Поиск с чередующимися окрестностями

Пусть последовательность σ задает некоторое допустимое решение задачи. Определим следующие окрестности.

• Окрестность $N_1(\sigma)$ содержит все последовательности, получающиеся из σ либо перемещением одной работы на новое место, либо взаимной перестановкой двух работ.

• Окрестность $N_l(\sigma)$, $l > 1$ содержит все решения, получающиеся из данного не более l последовательными переходами к соседним решениям по окрестности $N_1(\sigma)$.

Назовем *блоком* несколько подряд идущих работ последовательности σ.

• Окрестность $N^b(\sigma)$ содержит все последовательности, получающиеся из σ взаимной перестановкой двух непересекающихся блоков, возможно разных размеров.

• Окрестность $KL_k(\sigma)$, $k > 1$ является аналогом окрестности Кернигана-Лина [5] и строится по следующему правилу:

1. Найдем наилучшее решение σ' в окрестности $N_1(\sigma)$. Пусть это решение получается либо перемещением работы с позиции j', либо взаимной перестановкой работ в позициях j' и j''.

2. Положим $\sigma = \sigma'$.

Будем повторять шаги 1 и 2 не более k раз, используя позицию j' или пару $(j',\ j'')$ не более одного раза. В результате получим набор последовательностей. Каждую из них будем считать соседней для исходного решения σ.

Вместо окрестности N_1 для построения окрестности Кернигана-Лина можно использовать окрестность N^b. Эта окрестность шире и позволяет находить решения, которые сложно получить используя окрестность N_1. Окрестность N^b хорошо подходит для этой задачи, так как несколько работ могут создавать удачную подпоследовательность — блок, и расписание нужно изменять не для одной работы, а для всего блока. Просмотр окрестности N^b — трудоемкая процедура, поэтому далее будем рассматривать блоки из не более чем трех работ.

Поиск с чередующимися окрестностями (VNS) был разработан Хансеном и Младеновичем в 1997 [2,4]. Основная идея предлагаемого метода заключается в использовании разнородных окрестностей и их систематической смене в процессе локального поиска. Общая схема этого метода для решения данной задачи представлена ниже:

1. Найти начальное решение σ. Задать параметры l_{max}, k_{max}.

2. Пока не выполнен критерий остановки, делать следующее.

(a) Положить $l := 1$.

(b) Повторять пока $l \leq l_{max}$.

i. Выбрать решение $\sigma' \in N_l(\sigma)$ случайным образом.

ii. Применить локальный спуск сначала по окрестности N_1, а затем по окрестности $KL_{k_{max}}$. Обозначим найденный локальный оптимум σ^*.

iii. Если $F(\sigma^*) < F(\sigma)$, то $\sigma := \sigma^*$ и перейти на шаг 2(a), иначе положить $l := l + 1$.

(c) Применить локальный спуск по окрестности $KL_{k_{max}}(\sigma)$, используя окрестность N^b вместо N_1. Обозначим найденный локальный оптимум σ^*. Если $F(\sigma^*) < F(\sigma)$, то $\sigma := \sigma^*$.
Перейти на шаг 2(a).

3. Предъявить решение σ как результат работы алгоритма.

Начальное решение выбирается случайным образом. В качестве критерия остановки используется либо число полученных локальных оптимумов, либо общее время счета.