

СРЕДСТВА ПОДДЕРЖКИ РАЗРАБОТКИ
ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ АЛГОРИТМОВ

С.И.Катков, И.В.Поттосин

*Институт систем информатики им. А.П.Ершова СО РАН
Новосибирский государственный университет
Новосибирск, Россия*

Собственно параллельное программирование предполагает разработку программ, ориентированных на конкретную архитектуру из множества весьма различающихся архитектур. Создание же алгоритма (а не программы) предполагает выбор (и фиксацию) некоторой модели адекватной алгоритму архитектуры. Даже ограничившись крупнозернистым параллелизмом (параллелизмом процессов) и соответствующими ему многопроцессорными архитектурами, практически нереально рассчитывать на создание «универсального» параллельного алгоритма, в равной мере приспособленного для любой из таких архитектур.

С другой стороны, создание алгоритма на одном из производственных языков параллельного программирования вносит в описание алгоритма много технических деталей, затрудняющих понимание и обсуждение идеи алгоритма, его перепрограммирование на другую, хотя и близкую, архитектуру. Фонд параллельных алгоритмов хотелось бы создавать на некотором языке, позволяющем абстрагироваться от деталей конкретных архитектур.

Естественно представить такую методологию создания параллельных алгоритмов, при котором алгоритм фиксируется на некотором достаточно ясном и избавленном от детализации архитектуры языке программирования – языке публикаций, исполняется (отлаживается и отрабатывается) на широко доступной архитектуре, а затем при необходимости «погружается» в подходящую многопроцессорную архитектуру в виде программы на промышленном языке программирования. Язык публикаций выступает здесь и в своем качестве – как передатчик информации от человека к человеку -, а также как основа выражения мысли (для самого создателя) и как язык исполняемых спецификаций.

Разработанная в ИСИ им.А.П.Ершова СО РАН и НГУ система параллельного программирования СуперПаскаль [1] предназначена для поддержки именно такой идеологии.

Сам язык СуперПаскаль [2], предложенный как публикационный язык параллельных алгоритмов, представляет собой представительное, достаточное для вычислительных алгоритмов подмножество Паскаля – языка, используемого для публикации последовательных алгоритмов -, расширенное простыми и ясными средствами параллелизма. Средства эти позволяют породить совокупность параллельных процессов – разнотипных с помощью оператора `parallel` и одинаковых (с меняющимся процессным индексом) с помощью оператора `forall`. Родственные процессы связываются каналами – буферами на одно сообщение, возможно, разнотипное. Вся синхронизация осуществляется

передачей сообщений. В отличие от близкого по духу языка Мульти Паскаль [3] конфигурация процессов (т.е. в конце концов модель мультипроцессорной архитектуры) задается неявно с помощью установления канальных связей. Вместе с тем выстраивание такой конфигурации позволяет адекватно представить различные модели, нужные для вычислительных алгоритмов [4] – конвейер, матрицу, дерево etc. Хотя используя Паскаль более естественно иметь в виду мультипроцессорные архитектуры с общей памятью, специальная дисциплина в программировании может позволить и написание алгоритмов для архитектур с распределенной памятью.

Система программирования содержит конвертор с СуперПаскаля в Турбо Паскаль и отладчик (визуализатор) параллельного исполнения. Все это позволяет вести отладку и отработку параллельных алгоритмов на обычных персональных компьютерах. Конвертор связывает разработчика с уже существующим и традиционным окружением программирования и обеспечивает имитацию параллельного исполнения псевдопараллелизмом. Этот псевдопараллелизм при исполнении реализуется специальным модулем, который по существу является специальной, ориентированной на СуперПаскаль MPI-библиотекой.

В процессе исполнения вся история параллельного (именно параллельного) исполнения может быть запротоколирована, и отладчик на основании полученного после исполнения протокола визуализирует эту историю – иерархию процессов, динамику передачи сообщений, которая может быть просмотрена либо пошагово, либо выборочно по последовательности параллельных событий.

СуперПаскаль за счет ряда ограничений Паскаля, иногда достаточно мелких, обеспечивает детерминированность параллелизма. Начата разработка еще одной компоненты системы – анализатора конфликтов параллелизма, который предназначен, с одной стороны, для снятия ряда ограничений, а с другой – для более полного статического контроля нарушений детерминизма.

Существенным для отработки параллельных алгоритмов было бы получение ряда статических оценок параллельного исполнения, однако это предполагается осуществить уже при разработке «родного» для СуперПаскаля окружения программирования.

Литература

1. С.И.Катков, И.В.Поттосин. Система параллельного программирования, основанная на языке СуперПаскаль.//Труды Международного семинара «Распределенная обработка информации», Изд. СО РАН, 1998, -с.127-131.
2. P.Brinch Hansen/ SuperPascal – a publication language for parallel scientific computing//Cocurrence Practice and Experience, 1994, v.6(5), p.461-483.
3. B.P.Lester. The Art of Parallel Programming/Prentice Hall, 1993.
4. P.Brinch Hansen. Model programs for computational Science: A programming methodology for multicomputers.//Concurency Practice and Experience, 1993, v.5(5), p.407-423.