ОПТИМИЗАЦИЯ СХЕМ ОБРАБОТКИ БАЗ ЭКОНОМИЧЕСКИХ ДАННЫХ БОЛЬШОЙ РАЗМЕРНОСТИ

Шахновский Ю.С.
Национальный технический университет «Харьковский политехнический институт», г. Харьков

В последнее время растет интерес к обработке массивов экономических данных с целью выявления в них скрытых закономерностей (т.н. добыча данных, data mining). При этом обрабатываемые базы данных могут иметь весьма существенную размерность (высокочастотные данные по динамике биржевых котировок, кассовые транзакции в супермаркетах, данные по обращениям к рекламным баннерам в интернете и т.д.). В таких случаях часто целесообразно разбить общую базу данных большой размерности на ряд подмножеств с тем, чтобы элементы каждого подмножества обрабатывались в дальнейшем по своим алгоритмам.

Обычно эта задача решается последовательными проверками – на принадлежность первому подмножеству, второму и т.д. Элемент, для которого подмножество найдено, в дальнейших проверках не участвует. Если количество подмножеств велико, то время, затрачиваемое программой на разбиение, может быть намного больше, чем время дальнейшей обработки каждого элемента.

Общее время, потраченное на разбиение по классам всех элементов, зависит от порядка, в котором будут проводиться проверки принадлежности к каждому из выделенных подмножеств. Выгодно ставить подмножества, которым соответствует большое количество элементов, в начало списка проверок. Для случая, когда условия проверок описывают непересекающиеся подмножества, оптимальным порядком будет сортировка проверок по отношению $v(i)/t(i)$, где $v(i)$ – это частота, с которой встречаются элементы, принадлежащие классу i, а $t(i)$ – это время, затраченное на проверку принадлежности классу i одного элемента. Такой простой алгоритм не будет работать, если правила, задающие принадлежность классу, описывают пересекающиеся подмножества. В этом случае принадлежность элемента классу зависит от порядка проверок. И для проверок, имеющих пересечение, изменять начальный порядок нельзя.

Отношение «имеет пересечение» удобно задавать в виде направленного графа, где вершины соответствуют подмножествам, дуга между парой вершин – непустому пресечению подмножеств, а направление дуги задает порядок между подмножествами в начальной программе. Полученную задачу оптимизации на графах можно решать для графов разного вида. Был найден алгоритм ее решения за полиномиальное время для графов вида дерева. Для графов общего вида были разработан алгоритм ветвей и границ для получения точного решения и эвристические правила для поиска решения, близкого к оптимальному.