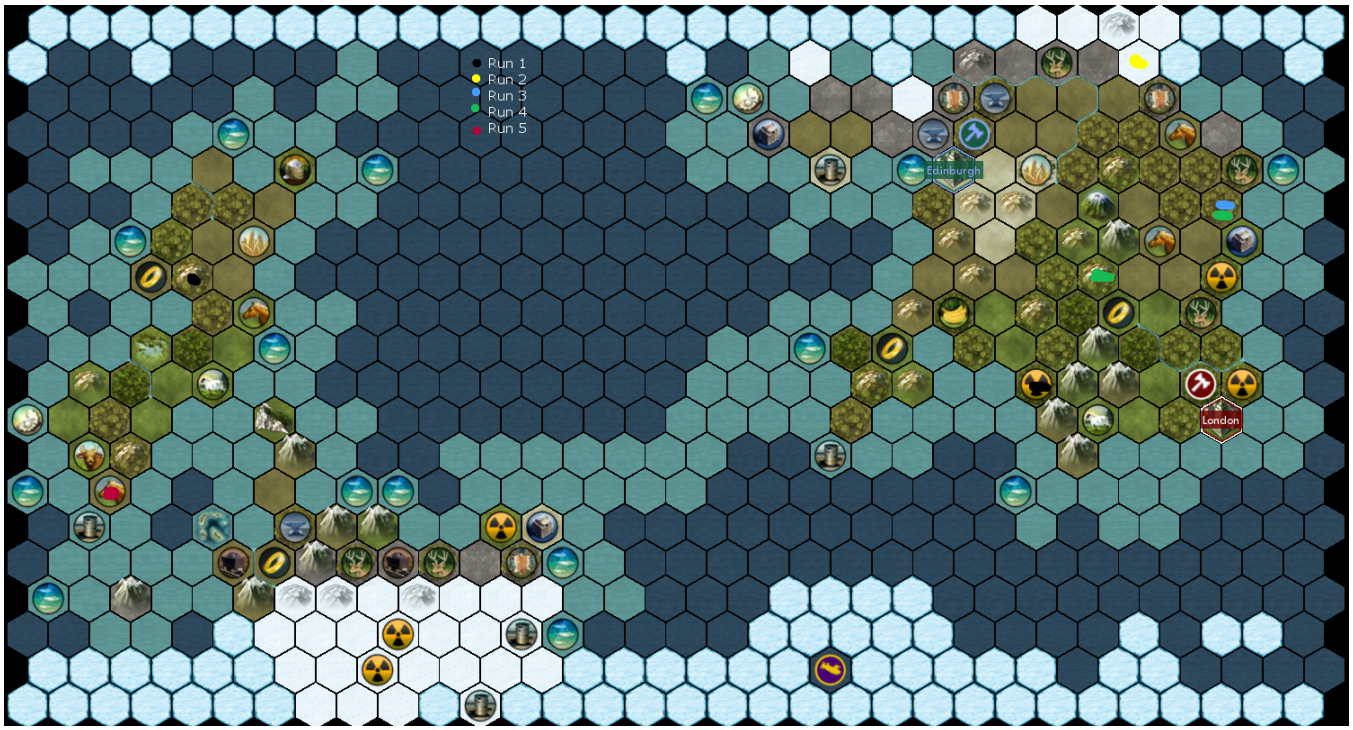
Food: A resource that every city has. Food decreases the number of turns it takes for the city to grow (Citizen increase.) Negative food can cause the city to reduce the population of the city and produces unhappiness (Citizen decrease.)
Culture: Culture is a statistic that controls how fast the borders of the Civilization's cities grow. Culture also decreases the amount of time it takes to get a social policy.
Production: A resource that each city has that determines the time it takes to produce units and buildings.
GPT (Gold Per Turn): The amount of gold the Civilization produces each turn. Going into negative gold will cause massive penalties.
Science: A resource that determines how quickly the Civilization gains new 'technologies'
Technologies: Grants a Civilization a new ability or benefit. For example Mining allows the Civilization to construct the 'mine' tile improvement on hills.
Tile: Every Civilization map is made up of tiles. Every tile has a type for example grassland, hills. Each type has a 'yield benefit' for example + 2 food +2 production which is granted to however controls that tile.
Tile Improvement: An improvement to a type of tile that increases the 'yield' it grants for example Hills grant +2 production bonus constructing a mine on that hill will increase the bonus to +3.
Social Policy: Grants the Civilization a bonus in a certain situation for example the 'Aristocracy' Social policy in the 'Tradition' Social policy tree provides, 15% production when building wonders and +1 happiness for every 10 Citizens in a city.
Wonders: Are buildings that provide a larger unique bonus than normal buildings. Wonders that aren't 'National Wonders' can only be built by one Civilization on the entire map. For example 'Stonehenge' is a wonder once built no other Civilization can build.
Happiness: A statistic that measures how happy the Civilization is. While Happiness is positive it decreases the time until the next 'gold age.' While Happiness is negative it decreases the time until the next goldage. If Happiness is negative and lower than -7 hostile units will start to spawn in the Civilization's cities mimicking 'riots.'
Golden Age: A number of turns where the Civilization receives bonus production, gold and culture.

Appendices:

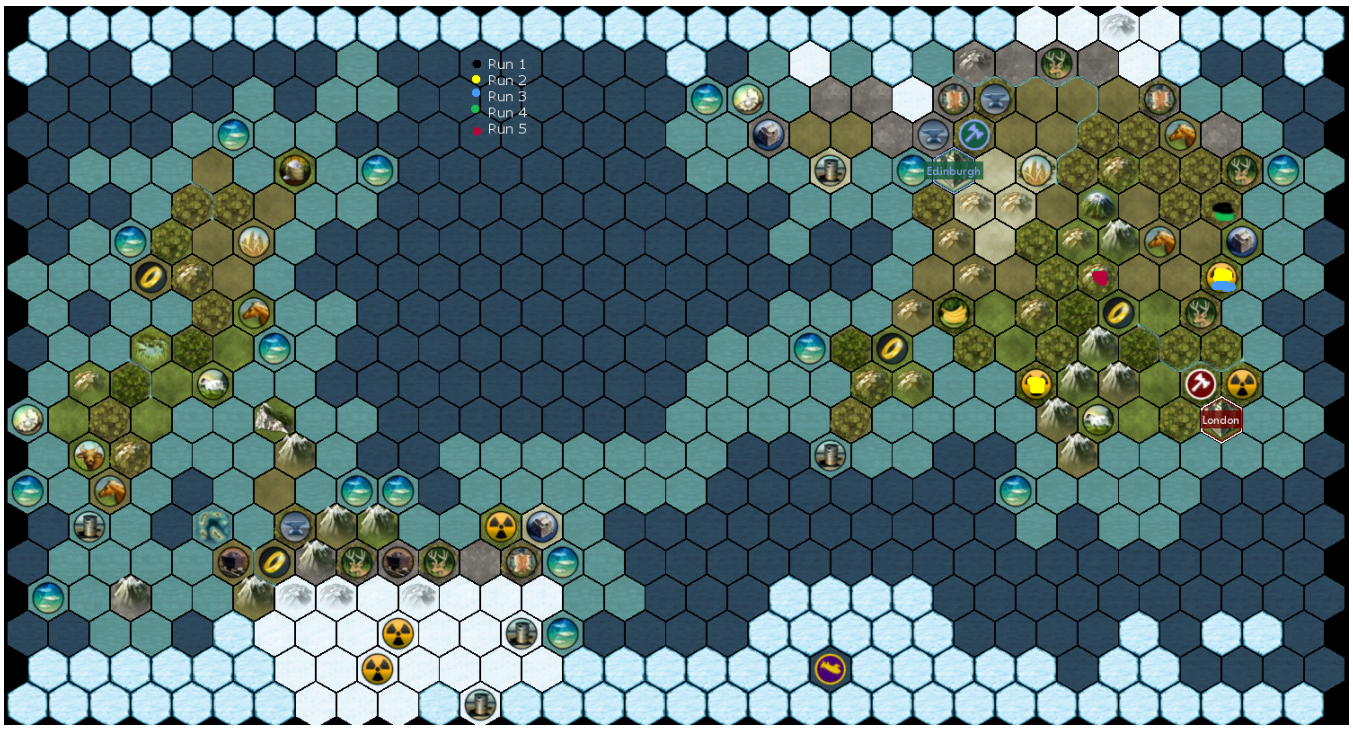
AI City Locations:
For high res images see: [This Link]
Unchanged AI:



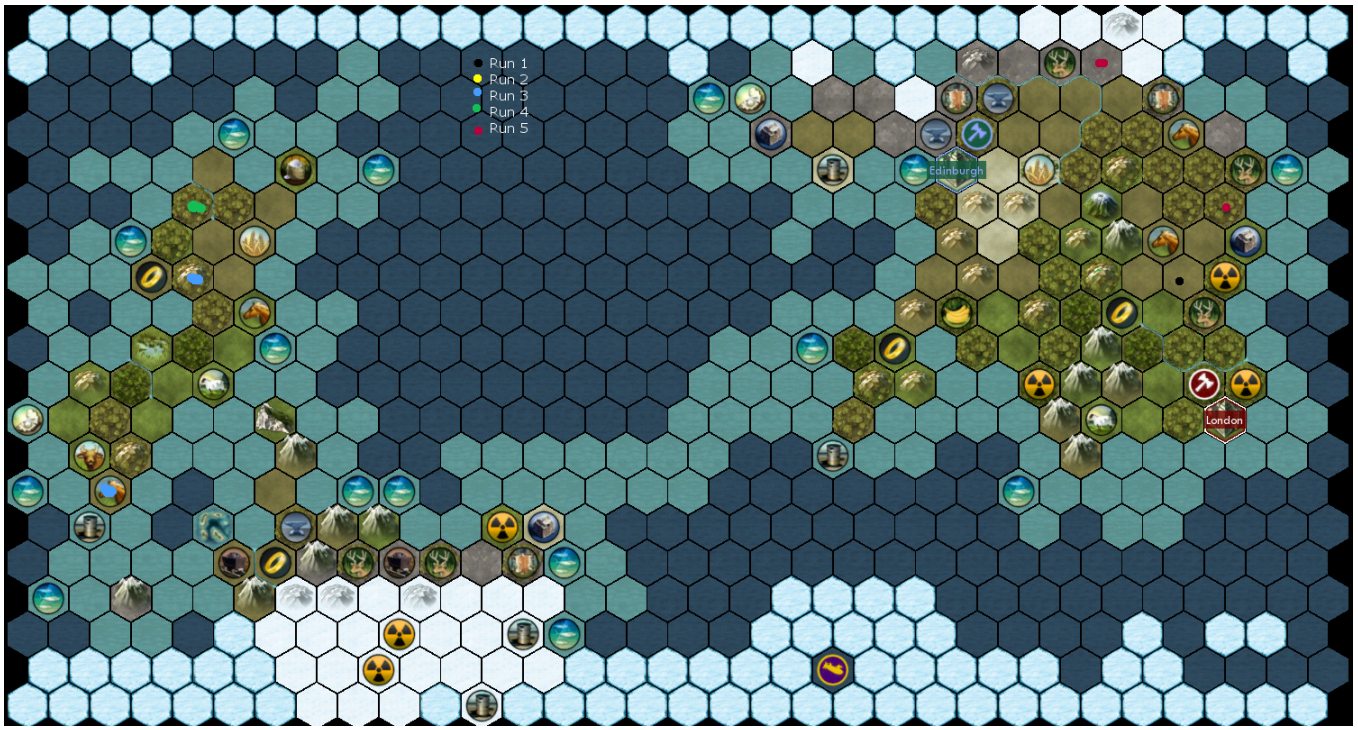
Unchanged AI Treated Like a Player:



AI V1:


Run 1
Run 2
Run 3
Run 4
Run 5
Edinburgh
London

AI V2:


Run 1
Run 2
Run 3
Run 4
Run 5
Edinburgh
London

Supervisor Meeting log:

Week (Friday)	Comments	Action
26/01/2018	Dicussed possible project ideas,	Decided upon doing a project to attempt to improve Civilization's AI in one area.
02/02/2018	Discussed Project Propsal, Went over a draft verision of it, Look more at project methodlogy and define some research paramters. Little bit more work to do on literature review	Will look at Agile, Kanban as a project methodlogy. Nailed down the research question fully and drew up some ideas for the experiment, Will look at expanding literature review
09/02/2018	Just final pieces of feedback before literature review hand in, Rewording and proof reading	Reworded some areas and fixed mistakes in grammer, Handed in Project propsal
16/02/2018	Explained Project Propsal feedback, Investigate statisitcal tools used to compare AIs, Report fails to identify impact of the current approach to AI diffculty on the player.	Will look at ways AI's are evaluated statistically and keep feedback in mind for final project report
23/02/2018	General Dicussion, Nothing really to report, Just setting up SDK's and modding tools	Will continue as planned
02/03/2018	Discussed project progress: Finished the experiment maps and modified an existing civilization,	Will continue as planned
09/03/2018	Ran into some issues with the modified Dll. Causing game crashes.	Nothing to report.
16/03/2018	Dicussed what would happen if it didn't work, that it doesn't matter because it wouldn't be research otherwise. Still having development issues.	No longer worried about project failure.
23/03/2018	Discussed project progress: Removed bonuses to AI, Fixed crashing issue	Continue as planned
30/03/2018		
06/04/2018	Easter Finished Verision 1	
13/04/2018	Easter Finished Version 2, Finished testing	

20/04/2018	Easter Begun drafting project.	
27/04/2018	Looked at current draft, Pointed out some flaws, discussed some possible reasons for the AI's behaviour, Dicussed some solutions	Fixed flaws with project portfolio, used some of the points in the discussion in the report and expanded upon them.

Kanban Boards:

To Do	In Progress	Done
Modify the game rules to not give the AI any advantages	Create both Civilization Maps that the AI will be tested on	
Modify the Civilization AI	Create a Civilization without any benefits that the AI 'play'	
Test and Analyse the modifications to the Civilzation AI		
Draft Project Report		

To Do	In Progress	Done
Test and Analyse the modifications to the Civilzation AI	Modify the game rules to not give the AI any advantages	Create both Civilization Maps that the AI will be tested on
Modify the Civilization AI		Create a Civilization without any benefits that the AI 'play'
Draft Project Report		

To Do	In Progress	Done
Test and Analyse the modifications to the Civilzation AI	Modify the Civilization AI	Create both Civilization Maps that the AI will be tested on
Draft Project Report		Create a Civilization without any benefits that the AI 'play'
		Modify the game rules to not give the AI any advantages

To Do	In Progress	Done
Draft Project Report	Test and Analyse the modifications to the Civilzation AI	Create both Civilization Maps that the AI will be tested on
		Create a Civilization without any benefits that the AI 'play'
		Modify the game rules to not give the AI any advantages
		Modify the Civilization AI

To Do	In Progress	Done
Draft Project Report	Modify the Civilization AI	Create both Civilization Maps that the AI will be tested on
		Create a Civilization without any benefits that the AI 'play'
		Modify the game rules to not give the AI any advantages
		Test and Analyse the modifications to the Civilzation AI

To Do	In Progress	Done
Draft Project Report	Test and Analyse the modifications to the Civilzation AI	Create both Civilization Maps that the AI will be tested on
		Create a Civilization without any benefits that the AI 'play'
		Modify the game rules to not give the AI any advantages
		Modify the Civilization AI

To Do	In Progress	Done
	Draft Project Report	Create both Civilization Maps that the AI will be tested on
		Create a Civilization without any benefits that the AI 'play'
		Modify the game rules to not give the AI any advantages
		Modify the Civilization AI
		Test and Analyse the modifications to the Civilzation AI

300/303COM Project Proposal

Full Name: Martyn Ian David Patrick

SID: 6457291

Supervisor: Dr Simon Billings

1. Project Details:

1.1 Research Question:

Can an AI effectively develop an empire during the ancient and classical era in civilization V while playing under the same rules as human players?

1.2 Keywords:

Ai; Civilization 5; Mod; Game Design; Video Games

1.3 Project Title:

An examination of the Artificial Intelligence (AI) in Civilization 5 and how it could be improved.

1.4 Client, Audience and Motivation:

This project will look at AI in games from both programming and game design perspectives. Mainly focusing on removing the ‘cheat’ bonuses that AI’s get in civilization 5 and looking at what happens when we remove these bonuses and how we might improve the AIs early growth without it using the bonuses as a crutch. At higher difficulties the bonuses AIs get almost completely negate the effects of the happiness system which is a massive advantage. CivFanatics which is a well known and trusted community site created a handy graph of all the different bonuses that AIs get (CivFanatics unknown). This project will look at if it was truly necessary to make the AI ‘cheat’ and explore the question “can we make an AI effective while playing under the same rules of the player?”

The project maybe of interest to those who are exploring the application of Artificial Intelligence applied to games. In addition anyone developing a ’4x’ strategy game, as many of the same elements can be transferred over. This project will be of special interest to Firaxis Games, the studio that created Civilization 5, as they may be able to take some of the results of this project and use them for future games.

1.5 Primary Research Plan:

For this project the primary research will involve the modification of the civilization 5 AI. Which will be improved iteratively so that it can be evaluated against previous iterations and the original game AI. This project will be judging how the AI 'effectively develop an empire during the ancient and classical era.' Because of this, the project will be focusing on the how the AI expands and grows its empire and ignoring the management of military / religious units. As this would add substantial complexity unrelated to the project core goals.
An average Civilization player will have done the following concerning 'empire growth':

1. Cities have a high population of 'citizens' for the given era
2. Each city has a number of tiles improved equal to the number of citizens that are able to 'work' those tiles.
3. The empire has made an effort to expand by founding more cities. (3 Cities by the end of the classical era is about average for a Civilization focusing on growth.)

These factors will be used to assess the overall 'empire growth' of the AI with the aim that the AI will match these as close of possible. The end game screens give you a turn by turn breakdown of each empire's overall growth. The author will use the end game screens in combination with the author's own metrics and data to analyse the AI's performance.

Mod Creation:

1. Create a small map (play area) to function as a controlled testing environment.
2. Modify one of the existing Civs this will be the Civilization our AI will be playing.
3. Iteratively modify and test the AI with player restrictions added in.

Analysis:

1. Compare each version of the AI against each other to see if the AI is becoming more effective at developing a small empire by improving tiles etc..
2. Compare the performance of the 'new' AI with player restrictions to the 'old' AI without restrictions.
3. Optional - Compare the performance of the 'new' AI against a player.

1.6 Project Plan:

Because of several limiting factors mainly: limited time, lack of the author's experience and the fact this is a single person project. Agile Kanban will be used in order to plan and execute the project. This is because Kanban is continuous flow which will allow the author to quickly adapt and change the project in order to account for unexpected changes (Dan Radigan Unknown). In addition to this Kanban makes it easy to see other tasks that are still pending, which will make it easier for the author to switch to another task. Thus helping to alleviate the author's lack of experience. Both a project backlog and a Kanban board will be produced and then updated anytime a change occurs.

2. Project Abstract and Literature Review:

2.1 Abstract:

This project aims to improve Civilization 5's AI 'expansionist' gameplay in the Ancient and Classical eras while treating the AI 'as a player.' The project will primarily be focused around producing a modification for the game where the AI has been changed and then comparing the 'empire growth' of the AI against: Older versions of itself, The 'default' AI without restrictions and finally players. This project will not be concerned with the military/religion elements of the game and will focus entirely on how the AI develop and grows an empire. As such: Cities placed, City Population and Tile improvements will be key indicators of the AI's growth.

2.2 Initial Literature Review:

AI in games has its foundation in chess and in the case of chess is considered a 'solved' problem however research is still ongoing in this area. This is because while we have chess programs capable of outperforming world champions these programs have a lack of learning capability and act like 'brute force searchers' (David et al 2014). A report which Omid David co-authored discussed some of the challenges that chess programs faced (David et al 2014):

1. The 'search space' the total number of different combinations of positions is estimated to be 10^{46} this means that any method based on a exhaustive search is ineffective.
2. The search space is not unimodal and because of this problem they argue that many top performing chess programs evaluation function's parameters are codependent on each other i.e. changing any number of the parameters produce wildly unpredictable results. As a result chess programs using 'conventional' means of learning are 'hand tuned' by the programmers themselves. The authors suggest that Genetic algorithms are a better fit for this type of problem.
3. The authors argue that the problem of tuning and learning is not well understood this is because all top chess programs are hand tuned by programmers which based their tuning on educated guessing and intuition. This suggests there are few if any other practical alternatives.

Because of these issues it is clear that there is still a great deal of work to be done in order to make 'truly intelligent' chess 'AI.' Not only do we need to make chess programs capable of learning more than moves that would result in a loss on the first turn but also chess programs that are tuned automatically. It is true that we have 'solved' AI chess as in we have solutions that work but that does not mean we have the most efficient solution nor do we have the most 'perfect' one. Like civilisation 5 the AI works and can provide some challenge

to the player especially when playing on higher difficulties but to provide a challenge it doesn't 'play by the rules.' Because of this it is clear that there is still room for improvement for Civilizations 5 AI.

Interestingly as part of an experiment run by Omid David et al, where footage of a specific grandmaster was used to train a chess program. They found that the chess program would only mimic the general 'grandmaster' style and not the style of a specific player if 1-ply searches are used (The program only looked for one move at a time and didn't plan ahead) (David et al 2014).

Starcraft 2 has recently become one of the hubs of AI and machine learning research this is because it represents some very interesting challenges not only from a 'strategic' decision making element but also how base mechanics function. A report by Oriol Vinyals et al looked into some of the challenges that Starcraft 2 presented (Vinyals et al Unknown):

1. Starcraft 2 often features multiple players competing for multiple different types of resources all of which the AI has to take into account. In addition the AI has to manage sometimes hundreds of units as well as take into account each individual unit abilities.
2. Imperfect information is often the only information the AI has access to this is because the AI has to move a camera in order to observe the map. There is also a fog of war which prevents the AI from seeing areas of the map where it has no units present.
3. The number of legal actions varies as the AI progresses through a 'research tree' in addition to this the things the AI builds and researches early have an effect later in the game.

Unlike in Starcraft, where there are a limited number of starting positions but a vast number of potential actions, Civilization 5's starting positions are randomly generated. This means there is a immediate challenge in determining standardised effective early moves.
This is the reason that the author is focusing on the early game, up to the classical era, as early advantages snowball leading to greater advantages in the mid and late game. Also diplomacy, the management of military and religious units will be ignored in an effort to reduce the complexity to a manageable level for this project. If successful the result should be an AI that, approaches the early game of Civilization in a way comparable to that of a human player. While the AI may still need advantages to make it competitive against good human players it should feel more natural for a human player to play against.

2.3 Bibliography/References:

Literature Review References:

David, O., van den Herik, H., Koppel, M. and Netanyahu, N. (2014) "Genetic Algorithms For Evolving Computer Chess Programs". *IEEE Transactions On Evolutionary Computation* [online] 18 (5), 779-789. available from <https://arxiv.org/pdf/1711.08337.pdf> [3 February 2018]

Vinyals, O., Ewalds, T., Bartunov, S., Georgiev, P., Sasha Vezhnevets, A., Yeo, M., Makhzani, A., Kuttler, H., Agapiou, J., Schrittwieser, J., Quan, J., Gaffney, S., Petersen, S., Simonyan, K., Schaul, T., van Hasselt, H., Silver, D., Lillicrap, T., Calderone, K., Keet, P., Brunasso, A., Lawrence, D., Ekermo, A., Repp, J. and Tsing, R. (2018) *Starcraft II: A New Challenge For Reinforcement Learning*. [online] 1-2. available from <https://arxiv.org/pdf/1708.04782.pdf> [1 February 2018]

Literature Review Bibliography:
The following sources have been read but not directly referenced in the literature review:

Lee, C., Wang, M., Yen, S., Wei, T., Wu, I., Chou, P., Chou, C., Wang, M. and Yan, T. (2016) "Human Vs. Computer Go: Review And Prospect [Discussion Forum]". *IEEE Computational Intelligence Magazine* [online] 11 (3), 67-72. available from <https://arxiv.org/ftp/arxiv/papers/1606/1606.02032.pdf> [3 February 2018]

Rahul Aralikatte,G Srinivasaraghavan (2015) “Phoenix: A self optimizing Chess engine” IEEE CICN 2015 [online] available from <https://arxiv.org/pdf/1603.09051.pdf> [3 February 2018]

Siddhesh Chaubal, Aditya Ayyarm, Chinmay Chauhan (Feb 6 2012) “AI in Chess” [online] available from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.361.6044&rep=rep1&type=pdf> [3 February 2018]

Ponsen, M. (2004) *IMPROVING ADAPTIVE GAME AI WITH EVOLUTIONARY LEARNING*. B.Sc. Delft University of Technology [Online] Available from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.109.6055&rep=rep1&type=pdf > [3 February 2018]

References used outside of Literature Review:
Dan Radigan (n.d) Agile Kanban [Online] Available From <https://www.atlassian.com/agile/kanban> [07/02/18]
CivFinantics (n.d) Civilization 5 AI bonuses [Online] Available From <https://www.civfanatics.com/civ5/info/difficulties/ > [07/02/18]