Богдан Уладзіслаў

ФПМІ, 3 курс, 3 група

ДЗ 3

NP-складанасць задачы $1||\sum w_j u_j$

Будзем даказываць NP-паўнату адпаведнай задачы распазнавання - з доказу будзе вынікаць і NP-складанасць нашай задачы. Задача распазнавання фармулюецца наступным чынам: для кожнай работы зададзеныя p_j, d_j, w_j, трэба пабудаваць расклад для адной машыны, пры якім $\sum w_j u_j < y = const$.

Доказ будзем праводзіць, выкарыстоўваючы эталонную задачу РАЗБІЦЦЁ ($\sum a_i = 2A$, ці існуе разбіццё a_i на два мноствы, кожнае з якіх дае A у суме). Пакажам, што калі задача РАЗБІЦЦЁ мае рашэнне, то будзем мець ТАК-прыклад у нашай задачы распазнавання і калі задача РАЗБІЦЦЁ рашэння не мае, то будзе мець НЕ-прыклад.

Прыклад нашай задачы распазнавання будуецца па наступных правілах: $p_j = a_j, w_j = a_j, d_j = A, y = A$ (будуецца за лінейны час).

1. Няхай задача РАЗБІЦЦЁ мае рашэнне. Рашэнне існуе - таму ёсць разбіццё на мноствы E_1 і E_2. Не губляючы агульнасці, усе патрабаванні з E_1 паспеюць выканацца да дырэктыўнага тэрміну, таму адпаведныя $u_j = 0$; патрабаванні з E_2 да дырэктыўнага тэрміну выканацца не паспеюць - будзем мець штраф $\sum w_j u_j = \sum a_j u_j = A \leq y$. Маем ТАК-прыклад задачы распазнавання.

2. Няхай задача РАЗБІЦЦЁ не мае рашэння - не існуе разбіцця на два мноствы з аднолькавай сумай роўнай A. Як і ў папярэднім пункце - ставім перагародку ў момант часу A (у папярэднім пункце гэтая перагародка ніколі не перашкаджала, бо адно з патрабаванняў скaнчвалася у момант часу A, а наступнае ў гэты ж момант пачыналася). Адсутнасць рашэння задачы РАЗБІЦЦЁ гарантуе, што перагародка перашкодзіць аднаму з патрабаванняў пачацца ў моманту часу $< A$ і скончыцца ў момант часу $> A$. Будзем мець прастой машыны (мінімум у адну часавую адзінку) непасрэдна перад момантам A. Гэта, у сваю чаргу, гарантуе, што патрабаванні, якія пачнуцца ў момант часу A і пазней, у суме па часе будуць $> A$. Улічваючы правілы будування прыклада, маем: $\sum w_j u_j =$[$u_j = 0$ для ўсіх патрабаванняў, што пачаліся і скончыліся раней за A.] $= \sum a_j u_j > A = y$. Маем НЕ-прыклад задачы распазнавання.

Паказалі палінаміяльную прывадзімасць задачы РАЗБІЦЦЯ да нашай задачы распазнавання, таксама задача распазнавання прыналежыць класу NP (бо пры наяўнасці прыклада (сертыфіката) праверка, ці гэта ТАК- ці НЕ-прыклад, здзяйсняецца за палінаміяльны час) - таму нашая задача распазнавання NP-поўная. Адсюль вынікае NP-складанасць пачатковай задачы аптымізацыі. Што і трэба было даказаць.