Анализ на Задача „Мечо Суперпух”

Макар и финалът да е само четири часа, съвсем естествено последната задача за тази година е доста сериозна и предоставя голямо поле за различни подходи, идеи и подобрения. Освен всичко, тя не позволява твърде лесно да се спечелят точки, като провокира състезателите да положат доста усилия за извеждането на какъвто и да е коректен отговор.

Най-удачно е ситуацията от задачата да се представи в теория на графите. Всеки пункт е връх, връзките между тях са ребра, като има две функции за тежестите им – *t(x)* и *p(x)*, които са съответно времето и парите, необходими за преминаване през ребро *x*.

За да може да се отговори коректно на всички заявки, трябва първо да се намерят минималните по време (измервайки теглата с *t(x)*) пътища. По този начин, за всяка заявка ще има намерен валиден маршрут, който ще я изпълнява навреме, макар и не за най-добрата възможна цена. Един от начините това да стане е като се намерят най-кратките разстояния между всеки два върха чрез алгоритъм на *Флойд*[1]. За съжаление сложността му – $O(N^3)$ – е твърде голяма за ограничението от $N<4001$ и времето за изпълнение на програмите не винаги е достатъчно. Това може да се оптимизира след като се забележи, че са необходими пътищата между най-много *K* $(K<1001)$ различни върха и така да се използва алгоритъм на *Дейкстра*[2] за намирането им. При използването на много по-удачната, предвид максималния брой ребра $(M<80001)$, реализация с приоритетна опашка, се стига до сложност $O(K.M.\log M)$. Едва тази стъпка води до програма, която е способна да изведе валидни резултати на всички тестове в рамките на зададеното време.

Следващият много правилен ход от стратегическа гледна точка, имайки предвид оскъдните четири часа за работа на състезателите, е да се модифицира вече написания алгоритъм на Дейкстра да използва другата функция на теглата – *p(x)*. Така за всяка поръчка може да се намери оптималния като цена маршрут и, ако той се окаже достатъчно бърз, да се използва вместо намерения в първата стъпка. Това е една не малка оптимизация, както личи от факта, че подобно решение донесе на Александър Георгиев първото място.

След като е готова тази база, върху която да стъпи крайното решение, следва трудният избор как оставащото време да се разпредели между възможностите да се оптимизират още пътищата, да се съединяват такива с припокриващи се части и, разбира се, да се тества.

Последната възможност за пореден път се оказа най-удачната на финала тази година, но какви са опциите в другите две? Подобряването на пътищата е може би по-лесната задача, защото е по-естествена и по-близка до неща, които много често се правят. При малки графи съществува идея, която гарантирано намира най-доброто решение като разширява графа, умножавайки върховете по възможните моменти във времето. Това, разбира се, не би работило на по-големите тестове, където възможните подходи са различни непълни търсения и евристика. Една от най-удачните може би е да се изберат най-скъпите ребра в намерените маршрути и да се търсят техни алтернативни пътища на по-ниска цена.

Сливането на няколко маршрута в един е много по-дълбок проблем, най-малкото защото влошаването на даден маршрут може да доведе до по-добро крайно сливане. Тази фаза донесе и най-много проблеми на участниците, които решиха да я използват в решенията си.

Връзки

[1] http://en.wikipedia.org/wiki/Floyd-Warshall_algorithm
[2] http://en.wikipedia.org/wiki/Dijkstra's_algorithm