Основная идея:

1. Создаётся цепочка из задач, которые должны быть выполнены для победы в игре. Текущей задачей для разработки плана выбирается невыполненная задача с наивысшим приоритетом. От заданной цепочки задач будет зависеть используемая стратегия и сложность бота.

2. Планировщик получает на вход актуальную задачу. Также у него есть доступ к текущему состоянию. Для поиска последовательности действий используется алгоритм А\*:

1) В начальном состоянии производится поиск всех доступных действий (раскрытие вершины), для каждого действия рассчитывается стоимость, следующей раскрывается вершина (новое состояние) с наименьшей стоимостью перехода;

2) При раскрытии вершины рассматриваются только доступные в данном состоянии действия;

3) Стоимость перехода складывается из стоимости действия в текущем состоянии (учитываются значения некоторых параметров, время на совершение действия и время, прошедшее с начала игры) и эвристической функции оценки (учитывается достигаемая цель).

3.1) Влияние параметров состояния на стоимость действия:

Самый простой способ оценки действия – это анализ тех параметров, на которые она непосредственно влияет. Например, если действие связано с наймом юнита, то чем больше таких юнитов, тем выше будет стоимость такого действия, и приоритет, возможно, будет отдан улучшению этих юнитов.

3.2) Влияние времени выполнения действия:

Чем больше времени требуется на найм, постройку и прочие действия, тем выше будет его стоимость.

3.3) Влияние времени, пройденного с начала игры:

Чем больше времени прошло с начала игры, тем большее предпочтение должно отдаваться действиям военного характера, нежели экономического, для обороны от развившегося соперника и последующей атаки на него.

3.4) Эвристическая оценка с учётом поставленной цели:

Если цель связана с постройкой или с наймом, предпочтение будет отдаваться увеличению экономической мощности; если цель связана с обороной базы, предпочтение будет отдаваться найму боевых юнитов, их перемещению, невзирая на ресурсные траты.

4) По достижении целевого состояния планировщик восстанавливает последовательность действий, пройденных от начального, и публикует их на доске объявлений.

5) Когда план опубликован, планировщик приступает к формированию нового, предполагая, что текущим состоянием является достигнутое действиями предыдущего плана. Параллельно планировщик прослушивает сообщения о выполнении предыдущего плана. Если он срывается, планировщик прерывает составление плана к следующей задаче и перепланирует достижение предыдущей.

3. Доска объявлений (Blackboard) служит для хранения опубликованной планировщиком последовательности действий. Каждое действие может быть выполнено одним из менеджеров, которые напрямую управляют юнитами.

4. Менеджеры:

1) BuildManager – отвечает за строительство зданий и их улучшение, выбирает рабочего и место для строительства;

2) ResourceManager – отвечает за сбор ресурсов, управляет рабочими для сбора (не выполняет действия с доски объявлений, а только задействует рабочих, не занятых другими менеджерами);

3) TrainManager – отвечает за выбор здания для найма и непосредственный найм юнитов;

4) ScoutManager – отвечает за разведку территории, поиск новых ресурсных месторождений и баз противников;

5) ProtectManager – отвечает за оборону базы, оценивает опасность нападающих противников и выбирает наиболее подходящих для отражения атаки юнитов;

6) AttackManager – отвечает за управление боевыми подразделениями, их перемещением и атакой.

5. Действия:

1) TrainAction - Для тренировки каждого типа юнитов (на каждого юнита по одному действию).

Условия для выполнения:

а) Наличие необходимого кол-ва ресурсов

б) Наличие здания, в котором производится найм

в) Особые требования к некоторым юнитам (требуется изучение какой-либо технологии или постройка определённого здания).

Менеджер: TrainManager.

2) BuildAction - Для постройки зданий (на каждое здание по одному действию).

Условия для выполнения:

а) Наличие необходимого кол-ва ресурсов

б) Наличие рабочего для строительства

в) Особые требования к некоторым зданиям (требуется изучение какой-либо технологии или постройка определённого здания).

Менеджер: BuildManager.

3) ImproveAction – Для изучения технологий (на каждую технологию по одному действию).

Условия для выполнения:

а) Наличие необходимого кол-ва ресурсов

б) Наличие здания, в котором проводится изучение

в) Доступность (изучены технологии, открывающие данную).

Менеджер: BuildManager.

4) GatherResourceAction – для всех менеджеров вводится режим ожидания, пока не наберётся достаточное кол-во ресурсов для следующего действия.

Условие выполнения: увеличение кол-ва требуемых ресурсов в единицу времени больше нуля.

Менеджеры: все.

5) ProtectBaseAction – Защита базы от нападения.

Условие: нападение на базу.

Менеджер: ProtectManager.

6) AttackEnemyBaseAction – Атака базы противника.

Условия:

а) Существует хотя бы одна обнаруженная вражеская база

б) Суммарная мощность (количество атакующих юнитов с учётом их улучшений) выше определённого значения

в) Экономические показатели на определённом уровне

г) Суммарная мощность (количество оборонительных зданий и атакующих юнитов с учётом их улучшений) резервов для обороны выше определённого показателя.

Менеджер: AttackManager.

7) ScoutAction – Разведка карты (по одному действий на каждое условие).

Условия:

а) Большая область тумана войны

б) ИЛИ нет обнаруженных баз противника

в) ИЛИ требуется поиск новых ресурсов.

Менеджер: ScoutManager.

Возможностью для непосредственного управления юнитами и зданиями (выделение, перемещение, атака и прочими) обладают менеджеры, каждый из которых самостоятельно выбирает способ выполнения действия, размещённого на доске объявлений.