Существует пласт задач. Обычно это задачи с дискретно меняющимися переменными для которых отсутствуют эффективные алгоритмы решения.

Что такое эффективный алгоритм чем он отличается от не эффективного?

Эффективный алгоритм это тот алгоритм время решения которого полиномиально зависит от размерности задачи т.е. можно выразить полиномом время

Т1 =

n-количество переменных

Если алгоритм экспоненциально зависит от размерности задачи то это не эффективный алгоритм.

Т2 = a en

Доказано что алгоритмы с дискретно меняющимися переменными в частности например с булевыми переменными является не эффективным т.е. полный перебор.

Перебор при решении задачи коммивояжора не эффективен. Если есть n городов то число перестановок = n!, а это выражается через экспоненту.

Задача о ранце нет эффективного решения, перебор это 2n выражается через экспоненту

И тогда в середине прошлого века появились методы типа ветвей и границ. Почти в то же время было предложено динамическое программирование. Они относятся к разным классам алгоритмов.

Метод ветвей и границ это построение дерева и спуск в лучшем направлении. А динамическое программирование это ветвление по дереву и обрубание плохих направлений.

Композитные алгоритмы пытаются совместить эти два метода. Когда мы совмещаем разные методы каждая итерация занимает больше времени, потому что мы не просто вычисляем оценки мы еще проверяем добавочно что можно обрубить. Если ничего нельзя обрубить то выходит что этот алгоритм работает хуже чем другие. Он спускается быстрей если только можно эффективно обрубать плохие направления.

Возможность обрубить плохие направления зависит от специфики задачи, для одной задачи это работает хорошо для другой плохо. Проверить эффективность можно только экспериментально.

5 мин



8z1+4z2+6z3 → max

4z1+5z2+3z3 ≤ 11



8z1+4z2+6z3+10z4+3z5 → max

4z1+5z2+3z3+8z4+7z5 ≤ 11

31

23

27

31

23

19

25

21

27

17

∞

23

19

13

15

∞

17

∞

11

∞

∞

13

∞

9

19

7

14

∞

16

∞

17,17,14

14,14,7

19,19,18

16,16,11

27,24,15

17,14,7

11,8,4

21,18,12

21,8,4

9,6,3

19,16,11

23,10,8

17,4,5

19,6,3

13,0,0

27,14,7

25,12,9

31,18,12

31,12,9

27,8,4

23,4,5

19,0,0

31,8,4

23,0,0

31,8,4

31,12,9

27,8,4

27,14,7

21,8,4

25,12,9

31,18,12

23,0,0

31

∞

27

27

21

25

23

31