

УДК 519.687+004.75

## ОЦЕНКА ЭФФЕКТИВНОСТИ РАСПРЕДЕЛЕННЫХ СИСТЕМ ПРИ РЕШЕНИИ ЗАДАЧ ПЕРЕМЕННОГО РАЗМЕРА

А.С. Хританков

Исследуется проблема описания производительности распределенных систем вычислительных кластеров при решении задач переменной трудоемкости. Для этого используется модель производительности интервальных систем с расписанием, которая в данной работе расширена для учета вероятностного характера трудоемкости задачи. Полученные результаты позволяют применить модель систем с расписанием для расчета верхней границы эффективности и погрешности в ее измерении при решении задач глобальной оптимизации стохастическими алгоритмами.

**Ключевые слова:** распределенные вычисления, оценка эффективности, глобальная оптимизация.

### Введение

Рассмотрим процесс решения задачи глобальной оптимизации эвристическим методом, основанном на подходе Монте-Карло. Важной особенностью является то, что исходная задача  $A$  разбивается на  $M$  однотипных подзадач  $\{A_k\}_1^M$ . Например, в методе Basin-Hopping [1] подзадачей является нахождение локального минимума итеративным алгоритмом из начальной точки, выбираемой случайным образом в окрестности предполагаемого минимума. Трудоемкость решения подзадачи напрямую зависит от количества шагов локального алгоритма и сложности вычисления оптимизируемой функции. В широком диапазоне значений параметров алгоритма трудоемкости решения  $\{L_k\}_1^M$  подзадач можно полагать независимыми в совокупности случайными величинами, имеющими одинаковое распределение. Пренебрегая затратами на генерацию начальных точек и относя накладные расходы на распараллеливание и коммуникации к решению подзадач, можно положить, что суммарная трудоемкость решения исходной задачи  $A$  выражается суммой  $L = \sum_{k=1}^M L_k$ . Согласно центральной предельной теореме, распределение случайной величины  $L$  при большом количестве подзадач близко к нормальному  $N(M\mu, M\sigma^2)$ , где  $\mu = EL_i$ ,  $\sigma^2 = DL_i$  в силу единого распределения величин  $L_i$ . Трудоемкость решения подзадачи для алгоритмов локальной оптимизации в задачах конформации молекул и докинга протеин-лиганд, согласно экспериментальным данным, имеет положительно асимметричное унимодальное распределение, при этом  $\sigma \approx 0,4\mu$ . Таким образом, отклонение трудоемкости решения задачи  $A$  из  $M = 400$  подзадач от среднего значения с вероятностью порядка 95% не превышает  $0,04M\mu$ . Вследствие того, что точные значения  $\mu$  и  $\sigma$  неизвестны и должны быть получены в виде оценок, возникает значительная ошибка (около 20%) в определении трудоемкости  $L$ , которую необходимо учитывать при сопоставлении экспериментальных данных с моделью работы системы. Кроме того, существенным для модели производительности системы является учет случайного характера изменения трудоемкости подзадач, который приводит к снижению эффективности работы.

В данной работе мы построим модель производительности распределенной системы при решении описанного класса декомпозируемых задач оптимизации  $A_M$ , оценим структурную эффективность при решении в рамках интервальной модели, рассмотрим экспериментальный метод оценки трудоемкости решения задачи и получим оценки его точности; сравним построенную модель с работой реального вычислительного комплекса с учетом полученных оценок погрешностей.

### Расширение модели систем с расписанием

Для оценки эффективности решения задачи на распределенной вычислительной системе мы будем использовать модель систем с расписанием [2, 3], которую в данном разделе мы кратко изложим. Распределенная система  $R$  представляется как совокупность  $n$  вычислителей, выделяемых для решения задачи  $A$  согласно расписанию  $\bar{h}(t)$ . В зависимости от цели исследования выбирается модель решения задачи на данной системе. Модель включает некоторый набор предположений относительно структуры задачи, механизма распараллеливания и позволяет рассчитать время решения задачи. Модель такого рода называется эталонной системой  $P$  и, по сути, является модельной системой с расписанием, работающей согласно сделанным предположениям по тому же расписанию, что и реальная система. Каждому вычислителю приписывается стоимость использования  $c_i$  в единицу времени, а также, в условиях модели  $P$ , эталонное время решения  $T_i$  – время, которое данный вычислитель затратил бы на решение задачи  $A$ . Эталонное время  $T$ , т.е. время решения задачи системой  $P$ , рассчитывается на основе параметров и предположений модели. Эффективность  $E_t$  по времени решения задачи системой  $R$  по отношению к системе  $P$  определяется как отношение эталонного времени  $T$  системы  $P$  к реальному времени решения  $T$ :

$$E_t = \frac{T}{T}.$$

Ресурсная эффективность  $E_\Phi$  определяется похожим образом – как отношение стоимости использования системы  $\Phi(t)$  за эталонное время  $T$  к аналогичной стоимости за реальное время:

$$E_\Phi = \frac{\Phi(T)}{\Phi(T)}.$$

В качестве примера рассмотрим универсальную эталонную систему  $P$ , в которой отсутствуют накладные расходы на распараллеливание решения и задача полагается произвольно делимой. Эталонное время решения  $T$  в системе  $P$  вычисляется как решение задачи минимизации:

$$\min_t t \text{ при условии } \int_0^t \sum_{i=1}^n \pi_i h_i(\tau) d\tau = L, \quad \pi_i = \frac{L}{T_i}.$$

Интервальная эталонная система [3], как и универсальная модель, предполагает, что трудоемкость  $L$  остается постоянной при повторном решении задачи и при переходе от последовательного к параллельному решению. При рассмотрении класса задач  $A_M$  данное предположение является слишком строгим. Расширим интервальную систему для решения класса задач с переменной трудоемкостью. Обозначим такую систему  $P_m$ , в дополнение к ограничениям интервальной системы будем полагать, что  $\{L_k\}_1^M$  – независимые в совокупности случайные величины. В дальнейшем будем считать, что их распределение имеет положительный коэффициент асимметрии и конечный коэффициент эксцесса. Напомним, что интервальная система предполагает решение задачи как набора одинаковых по трудоемкости подзадач на системе из кластеров одинаковых вычислителей, управляемых совместно. Подзадачи назначаются на кластер пакетами, промежуток времени решения пакета задач называется этапом решения.

Для эталонной системы  $P_m$  можно оценить распределение трудоемкости решения задачи. Действительно, так как  $L = \sum_{k=1}^M L_k$ , то при больших  $M$ , согласно центральной

предельной теореме, распределение  $L$  можно полагать приближенно нормальным  $N(M\mu, M\sigma^2)$ , где  $\mu$  и  $\sigma^2$  – математическое ожидание и дисперсия  $L_k$ . Итак, мы построили систему  $P_m$ , которая отражает основное свойство задач класса  $A_M$  – декомпозируемость на множество подзадач, трудоемкости которых являются одинаково распределенными случайными величинами, и учитывает существенные ограничения на управляемость отдельными вычислителями, характерные для современных распределенных систем.

### Оценка структурной неэффективности в интервальной модели

Воспользуемся расширением интервальной модели и оценим величину структурной неэффективности, возникающей вследствие различной трудоемкости подзадач. Рассмотрим решение пакета из  $M = m \cdot p$  подзадач класса  $A_M$  на кластере из  $p$  вычислителей. Вследствие того, что подзадачи имеют разную трудоемкость, вычислители, завершившие последовательный расчет  $m$  подзадач, будут простаивать, так как следующий пакет подзадач не может быть получен, пока не будет решена последняя подзадача из данного пакета.

Пусть  $X_i$  – случайная величина, выражающая суммарную стоимость решения  $m$  подзадач на вычислителе  $i$ , со средним значением  $\mu_m$  и дисперсией  $\sigma_m^2$ . В среднем долю  $\varepsilon_p$  стоимости вычислений, затраченных на решение подзадач, можно оценить как

$$E\varepsilon_p = E \frac{\sum_{i=1}^p c_i X_i}{\max_{i=1..p} X_i \sum_{i=1}^p c_i}, \Delta_j = \max_{i=1..p} X_i - X_j, X_{i^*} = \max_{i=1..p} X_i,$$

$$E\varepsilon_p = \sum_{i=1}^p \frac{c_i}{c_\Sigma} E \left( \frac{X_i}{X_{i^*}} \right) = \sum_{i \neq i^*} \frac{c_i}{c_\Sigma} \frac{EX_i}{EX_{i^*}} + \frac{c_{i^*}}{c_\Sigma} E \frac{X_{i^*}}{X_{i^*}}.$$

На практике  $E\Delta_j$  лежит в пределах от  $\sigma_m$  до  $4\sigma_m$  при  $p$  от 10 до 50, возрастая как логарифм от числа вычислителей  $p$ ,  $E\Delta_j = a \ln p \cdot \sigma_m$ . Дисперсию  $\sigma_m^2$  можно оценить как дисперсию суммы  $m$  случайных величин,  $\sigma_m^2 \approx m \cdot \sigma^2$ . Для вычислителей из одного кластера логично положить  $c_i = c_0 = c_\Sigma / p$ , получим:

$$E\varepsilon_p = \frac{p-1}{p} \frac{EX_i}{EX_{i^*}} + \frac{1}{p} = \frac{EX_i}{EX_{i^*}} + \frac{1}{p} \frac{E\Delta_i}{EX_{i^*}}, \quad (1)$$

$$E\varepsilon_p \approx \left( 1 + \frac{p-1}{\sqrt{p}} \ln p \frac{C}{\sqrt{M}} \right)^{-1}, \sigma = \theta\mu, EX_{i^*} = EX_i + E\Delta_i, C = a\theta. \quad (2)$$

По построению, величина  $\varepsilon_p$  является оценкой сверху ресурсной эффективности эталонной системы  $P_m$  по отношению к универсальной системе  $P$ , усредненной по различным возможным трудоемкостям подзадач.

Из выражения (2) удастся получить функцию изоэффективности  $M = F(p, \varepsilon_0)$ , указывающую, какого размера (в нашем случае – число подзадач в пакете) должна быть задача, чтобы ее эффективность составляла  $\varepsilon_0$  при  $p$  вычислителях в кластере:

$$M \approx C^2 \left( \frac{\varepsilon_0}{1 - \varepsilon_0} \right)^2 p \ln^2 p. \quad (3)$$

Оценку средней продолжительности  $E\tau$  этапа решения кластера для пакета из  $M$  задач получим из выражения для среднего  $EX_{i^*}$  максимальной трудоемкости задачи на кластере:

$$E\tau = \frac{\mu_m}{c_0} (1 + Cm^{-3/2} \ln p). \quad (4)$$

Следовательно, длительность этапа решения не остается постоянной и при постоянной эффективности возрастает как логарифм количества вычислителей в кластере.

Пусть в распределенной системе имеется  $N$  кластеров, долю кластера в полной стоимости решения задачи обозначим через  $\xi_j > 0$ ,  $\sum_{j=1}^N \xi_j = 1$ . Тогда ресурсная эффективность  $E_\varphi$  системы вычисляется как взвешенная сумма эффективностей кластеров  $E_\varphi^j$ :

$$E_\varphi = \sum_{j=1}^N \xi_j E_\varphi^j. \quad (5)$$

Для определения долей  $\xi_j$  для конкретного расписания нужно указать алгоритм балансировки, например, представленный в работе [3] с продолжительностью этапа решения (4).

### Экспериментальная оценка трудоемкости задачи

Для расчета эффективности решения задачи необходимо знать трудоемкость ее решения  $L$ . Метод экспериментальной оценки состоит в последовательном решении ограниченного набора подзадач, измерения трудоемкости их решения и оценки параметров распределения. После этого, используя предположение о нормальности распределения  $L$ , рассчитывается среднее значение трудоемкости и дисперсия. Случайным образом выберем  $K$  подзадач из  $M$  и решим их на вычислителе  $i$  последовательно, измеряя время их решения  $T_i^k$ . Функцию трудоемкости  $L(\cdot)$  определим как  $L_k = L(A_k) = T_i^k$ . В данной работе мы будем использовать оценки  $m_K = \sum_{k=1}^K L_k / K$  для среднего и  $s_K^2 = \sum_{k=1}^K (L_k - m)^2 / (K - 1)$  для дисперсии. Отсюда получим оценку среднего значения  $\mu_\Sigma = Mm_K$  и стандартного отклонения  $\sigma_\Sigma = Ms_K$  для трудоемкости все задачи  $L$ . Рассмотрим центральный доверительный интервал для трудоемкости

$$I_{\alpha,\beta} = [E\mu_\Sigma - \alpha\delta_\Sigma; E\mu_\Sigma + \alpha\delta_\Sigma],$$

где отклонение  $\delta_\Sigma$  складывается из математического ожидания  $E\sigma_\Sigma$  и дисперсий оценок  $\mu_\Sigma$  и  $\sigma_\Sigma$ :

$$\delta_\Sigma = E\sigma_\Sigma + \beta \sqrt{D\sigma_\Sigma + D\mu_\Sigma + 2\text{Cov}(\mu_\Sigma, \sigma_\Sigma)}.$$

Выражения для входящих в формулу величин получены в книге [4]. Обозначая коэффициент эксцесса распределения  $L_k$  через  $\kappa = \mu_4 / \sigma^4 - 3$ , коэффициент корреляции между оценками через  $\rho$  и полагая  $\sigma = \theta\mu$ , получим при  $M \approx K^2$ :

$$\delta_\Sigma = M\mu \cdot \theta \left( \frac{1}{\sqrt{M}} + \frac{\sqrt{\kappa+6}}{2\sqrt{K}} + \frac{\rho}{K} \sqrt{\frac{\kappa+2}{\kappa+6}} + O(1/K^2) \right). \quad (6)$$

В эксперименте наблюдались значения  $\kappa \in [2;4]$ ,  $\theta \in [0,3;0,5]$ , следовательно, по порядку величины, при  $\alpha = 1$  и  $\beta = 1$  (вероятность попадания  $P\{L \in I_{1,1}\} \approx 80\%$ ), и  $K = 25$ , доверительный интервал для  $L$  составляет

$$I_{1,1} = [0,85 \cdot Mm_K; 1,15 \cdot Mm_K]. \quad (7)$$

Основной вклад в точность оценок вносит второе слагаемое в формуле (1), связанное с дисперсией оценки стандартного отклонения. При большом значении коэффициента эксцесса для сохранения точности расчета нужно пропорционально увеличить объем выборки  $K$ .

### Результаты экспериментов

Для комплекса VnB-Grid при выделении 5 модельных кластеров на МВС-100К в силу схемы управления вычислителями  $M = p$ . При использовании алгоритма SMBH [5] с максимальным числом итераций локального алгоритма  $N_{\max} = 8192$  и радиусе окрестности возмущения  $r = 0,9$ , экспериментально установлено  $\theta \approx 0,4$ ; исходя из распределения трудоемкости решения подзадач, численно рассчитан коэффициент  $a = 0,84$ . В данном эксперименте трудоемкость рассчитывалась по  $K = 120$  подзадачам. Экспериментально полученные значения ресурсной эффективности, доверительные интервалы  $I_{1,1}$  (7), и графики функций (1) и (2) приведены на рис. 1.

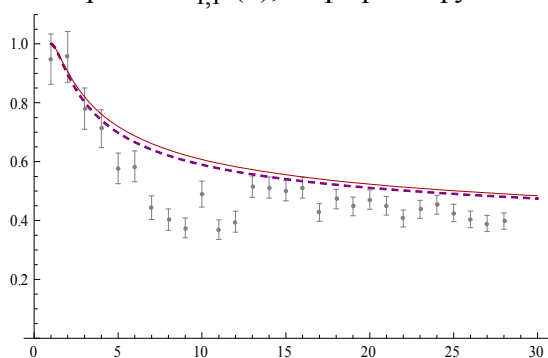


Рис. 1. Сравнение модели структурной неэффективности и экспериментальных данных: приближенная модель (пунктир) и точная модель (сплошная)

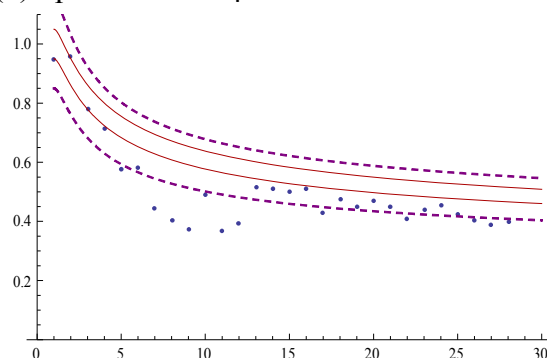


Рис. 2. Возможности обнаружения отклонений результатов экспериментов от модели при оценке по ограниченной выборке (пунктир) и каждой подзадаче (сплошная)

Заметим, что доверительный интервал при  $p \leq 12$  больше, так как задача состояла из 120 подзадач, в то время как при  $12 < p \leq 28$  – из 512 подзадач. Из графика видно, что различия между (1) и (2) имеют порядок точности измерений, поэтому использование приближения (2) в теоретических выкладках вполне оправдано. Полученная модель оценивает эффективность сверху, отклонения экспериментальных данных, по большей части, объясняются простым вычислителей кластера при недостатке задач, что заметно при  $p=7-12$ , и замедлением при использовании нескольких ядер одного процессора. При этом, как видно из рис. 2, определение трудоемкости по выборке размера  $K \approx \sqrt{M}$  позволило бы обнаружить снижение эффективности только при  $p=7-12$ , что вполне достаточно для реальных задач, когда повторное решение задачи не имеет смысла, а измерение трудоемкости подзадач невозможно в процессе решения. Сплошными линиями на рис. 2 показаны границы при  $K \approx M$ , видно, что при исследовании

поведения системы этого достаточно для обнаружения небольших отклонений от модели, ошибка определения эффективности составит при  $M \approx 500$  порядка 5%.

### **Заключение**

Рассмотрена проблема описания производительности распределенных вычислительных систем при решении задач с переменной трудоемкостью, таких как задачи глобальной оптимизации при использовании методов типа Монте-Карло. Предложенное обобщение модели интервальных систем позволяет оценить сверху эффективность решения на системах вычислительных кластеров (5). Полученные оценки погрешности (6) в определении трудоемкости экспериментальным путем по небольшой доле подзадач на имеющихся экспериментальных данных подтверждают адекватность модели.

В дальнейшем планируется продолжить исследования эффективности интервальных систем, уточнить верхнюю оценку, рассмотрев падение эффективности при недостатке подзадач, обобщить результаты по интервальным системам на случай переменной трудоемкости задачи. Кроме того, необходимо рассмотреть влияние структуры расписания на эффективность по времени.

### **Литература**

1. Wales D.J., Doye J.P.K. Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing up to 110 Atoms // *Journal of Physical Chemistry*. – 1997. – № 101. – P. 5111–5116.
2. Посыпкин М.А., Хританков А.С. О понятии ускорения и эффективности в распределенных системах // *Научный сервис в сети Интернет'2008: решение больших задач: Труды конференции*. – 2008. – С. 149–156.
3. Хританков А.С. Об одном алгоритме балансировки вычислительной нагрузки в распределенных системах // *Параллельные вычислительные технологии (ПаВТ'2009): Труды международной научной конференции (Нижний Новгород, 30 марта – 3 апреля 2009 г.)*. – Челябинск: Изд. ЮУрГУ, 2009. – 839 с., С. 778–783.
4. Крамер Г. Математические методы статистики. – 2-е изд. – М.: Мир, 1975. – 648 с.
5. Посыпкин М.А. Методы и распределенная программная инфраструктура для численного решения задачи поиска молекулярных кластеров с минимальной энергией // *Параллельные вычислительные технологии (ПаВТ'2009): Труды международной научной конференции (Нижний Новгород, 30 марта – 3 апреля 2009 г.)*. – Челябинск: Изд. ЮУрГУ, 2009. – 839 с., С. 269–281.

*Хританков Антон Сергеевич* – Московский физико-технический институт (государственный университет), аспирант, anton.khritankov@gmail.com