

Е. А. Барковский, А. А. Лазутина, А. В. Соколов
(Москва, МГУ; Петрозаводск, ИПМИ КарНЦ РАН). **Модель управления двумя work-stealing деками, двигающимися друг за другом в общей памяти.**

На сегодняшний день «work-stealing» является одной из самых распространенных стратегий динамической балансировки многопроцессорных вычислений [1, 2].

В этом методе каждый процессор решает ряд задач, указатели на которые хранятся в его деке. Когда процессор создает новую задачу, он добавляет указатель на нее в свой дек; когда процессору нужна задача, он берет указатель из вершины дека. Если процессор узнает, что его дек пуст, он перехватывает указатели на задачи у другого процессора. Таким образом, мы работаем с деком с ограниченным входом [3].

В этом докладе мы предлагаем новый метод представления work-stealing деков: все дека расположены в одном, заранее не разделенном, участке общей памяти, где они двигаются друг за другом, по кругу (заявка на патент поддержана [4]).

Допустим, в общей памяти размера m мы работаем с двумя деками, элементы которых равны одной структурной единице. Размер первого дека — x , размер второго — y , z — расстояние между началом первого и концом второго дека.

Весь процесс работы системы разделен на операции, выполняемые в дискретном времени с заданными вероятностями:

Включение элемента в первый дек (p_1); Включение элемента во второй дек (p_2); Включение элементов в два дека одновременно (p_{12}); Исключение элемента из первого дека (q_1); Исключение элемента из второго дека (q_2); Исключение элементов из деков одновременно (q_{12}); Включение в первый дек и одновременное исключение из второго (pq_{12}); Включение во второй дек и одновременное исключение из первого (pq_{21}); дека не меняют своего размера с вероятностью r (отсутствие операций).

Дек, ставший пустым, начинает перехватывать элементы (work-stealing) у другого дека и возобновляет свою работу из середины свободной памяти. Таким образом, исключение из пустого дека не является ошибкой. Была рассмотрена стратегия work-stealing – перехват одного элемента [1].

Были построены математическая и имитационная модели этого процесса.

Работа выполнена при финансовой поддержке РФФИ, грант 18-01-00125-а.

СПИСОК ЛИТЕРАТУРЫ

1. *Blumof R. D., Leiserson C. E.* Scheduling Multithreaded Computations by Work Stealing. — J. ACM, 1999, No 46, p. 720–748.
2. *Herlihy M., Shavit N.* The Art of Multiprocessor Programming. Elsevier, 2008.
3. *Knuth D.* The Art of Computer Programming, Vol. 1. Addison-Wesley, 2001.
4. *Барковский Е. А., Соколов А. В.* Способ управления памятью компьютерной системы, 2016, No. 2016143800.