Первая задача.

Существующий сменно-суточный бригадный планировщик УТХ (см. статью по ИСУЖТ-2014) работает по принципу изолированной локальной оптимизации. То есть оптимизируется локальная для интервала полезность назначения бригады на каждый трехчасовой интервал. Этот подход дает глобальное решение только при монотонности функции полезности бригады по трех-часовым интервалам. При этом требуется второй проход оптимизационного алгоритма, чтобы оптимально учесть количество разрешений у бригады на работу по разным участкам обслуживания.

Можно изменить алгоритм так, чтобы глобальная сумму всех полезностей при назначении бригад была оптимальной вне зависимости от монотонности (и вообще вида) функции полезности.

Для этого можно использовать алгоритм асинхронных параллельных аукционов с учетом однотипности, описанный в статье Бертсекаса 89 года. Алгоритм реализован в программном коде работ по УТХ 2014 года, но потребует адаптации для данной задачи.

24.12.2014. Сформулированы следующие подзадачи для решения большой задачи 1:

1.1. Организовать запуск планировщика в любое время суток, так, чтобы планирование производилось на сутки вперед, начиная не от начала следующих ж.д. суток, а от начала следующей 12-часовой смены. (Решена 15.12.2014)

1.2 Организовать передачу на вход планировщику наряд-заданий на явку бригад не по сменам, а по трехчасовым интервалам. (Решена 22.12.2014)

1.3 Организовать запуск планировщика в любое время, так, чтобы планирование производилось на сутки вперед , начиная не от начала следующей за запуском 12-часовой смены, а от начала следующего за запуском 3-часового интервала. (Решена 23.12.2014)

1.4 В существующем коде избавиться от двойного прогона алгоритма планирования для каждого 3-часового интервала. Таким образом, планировщик будет работать некорректно, не оптимизируя задачу, но он составит скелет для дальнейшей разработки. (Поставлена 24.12.2014. Решена 24.12.2014.)

1.5 В коде пункта 1.4 избавиться от векторного вида функции полезности, задавая ее в виде скаляра, учитывающего с заданными коэффициентами все компоненты вектора полезности и скаляр полезности второго прогона, удаленного в п.1.4. Таким образом, планировщик будет снова работать корректно, оптимизируя задачу, при условии монотонности новой скалярной функции полезности от номера 3-часового интервала. (Поставлена 24.12.2014. Трудоемкость – 4 часа)

1.6 А.Такмазьяном на Jason реализовано решение транспортной задачи с помощью «задачи о назначениях с одинаковыми объектами» (TAP SO – task assignment problem with similar objects), описанной на с. 74 в статье Бертсекаса 89 года. В коде А.Такмазьяна в агенте “transportation.asl” изменить а) интерфейс и б) план +!start так, чтобы код реализовывал в точности задачу TAP SO (Поставлена 29.12.2014. Трудоемкость - 16 часов)

1.7 Избавиться от локального планирования, реализуемого в настоящем коде планировщика агентами team и part, изменив их код так, чтобы они решали «задачу о назначениях с одинаковыми объектами» (TAP SO). (Поставлена 29.12.2014. Трудоемкость – 20 часов.)

Вторая задача (пока отменяется)

Существующие текущие планировщики УТХ бригад и локомотивов требую на своем входе план подвязки бригады, заступающей на работу в данном депо, к локомотиву и поезду. Существенные данные при этом описываются структурой, выражающей связь бригады, времени отправления и (при необходимости) локомотива.

Известными являются: а) список локомотивов, доступных в данный момент в депо; б) план явки бригад по трех-часовым интервалам, сформированный ранее сменно-суточным планировщиком (см. задачу 1); в) список всех бригад данного депо, находящихся в данный момент на отдыхе; г) список поездов, следующих через данное депо, с подвязанными локомотивами.

Необходимо написать планировщик, образующий указанную в первом абзаце структуру-связь, учитывающий при этом возможную смену локомотива из данных (г) на локомотив из данных (а), подвязывающий бригаду из данных (б), а при недостатке таких бригад, из данных (в).

Все необходимые алгоритмы, интерфйсы и коды существующих планировщиков предоставляются.