Kohan SAI Creation Overview v1.0

1 Creating an SAI

In your *Kohan* install folder, there is a sub-folder called **SAI**. This contains all the strategic artificial intelligence (SAI) profiles distributed with *Kohan*. Do not modify any of these files, as that will break the play of campaigns and may affect multi-player games. It is specifically important that you do not touch the *default.ini*. This contains all the fields and default settings for Kohan AIs. Any field that is not specified in an SAI profile uses the value specified by this file.

To create a new SAI player, simply copy an existing file to create your new one. For example, if you wanted to create an AI player called George based on Khan, simply copy Khan.ini to George.ini. You can also create your SAI based on the default.ini, but it is recommended that you do not define a field if you are going to use the default value.

You do not need to distribute your SAI profiles for multiplayer games. The host runs all AI players, and can place local profiles as players in the game. In this way, you can test your AI against other human players without needing to give it out. If the host drops, any custom AI profiles that were in use will be replaced by the default profile on the backup host.

2 Goal Allocation and Priorities

2.1 GeneralSettings

The first section is GeneralSettings. The profile's **name** and **description** is there as well as its **skill_level**, 0 to 2 with 0 being the easiest. Currently, one of the larger factors in deciding skill level is **full_recompute_time**. Generally, the hard SAIs think between 20 to 40 seconds, the medium SAIs think between 35 to 55 seconds and the easys are greater than 55 seconds. *(Note: It takes a good bit of testing to determine which skill levels the various AIs are.)*

Next, the **sai_wait_threshold** determines the number of minutes that an SAI will wait to accumulate gold to achieve a higher cost objective. In *Immortal Sovereigns*, the default is 0; in *Ahriman's Gift*, the default is 3 minutes.

The next two items, the **influence_range** and the **influence_decay_rate**, are ignored outside of the default.ini. They help the profile decide how far to calculate out the other players' influences. The **influence_range** decides how far influence (based on combat values) spreads and the decay rate tells how much influence is transmitted from one region to the next. For example, if a profile having a range of 3 and a decay rate of 0.5 sees a region with an influence of 10.0 for some other player, the four surrounding regions would have a residual influence of 5.0. The next farther out ring of eight regions has an influence of 2.5. The third and final ring has an influence of 1.25. All other regions have 0 influence due to this influence source. Influence from multiple sources act upon regions in a cumulative fashion.

Finally, the last two items in GeneralSettings are the **early_game_threshold**, which indicates the population level (company limit) that defines the early game for the SAI and the **early_game_full_recompute_time**, which overrides the **full_recompute_time** until the **early_game_threshold** has been met.

2.2 GoalPriorityList

The next section, GoalPriorityList, sets the base priorities of each goal. Some of the goal values are used without calculations or modification, such as **explore**. **Distance** is always multiplied by the number of regions and/or tiles separating the company from its target. **Combat** is always multiplied by the combat values or difference between the attacker's and defender's combat value. **Building_defend** and **building_attack** are for each friendly and enemy building in a region. **Script** is not currently used, but was planned to be for campaign specific and/or map specific goals. **Repeat** is used as a bonus when giving the same goal twice (although it currently does not function). **Guard** is the base value to guard a region. **Reinforce**, like **repair**, goes off percent damage, this time for companies, not buildings. **Reinforce** is the priority for moving injured units into an allied supply zone. **Repair** is multiplied by the percent of damage done to the building (i.e., a town at one hit point has nearly the full repair value).

The other goals are for selecting construction goals. **Build_city**, **build_outpost**, and **build_mine** are the base values to build a city, an outpost and a mine, respectively. **Upgrade_city** will usually happen if a player can afford it, so this value doesn't do too much. **Build_structure**, **upgrade_structure**, and **commission_unit** are the base values for a city to do the action, but they are all affected by the resource income of the player. **Upgrade_building** controls whether or not the SAI will upgrade any non-city building, like an outpost to a fort.

2.3 GoalPriorityBonus

The GoalPriorityBonus section sets the base randomness for each goal, its **fuzziness**. Also set here is the **destroy_captured_base_percent**, which is only used when the SAI has a building with no militia or nearby friendly units and lots of enemy units in the area. The **campaign_attack_delay** and **scenario_attack_delay** are how many seconds a respective profile will wait issuing an attack order in campaigns and scenarios, but it will defend itself. The default setting of 2 minutes (120 seconds) is too short to do much, since it normally takes longer for a unit to reach full strength. Finally, the **percent_engage_captain** is the percent an SAI will set the captain or non-mage hero to engage rather than command. Non-mage heroes are defined as those heroes who have less than 2 spells. Also, heroes below 1/3 of their max HP are always set to command. **Wake_hero_threshold** sets the minimum amount of gold income necessary before an SAI will wake one of its heroes. Finally, **percent_borrow_for_upgrade** determines the percentage chance for upgrading on a whim by stealing from other accounts (accounts are how the SAI handles its economy).

2.4 MaxEvalGoals

MaxEvalGoals is used to calculate the maximum number of each goal group to use when doing the calculations. The number is multiplied by the number of settlements the player has. **Explore**, **attack**, **guard**, **construct**, and **upgrade** all indicate the maximum number of goals per settlement in each of those areas that the SAI will consider.

2.5 MaxExecGoals

MaxExecGoals is the maximum number of goals to actually execute in each goal group. Some, like **guard** goals, will tend to be larger because when other goal types fail to find a target they default to **guard** goals. **Explore**, **attack**, **guard**, **construct**, and **upgrade** all indicate the maximum number of goals per settlement that the SAI will execute.

2.6 StrengthRatios

StrengthRatios are the most important thing for figuring out attacks and other goals. The **matching_ratio_min** and **matching_ratio_max** are the minimum and maximum ratios of my total combat value to the enemy's total combat value (CV) when the SAI decides if it will attack

or not. If the SAI's total CV is 50 and the enemy's is 20, the ratio will be 2.5.
Building_bonus_min and **building_bonus_max** are the min and max extra CV the SAI needs to bring along for every building in the target attack region. The **explore_min**, **explore_max**, **guard_min**, and **guard_max** min and max's are likewise used for the minimum and maximum total CV needed to explore or guard each goal.

3 Economic Allocation and Priorities

3.1 GoldRatio

The GoldRatio is the division of the gold vault into build, upgrade, and unit accounts. The **build** account is for upgrading cities and outposts and building mines and outposts. The **upgrade** account is for adding and upgrading components to settlements. The **unit** account is for awakening heroes and commissioning companies. Maintenance is paid before any gold is allocated to the various accounts. The gold ratios do not have to add up to zero, they will be normalized and any negative values will be made positive.

There is a rather big exception to the division of gold. When the SAI is able to upgrade a settlement to the next level (town to city, etc.), and it can afford it even if there isn't enough in the build account, it will pull money from the other two accounts until it can buy it. There are other situations when money will be reallocated between accounts. For example, when at the maximum number of companies, some money will be reallocated from **unit** to **upgrade.**

3.2 BuildTemplates (Ahriman's Gift Only)

Build templates are ignored in default.ini, but can be specified for each individual SAI. They tell the SAI to try to construct components and upgrades in a settlement in a specific way. There is no restriction on the number of templates that can be specified. Likewise, there are no restrictions on the number of components and upgrades that can be specified. Specifying an upgrade automatically requires the basic **component** (i.e., specifying "CarpentryGuild" means the Woodmill will be built first). You can specify more than 7, and you can specify more than one upgrade for a base component; it will be impossible to satisfy, but the SAI will add a bonus to each component's priority.

3.2.1 Filters

Maxinstances tells the SAI to try to assign this template to the number of buildings specified in this field. Not to be confused with **mincities** and **maxcities**.

Mineconomy, **maxeconomy**, **mincities**, **maxcities**, **race**, **onlyfaction**, **exceptfaction** are all filters or pre-requisites for the SAI to consider a particular settlement as a candidate for the template. **Onlyfaction** allows you to specify the template for a building of one particular faction: Ceyah, Council, Nationalist, and Royalist. **Exceptfaction** does the opposite of **onlyfaction** by specifying that faction which cannot use this template. **Mincities** and **maxcities** refers to the total number of settlements owned by the player. **Mineconomy** and **maxeconomy** refers to the gold rate of the player (not the gold stockpile), so "mineconomy = 1" means the template should only be used with a positive economy.

Race allows you to set a race for the template in question: possible races are Drauga, Gauri, Haroun, Slaan, and Mareten (which includes Ceyah).

Bonus is the bonus priority applied to each matching component or upgrade. The SAI still applies other logic taking into account the state of its economy.

3.3 StructurePriorityList

StructurePriorityList lists components and the base priority for each structure type. They are influenced by the surplus or deficit of their various upkeep or production resources. For example, considering a temple with a rate of 0 stone and 0 mana will receive a –3 penalty. In general, setting a value in the StructurePriorityList over 15 can cause the SAI to make inappropriate decisions when selecting the components to build.

3.4 StructureUpgradePriorityList

StructureUpgradePriorityList works similarly to StructurePriorityList, listing all available upgrades to the base components, but upgrades receive a 15-point bonus when the SAI is stressing upgrades. This is decided by the **target_upgrade_ratio** value in the StructurePriorityBonus section.

3.5 StructurePriorityBonus

Target_upgrade_ratio comes into play when a smaller percentage of the components are upgraded. When that ratio is not met, a 15-point bonus is added to the upgrade priority. The **fuzziness** factor is a random value added to all component and upgrade values. The **wall_bonus_time** is used when the settlement is attacked; for this duration, there is a bonus for making a wall (with respect to any other component).

3.6 CompanyFrontPriorityList & CompanyFrontRepeatPenalty

The CompanyFrontPriorityList lists frontline units and the base priority for each unit. For every company of that type in play, the penalty in CompanyFrontRepeatPenalty is assessed to reduce the priority of companies. When a priority is set to -100 that is an invalid priority and this unit will never be built. Note that the internal name of units used by the SAI profile does not always match the displayed name in game, so you should always consult the default.ini.

When considering the company type to commission, there are several factors that influence the SAI's decision. The primary factor is the economic situation. This should be considered when assigning priorities – they should never be outrageous compared to the other units. For example,

a Shadow Beast company at a current rate of 0 mana receives approximately a –20 penalty. If the Shadow Beast priority was 100 and the repeat penalty was –10, the SAI would blindly commission approximately 6-7 companies of Shadow Beasts before it became favorable to commission something else. This would damage its economy, and would probably break its play.

3.7 CompanyFrontPriorityBonus

The CompanyFrontPriorityBonus only has one setting, the **fuzziness** factor.

3.8 CompanySupportPriorityList & CompanySupportRepeatPenalty

The CompanySupportPriorityList and CompanySupportRepeatPenalty function just like the CompanyFrontPriorityList and CompanyFrontRepeatPenalty except for the support units. These priorities are only considered if the SAI decides to commission a support company (see below).

3.9 CompanySupportChoiceRatio

The ratios of types of companies are laid out in the CompanySupportChoiceRatio section. They are **support** (specialty support units like Archmages and Rangers), **complementary** (archer support), **homogenous** (6 front line units), and **none** (a short company of 4 units). The decision making process slides downhill. For example, if the SAI decides to make a support company, but does not have any support units, it will instead try to make a complementary company. In practice, the number of homogeneous companies will be a little higher than the ratio assigned here. The economic situation may also result in short companies (4 units). The SAI will commission the best company it can afford.

3.10 CompanySupportPriorityBonus

CompanySupportPriorityBonus has all the miscellaneous settings for support units. Both engineers and settlers are handled a little differently. Settlers have a straight percentage to be 6 versus 4 unit companies, and that is covered by **percent_big_settler**. Engineers have both a chance to be 6 unit companies (**percent_big_engineer**) and a chance to use the regular support unit's logic (**percent_mixed_engineer**). The **matching_support** is a penalty or bonus for getting two of the same type of specialty support unit in a company. **Fuzziness**, of course, is the random factor added to the support calculations.

4 Political Tendencies & Treaties

4.1 PoliticalTendencies

The PoliticalTendencies section contains all the various shifting of the opinions the SAI player has for other players. **Aloof** is the percentage drift towards the base opinion of another player calculated each think turn for that player (it is best to keep it less than 0.1, otherwise it will be difficult to obtain or maintain political relations). **Levelheaded** is the reduction or increase of the effect of ALL events, regardless of who does them and whether they were good or bad things. **Chaotic** is the random factor that occurs each think turn. Note, this number is a spread, so a value of 0.1 (10%) would result in a random shift between -5% (-0.05) and +5% (+0.05) each turn. **Loyal** is the percent increase of the opinion effect from good events by an ally and the corresponding reduction of the opinion effect from bad events by an ally. **Vindictive** is sort of the opposite of **loyal**, increasing an enemy's bad event effects and reducing good ones. **Altruistic** and **suspicious** are exact opposites. The former increases the opinion effects of ALL good events, no matter who did them, and reduces all bad ones. The latter increases the effects of all bad events and reduces all good ones. Finally, **greedy** affects the impact of tribute on a player's opinion. The number here represents the shift away from 100%, so a 0 value would leave the effect of tribute unchanged. A -1 value (-100%) would lead to NO effect for tribute (100% + -100% = 0%) while a +1 value would DOUBLE the effect of tribute (100% + +100% = 200%).

4.2 PoliticalTendenciesBonus

This section is not used in the current build.

4.3 PoliticalTreatyOfferThreshold & PoliticalTreatyAcceptThreshold

PoliticalTreatyOfferThreshold and PoliticalTreatyAcceptThreshold are the thresholds where the SAI will offer and accept treaties. For example, in the default.ini, if the SAI opinion of you drops below -50, it will declare war, but if the SAI is already at war with you, it won't offer peace until its opinion of you has risen to +5. Since declarations of war and cancellations of alliances do not require a response, those are not listed under PoliticalTreatyAcceptThreshold. What is there is the opinion the SAI has to have of a player to accept the player's offer of peace or alliance. Please note the large spread between **DeclareWar** and **OfferPeace**. This is to prevent the SAI from yo-yo-ing offers, like declaring war, then immediately offering peace or vice versa.

4.4 Political Treaty Bonus

There are three items in PoliticalTreatyBonus. **Fuzziness** is the random factor used when evaluating treaty offers. **Time_between_treaties** is how long to wait between sending new treaty offers to a specific player. Last but not least, **time_to_consider_treaty** is how long to sit on a treaty offer before responding to it.