

## **13 Управление балансировкой загрузки в вычислительных сетях**

*Н.О.Амелина, О.Н.Граничин*

В настоящее время распределенные сетевые системы с параллельными вычислениями используются все чаще. Для таких систем важно эффективное решение задачи о распределении пакетов заданий между несколькими узлами. Подобные задачи возникают не только в вычислительных сетях, но также в производственных сетях [13.1], сетях обслуживания [13.2], транспортных, логистических, сенсорных и других сетях. Задания могут выполняться тем узлом (агентом), кому они поступили, или перераспределяться между узлами (агентами). Задача балансировки загрузки сети состоит в том, чтобы поддерживать равномерную загрузку у всех узлов сети (агентов) с течением времени. По теме балансировки загрузки сетей в последнее время появляется все большее количество исследовательских статей, что свидетельствует о научной актуальности этой проблемы. В основном, эти публикации относятся к области информатики и системного программирования. В них, как правило, не рассматриваются возможные помехи и «разрывы» связей между узлами. В рамках одного компьютера и многих вычислительных ядер это предположение может быть достаточно реалистичными. Но, при рассмотрении сетевых систем роль помех и «разрывов» связей возрастает.

Большинство стратегий динамической балансировки загрузки можно отнести к классу централизованных или к классу полностью распределенных (децентрализованных). При централизованной стратегии имеется специальный ресурс, который собирает информацию о состоянии всей сети и принимает решение о распределении заданий для каждого из узлов. При полностью децентрализованных стратегиях на каждом узле выполняется протокол балансировки загрузки, обменивающийся информацией о состоянии с другими узлами сети, и перемещение заданий происходит только между соседними узлами.

### **13.1 Балансировка загрузки сети**

Пусть  $N = \{1, \dots, n\}$  — набор вычислительных узлов (процессоров), каждый из которых выполняет поступающие задания по принципу очереди.

### 13.1.1 Централизованная система

Сначала рассмотрим задачу балансировки загрузки централизованной сети, когда предполагается, что задания в системе поступают и распределяются через центральный сервер (брокер загрузки), который связан со всеми узлами в сети.

Пусть система из  $n$  вычислительных узлов (вычислительный кластер) работает итеративно, обрабатывая пакеты заданий известного размера  $z_k$  на каждой итерации  $k$ . Предположим, что весь пакет  $z_k$  может быть произвольным образом разделен на  $n$  заданий  $u_k^j$ ,  $j = 1, \dots, n$ ,

$$\sum_{j=1}^n u_k^j = z_k,$$

для всех узлов, и время вычисления для узла  $j$  определяется по формуле

$$t_k^j(u_k^j) = u_k^j / r_k^j,$$

где  $r_k^j \in \mathbb{R}$  — производительность узла  $j$  на  $k$ -ой итерации (предполагаем, что производительности положительные).

Задача *минимизировать* общее время выполнения заданий  $z_k$ :

$$T_k(\mathbf{u}_k) = \|\mathbf{t}_k(\mathbf{u}_k)\|_\infty = \max_{j \in N} t_k^j(u_k^j) \rightarrow \min_{\mathbf{u}_k}, \quad (13.1)$$

где  $\mathbf{u}_k = (u_k^1, u_k^2, \dots, u_k^n)^\top$ ,  $\mathbf{t}_k = (t_k^1(u_k^1), t_k^2(u_k^2), \dots, t_k^n(u_k^n))^\top$ .

Совершенный алгоритм балансировки загрузки тот, который обеспечивает загрузкой все узлы и сводит к минимуму необходимость коммуникации между узлами для обеспечения балансировки и передачи информации между узлами. Балансировка загрузки особенно сложна для решения, если задания поступают динамически с течением времени и непредсказуемо в ходе выполнения алгоритма. Это касается многих рекурсивных алгоритмов.

Когда производительность узлов известна, то лучшей стратегией управления будет пропорциональное распределение заданий:

$$u_k^1 / r_k^1 = u_k^2 / r_k^2 = \dots = u_k^n / r_k^n.$$

Такая стратегия управления называется «балансировкой загрузки». Обозначим  $\mathbf{x}_k = (1/r_k^1, 1/r_k^2, \dots, 1/r_k^n)^\top$  — вектор из величин обратных к производительностям, и  $\mathbf{u}_k = \mathbb{U}(\mathbf{x}_k, z_k)$  — соответствующую стратегию управления.

На практике обратные производительности узлов могут быть неизвестны:  $\mathbf{x}_k = \mathbf{x}^* + \mathbf{w}_k$ , или они могут меняться с течением времени:

$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{w}_k$ , где  $\mathbf{w}_k \in \mathbb{R}^n$  — неизвестные независимые случайные векторы. Кроме того, размеры заданий могут быть зашумлены.

Для построения управляющих воздействий в условиях неопределенностей обычный способ состоит в использовании на каждой итерации  $k$  оценок вектора обратных производительностей  $\hat{\mathbf{r}}_k$  вместо известного точно вектора  $\mathbf{x}_k$ . Оценки естественно определить из условия

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|^2 \rightarrow \min. \quad (13.2)$$

При этом для вычисления  $\mathbf{u}_k$  на итерации  $k$  можно воспользоваться стратегией

$$\mathbf{u}_k = \mathbb{U}(\hat{\mathbf{x}}_k, z_k).$$

Проблема минимизации (13.2) относится к классу задач минимизации функционалов типа среднего риска [13.3], если рассмотреть эмпирический функционал качества

$$F(\hat{\mathbf{x}}_k) = \left( \frac{\langle \hat{\mathbf{x}}_k - \mathbf{x}_k, \mathbf{u}_k \rangle}{z_k} \right)^2 = \left( 1 - \frac{\|\mathbf{t}_k\|_1}{z_k} \right)^2 \rightarrow \min_{\hat{\mathbf{x}}_k}. \quad (13.3)$$

Здесь  $\langle \cdot, \cdot \rangle$  — операция скалярного произведения,  $\|\cdot\|_1$  — сумма модулей компонент (норма  $\ell_1$ ).

Для определения оптимального вектора  $\mathbf{x}^*$  целесообразно использовать рандомизированные алгоритмы стохастической аппроксимации (или SPSA, simultaneous perturbation stochastic approximation) [13.4, 13.5]. Эти же алгоритмы работоспособны и для отслеживания изменений  $\mathbf{x}_k$  [13.6].

**Алгоритмы.** Алгоритм SPSA с одним измерением:

1. *Инициализация и выбор коэффициентов.* Положим  $k = 0$ , выберем некоторое начальное значение оценки  $\hat{\mathbf{x}}_0 \in \mathbb{R}^n$  и достаточно небольшие  $\alpha > 0$  и  $\beta > 0$ .
2. *Итерация  $k \rightarrow k + 1$ .*
  - 2а.  $k := k + 1$ .
  - 2б. Получаем следующее задание  $z_k$ .
  - 2в. Генерируем случайный вектор  $\mathbf{\Delta}_k$  в соответствии с распределением Бернулли с независимыми одинаково распределенными компонентами, которые равны  $\pm 1$  с вероятностью  $\frac{1}{2}$ .
  - 2г. Вычислим следующий вход по правилу

$$\mathbf{u}_k = \mathbb{U}(\hat{\mathbf{x}}_{k-1} + \beta \mathbf{\Delta}_k, z_k).$$

**2д.** Запускаем кластер с входом  $\mathbf{u}_k$  и ждем пока все задания не закончатся.

**2е.** Получаем новое эмпирическое значение  $y_k$  функционала (13.3) по формуле

$$y_k = \left(1 - \frac{\|\mathbf{t}_k\|_1}{z_k}\right)^2. \quad (13.4)$$

**2ж.** Получаем новую оценку

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{n-1} - \frac{\alpha}{\beta} \mathbf{\Delta}_k y_k.$$

3. Повторяем шаг 2.

*Алгоритм SPSA с двумя измерениями* отличается от предыдущего алгоритма только в четырех операциях:

**2г-1.** Вычисляем следующие входы по правилу

$$\mathbf{u}_{2k-1} = \mathbb{U}(\hat{\mathbf{x}}_{k-1}, z_{2k-1}).$$

**2д-1.** Запускаем кластер с входом  $\mathbf{u}_{2k-1}$  и ждем пока все задания не закончатся.

**2е-1.** Получаем новое эмпирическое значение  $y_{2k-1}$  по (13.4).

**2г-2.** Вычисляем следующие входы по правилу

$$\mathbf{u}_{2k} = \mathbb{U}(\hat{\mathbf{x}}_{k-1} + \beta \mathbf{\Delta}_k, z_{2k}).$$

**2д-2.** Запускаем кластер с входом  $\mathbf{u}_{2k}$  и ждем пока все задания не закончатся.

**2е-2.** Получаем новое эмпирическое значение  $y_{2k}$  по (13.4).

**2ж.** Получаем новую оценку

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} - \frac{\alpha}{\beta} \mathbf{\Delta}_k (y_{2k} - y_{2k-1}).$$

### 13.1.2 Децентрализованная система

В случае, когда в сети нет узла, которому доступна информация о всех других узлах, то нет возможности выделить единый центр принятия решений. Задача балансировки загрузки может быть эффективно решена и для распределенной децентрализованной системы.

Рассмотрим модель системы распределения однотипных заданий между  $n$  разными узлами (агентами) для параллельной работы с обратной связью. Пусть задания поступают в систему в различные моменты времени и на разные узлы.

В каждый момент времени  $t$  состояние агента  $i$ ,  $i \in N$ , описывается двумя характеристиками:

- $q_t^i \geq 0$  — длина очереди из атомарных элементарных заданий узла  $i$  в момент времени  $t$ ;
- $r_t^i > 0$  — производительность узла  $i$  в момент времени  $t$ .

Изменения состояний агентов определяются следующими уравнениями:

$$q_{t+1}^i = q_t^i - r_t^i + z_t^i + u_t^i, \quad i \in N, \quad t = 0, 1, 2, \dots, \quad (13.5)$$

где  $z_t^i$  — новое задание, поступившее на узел  $i$  в момент времени  $t$ ,  $u_t^i$  — результат перераспределения заданий между узлами (добавление или уменьшение), получающийся в итоге применения выбранного протокола перераспределения заданий. В уравнениях динамики предполагаем, что  $\sum_i u_t^i = 0$ ,  $t = 0, 1, 2, \dots$

Будем считать, что для формирования стратегии управления каждый узел  $i \in N$  в момент времени  $t$  имеет следующую информацию:

- зашумленные данные о своей длине очереди

$$y_t^{i,i} = q_t^i + w_t^{i,i}, \quad (13.6)$$

- зашумленные наблюдения о длинах очередей соседей, если  $N_t^i \neq \emptyset$

$$y_t^{i,j} = q_{t-d_t^{i,j}}^j + w_t^{i,j}, \quad j \in N_t^i, \quad (13.7)$$

где  $w_t^{i,j}$  — помехи, а  $0 \leq d_t^{i,j} \leq \bar{d}$  — целочисленная задержка,  $\bar{d}$  — максимально возможная задержка,

- данные о своей производительности  $r_t^i$  и о производительностях узлов соседей  $r_t^j$ ,  $j \in N_t^i$ .

Будем рассматривать две постановки задачи: стационарную и нестационарную.

Стационарный случай. Все задания поступают в систему на разные узлы в начальный момент времени и производительности узлов не меняются с течением времени.

Если все задания выполняются только тем агентом, которому они поступили, то время реализации всех заданий определяется как

$$T_{\max} = \max_{i \in N} q_0^i / r_0^i.$$

Нестационарный случай. Новые задания могут поступать в систему на любой из  $n$  узлов в различные моменты времени  $t$  и производительности узлов могут меняться с течением времени.

Для момента времени  $t$  определим  $T_t$  — время до окончания выполнения всех заданий на всех узлах.

Если в стационарном случае начиная с момента времени  $t$  задания не перераспределяются между узлами, то время реализации всех заданий определяется как

$$T_t = \max_{i \in N} \frac{q_t^i}{r_t^i}. \quad (13.8)$$

Будем называть отношение  $\frac{q_t^i}{r_t^i}$  загруженностью узла  $i$  в момент времени  $t$ .

Поставим цель управления

$$T_t \rightarrow \min_{\bar{u}_t}. \quad (13.9)$$

Для достижения поставленной цели естественно использовать протокол перераспределения заданий с течением времени. Это позволит увеличить пропускную способность системы и уменьшить время выполнения заказов в системе.

**Лемма 13.1 (Об оптимальной стратегии управления.)** *В стационарном случае из всех возможных вариантов распределения общего количества заданий, не обработанных к моменту времени  $t$ , наименьшее время работы системы соответствует тому, при котором*

$$q_t^i / r_t^i = q_t^j / r_t^j, \quad \forall i, j \in N. \quad (13.10)$$

**Доказательство.** Возьмем в качестве состояния узла  $i$ :  $x_t^i = q_t^i / r_t^i$ .

Проведем доказательство от противного. Предположим, что при некоторой оптимальной стратегии не все  $x_t^i$  равны между собой, т. е.

существует такой узел с номером  $k \in N$  и такое подмножество узлов  $\tilde{N}_t$ , что  $x_t^k > x_t^j, \forall j \in \tilde{N}_t$ .

Обозначим через  $l = |\tilde{N}_t|$  количество узлов в  $\tilde{N}_t$ . Состояние остальных  $n - l$  узлов равны  $x_t^k$ .

Пусть разница между состоянием  $k$ -го узла и наибольшим из множества  $\tilde{N}_t$  равна  $\epsilon_t$ , т. е.

$$\epsilon_t = x_t^k - \max_{j \in \tilde{N}_t} x_t^j. \quad (13.11)$$

Рассмотрим новую стратегию распределения заданий. Уменьшим загрузку всех  $n - l$  узлов, имевших максимальную загрузку, на  $\frac{\epsilon}{2(n-l)}$  (т. е. всего на  $\frac{\epsilon}{2}$ ) и добавим эти  $\frac{\epsilon}{2}$  заданий к любому из  $l$  узлов из  $\tilde{N}_t$ . Для новой стратегии получили, что время обработки заданий в системе будет меньше исходного на  $\frac{\epsilon}{2(n-l)}$ , т. е. меньше, чем минимальное по предположению. Получили противоречие.  $\square$

**Следствие 13.1** *Если взять  $x_t^i = q_t^i/r_t^i + t$  в качестве состояния узла  $i$  динамической сети, то цель управления — достижение консенсуса в сети — будет соответствовать оптимальному распределению заданий между узлами в стационарном случае.*

Таким образом, достаточно рассматривать задачу о поддержке равномерной загрузки всех узлов сети.

В работах [13.7, 13.8] для задачи балансировки загрузки сети были исследованы свойства алгоритма управления, называемого «протоколом локального голосования», в котором значение управления для каждого узла определялось взвешенной суммой разностей между информацией о его состоянии и информацией о состоянии его соседей

$$\bar{u}_t^i = \gamma \sum_{j \in \bar{N}_t^i} b_t^{i,j} (y_t^{i,j} - y_t^{i,i}), \quad (13.12)$$

где  $\gamma > 0$  — параметр шага протокола управления,  $\bar{N}_t^i \subset N_t^i$ ,  $b_t^{i,j} > 0 \forall j \in \bar{N}_t^i$ . Положив  $b_t^{i,j} = 0$  для всех остальных пар  $(i, j)$ , определим матрицы  $B_t = [b_t^{i,j}]$ .

Отметим, что этот протокол отличается от часто встречающихся других, когда параметр шага протокола управления  $\gamma$  выбирается разным для разных  $i$  (например,  $\gamma^i = 1/d^i(B_t)$ ).

В этом протоколе управления для всех  $i \in N$ , для всех  $t$  определим  $\bar{N}_t^i = N_t^i$  и  $b_t^{i,j} = r_t^j/r_t^i, j \in N_t^i$ .

В действительности, применение протокола (13.12) на практике оправдано при выполнении дополнительных предположений. Прежде

всего предполагается мгновенный обмен соответствующими потоками входных данных. Кроме того, надо дополнительно проверять согласованность пересылок пакетов, так как возможны различные коллизии за счет задержек и получения информации с помехами. В частности, использование протокола (13.12) на практике в задаче балансировки загрузки узлов децентрализованной вычислительной сети предполагает дополнительные согласования размеров пересылаемых пакетов между агентами. При перераспределении ресурсов (либо заданий между узлами) не допускается ни «перерасход», ни «недорасход» ресурсов. Для выполнения этого условия в каждом узле происходят дополнительные проверки и согласования с соседями. По протоколу локального голосования каждый узел определяет, сколько он «отдаст» или «примет» заданий. Далее те узлы, которые готовы принять задания, посылают запросы соседям о том, сколько они фактически готовы передать в данный момент. В ответ на эти запросы каждый узел, «принимающий» задания, дает подтверждение, сколько он примет от того или иного узла заданий, и согласовывает это количество с ним (при этом каждый узел ориентируется на свои текущие значения  $\tilde{u}_t^i$ , которые рекомендуются протоколом локального голосования). Предполагается, что процедура согласования и передачи заданий по времени занимает существенно меньше времени, чем длительность одного такта работы динамической сети. Например, в задаче распределения заказов при организации грузоперевозок согласование заданий на следующий день происходит вечером после окончания текущего рабочего дня (см. [13.9]).

Если, начиная с момента времени  $\bar{t}$  в систему перестают поступать новые задания ( $z_t^i = 0, t \geq \bar{t}$ ) и производительности не меняются ( $r_t^i = r^i, t \geq \bar{t}$ ), то теорема 1 из [13.10], сформулированная также в разделе «Приближенный консенсус в стохастических динамических сетях с неполной информацией и задержками в измерениях», позволяет свести исследование динамики балансировки загрузки узлов к исследованию соответствующей дискретной усредненной модели, которое может выполняться аналитически, либо численно.

При реализации протокола (13.12), конечно, осуществляется округление. Применение алгоритма эффективно, когда очереди из заданий достаточно длинные. В этом случае, так как изучается приближенный консенсус, такое округление существенно не повлияет на результат.

В качестве примера рассмотрим систему, состоящую из шести вычислительных блоков (рис. 13.3.1). Топология сети в каждый момент времени случайна. Связь 2—1 присутствует с вероятностью 1/2, так-



же как связь 3—1 (рис. 13.3.1-б). Вероятности появления дуги 2—1 и 3—1 одинаковые:  $p_a^{13} = p_a^{12} = 1/2$ .

В случае с равномерно распределенными задержками в измерениях, когда целочисленная задержка  $d_t^{ij}$  равна 0 или 1 с вероятностью  $1/2$ ,  $\bar{d} = 1$ ,  $p_0^{ij} = p_1^{ij} = 1/2$ , расширим пространство состояний:

$$\bar{X}_t = [x_t^1, \dots, x_t^n, x_{t-1}^1, \dots, x_{t-1}^n] \in \mathcal{R}^{2n}. \quad (13.13)$$

Матрица  $G$  соответствующей усредненной дискретной модели выглядит следующим образом:

$$G = \begin{pmatrix} \frac{1}{2}H\alpha & \frac{1}{2}H\alpha \\ 0 & 0 \end{pmatrix}, \quad (13.14)$$

где

$$H = \begin{pmatrix} 0 & \frac{1}{2}b^{1,2} & \frac{1}{2}b^{1,3} & 0 & 0 & 0 \\ 0 & 0 & 0 & b^{2,4} & 0 & 0 \\ 0 & 0 & 0 & 0 & b^{3,5} & 0 \\ 0 & 0 & 0 & 0 & b^{4,5} & 0 \\ 0 & 0 & 0 & 0 & 0 & b^{5,6} \\ b^{6,1} & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (13.15)$$

### 13.2 Балансировка загрузки при нестационарной постановке задачи

Большинство современных систем характеризуются нестационарностью, т. к. в условия могут меняться с течением времени (например, в систему могут поступать новые задания). Поэтому важно рассмотреть задачу балансировки загрузки сети при нестационарной постановке задачи [13.14, 13.15]. В этом разделе будет исследована применимость протокола локального голосования для децентрализованной балансировки загрузки сети с переменной топологией и помехами в измерениях при нестационарной постановке задачи.

Предположим, что выполнено следующие условие

$$\mathbf{A1} : \quad p_t^i \geq p_{\min} > 0, \quad \forall i \in N, t = 0, 1, \dots$$

Если взять  $x_t^i = q_t^i/p_t^i$  в качестве состояния узла  $i$  динамической сети в момент времени  $t = 0, 1, \dots, T$ , то цель управления — достижение консенсуса в сети — будет соответствовать оптимальному распределению заданий между узлами, как показано в предыдущем разделе, а также в [13.10].

При введенных обозначениях уравнения динамики каждого агента можно переписать в следующем виде:

$$x_{t+1}^i = x_t^i + \bar{u}_t^i + f_t^i, \quad (13.16)$$

где  $\bar{u}_t^i = \bar{u}_t^i/p_t^i$ ,  $i \in N$  — «нормализованные» управления,  $f_t^i = -1 + z_t^i/p_t^i$  — возмущения.

Для поддержки равномерной загрузки всех узлов сети (с целью увеличения общей пропускной способности системы и сокращения времени выполнения заказов) естественно использовать протокол перераспределения заданий с течением времени.

Будем считать, что для формирования управления  $\bar{u}_t^i$  каждый узел  $i \in N$  имеет зашумленные данные о своем собственном состоянии

$$y_t^{i,i} = x_t^i + w_t^{i,i} \quad (13.17)$$

и, если множество  $N_t^i$  непусто, зашумленные наблюдения о состояниях соседей

$$y_t^{i,j} = x_t^j + w_t^{i,j}, \quad j \in N_t^i, \quad (13.18)$$

где  $w_t^{i,i}, w_t^{i,j}$  — помехи (шум).

Приближенно, с учетом возможных дополнительных согласований по протоколу передачи пакетов на основе протокола (13.12) динамика системы в рассматриваемом случае имеет вид

$$x_{t+1}^i = x_t^i + \gamma \sum_{j \in \bar{N}_t^i} b_t^{i,j} (y_t^{i,j} - y_t^{i,i}) + f_t^i, \quad i \in N, \quad (13.19)$$

а в векторно-матричном виде:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \gamma \mathbb{L}(B_t) \mathbf{x}_t + \gamma \mathbf{w}_t + \mathbf{f}_t, \quad (13.20)$$

где используются обозначения  $\mathbf{x}_t$ ,  $\mathbf{w}_t$  и  $\mathbf{f}_t$  для векторов составленных из соответствующих элементов  $x_t^1, \dots, x_t^n$ ,  $\sum_{j \in \bar{N}_t^1} b_t^{1,j} (w_t^{1,1} - w_t^{1,j}), \dots, \sum_{j \in \bar{N}_t^n} b_t^{n,j} (w_t^{n,n} - w_t^{n,j})$ , и  $f_t^1, \dots, f_t^n$ .

Пусть  $(\Omega, \mathbb{F}, \mathbb{P})$  — основное вероятностное пространство, и выполнены следующие условия:

**A2: а)** Для всех  $i \in N$ ,  $j \in N_t^i \cup \{i\}$  помехи наблюдений  $w_t^{i,j}$  — центрированные, независимые одинаково распределенные случайные величины с ограниченными дисперсиями:  $E(w_t^{i,j})^2 \leq \sigma_w^2$ .

Здесь и далее  $E$  — символ математического ожидания.

**б)** Для всех  $i \in N$ ,  $j \in N_{\max}^i$  появление «переменных» дуг  $(j, i)$  в графе  $\mathbb{G}_{A_t}$  — независимые случайные события (т. е. матрицы

$A_t$  — независимые одинаково распределенные случайные матрицы). Для всех  $i \in N$ ,  $j \in \bar{N}_t^i$  веса  $b_t^{i,j}$  в протоколе управления независимые случайные величины с математическими ожиданиями  $E b_t^{i,j} = b^{i,j}$  и ограниченными дисперсиями  $E(b_t^{i,j} - b^{i,j})^2 \leq \sigma_b^2$ .

**в)** Для всех  $i \in N$ ,  $t = 0, 1, \dots$  величины  $f_t^i$  в (13.16) случайные, независимые и одинаково распределенные с математическим ожиданием  $E f_t^i = \bar{f}$  и дисперсией  $E(f_t^i - \bar{f})^2 = \sigma_f^2$ .

Кроме того, все эти случайные величины не зависят между собой.

**A3:** Граф  $\mathbb{G}_{A_{\max}}$ , определяемый матрицей смежности  $A_{\max}$  с элементами

$$a_{\max}^{i,j} = b^{i,j}, \quad i, j \in N,$$

имеет остовное дерево и  $a_{\max}^{i,j} > 0$  для любой дуги  $(j, i) \in E_{\max}$ .

**A4:** Параметр шага протокола управления  $\gamma > 0$  удовлетворяет условиям

$$\gamma \leq \frac{1}{d_{\max}(A_{\max})}, \quad (13.21)$$

и

$$\lambda_{\max}(Q)\gamma \leq \operatorname{Re}(\lambda_2(A_{\max})), \quad (13.22)$$

где  $\operatorname{Re}(\lambda_2(A_{\max}))$  — вещественная часть второго по абсолютной величине собственного числа матрицы  $A_{\max}$ , а  $\lambda_{\max}(Q)$  — максимальное собственное число матрицы

$$Q = E(\mathbb{L}(A_{\max}) - \mathbb{L}(B_t))^T (\mathbb{L}(A_{\max}) - \mathbb{L}(B_t)).$$

Заметим, что в силу определения и свойств **A26**  $\mathbb{L}(A_{\max}) = E\mathbb{L}(B_t)$ . Кроме того,  $0 < \operatorname{Re}(\lambda_2(A_{\max})) < 1$  при выполнении условия **A3** ([13.12]).

*Определение:*  $n$  узлов достигают асимптотического среднеквадратичного  $\varepsilon$ -консенсуса, если  $E\|x_0^i\|^2 < \infty$ ,  $i \in N$ , и существует такая последовательность  $\{x_t^*\}$ , что  $\lim_{t \rightarrow \infty} E\|x_t^i - x_t^*\|^2 \leq \varepsilon$  для всех  $i \in N$ . Здесь и далее  $\|\cdot\|$  — евклидова норма вектора.

Пусть  $x_0^*$  — среднее значение начальных данных

$$x_0^* = \frac{1}{n} \sum_{i=1}^n x_0^i$$

и  $\{x_t^*\}$  — траектория усредненной системы

$$x_{t+1}^* = x_t^* + \bar{f}, \quad (13.23)$$

где  $\bar{f}$  — среднее значение из условия **A2в**.

*Теорема 1:* Если выполнены условия **A1–A4** тогда для траекторий систем (13.20) и (13.23) выполняется оценка

$$\mathbb{E}\|\mathbf{x}_{t+1} - x_{t+1}^* \mathbf{1}\|^2 \leq \frac{\Delta}{\rho} + (1 - \rho)^t \left( \|\mathbf{x}_0 - x_0^* \mathbf{1}\|^2 - \frac{\Delta}{\rho} \right), \quad (13.24)$$

где  $\mathbf{1}$  — вектор из единиц,

$$\begin{aligned} \rho &= \gamma \operatorname{Re}(\lambda_2(A_{\max})) - \gamma^2 \lambda_{\max}(Q), \\ \Delta &= 2\sigma_w^2 \gamma^2 (n^2 \sigma_b^2 + \sum_{i=1}^n \sum_{j=1}^n (b^{i,j})^2) + n\sigma_f^2, \end{aligned}$$

т. е. при  $\mathbb{E}\|x_0^i\|^2 < \infty$ ,  $i \in N$ , в системе (13.20) достигается асимптотического среднеквадратичного  $\varepsilon$ -консенсуса при

$$\varepsilon = \frac{\Delta}{\rho}.$$

**Доказательство.** Для разницы траекторий систем (13.20) и

$$x_{t+1}^* \mathbf{1} = x_t^* \mathbf{1} + \bar{f} \mathbf{1} \quad (13.25)$$

в силу определений имеем

$$\begin{aligned} D_{t+1} &= \mathbf{x}_{t+1} - x_{t+1}^* \mathbf{1} = \mathbf{x}_t - \gamma(\mathbb{L}(B_t)\mathbf{x}_t + \mathbf{w}_t) + \mathbf{f}_t - x_t^* \mathbf{1} - \bar{f} \mathbf{1} = \\ &= D_t - \gamma(\mathbb{L}(B_t)D_t + \mathbf{w}_t) + \mathbf{f}_t - \bar{f} \mathbf{1}, \end{aligned}$$

т. к. вектор  $\mathbf{1}$  является собственным вектором матрицы-Лапласиана  $\mathbb{L}(B_t)$ , соответствующим нулевому собственному значению:  $\mathbb{L}(B_t)\mathbf{1} = 0$ . Далее, добавив и вычтя  $\gamma\mathbb{L}(A_{\max})D_t$ , получаем

$$D_{t+1} = (I - \gamma\mathbb{L}(A_{\max}))D_t + \gamma(\mathbb{L}(A_{\max})D_t - \mathbb{L}(B_t))D_t + \gamma\mathbf{w}_t + \mathbf{f}_t - \bar{f} \mathbf{1},$$

где  $I$  единичная матрица соответствующей размерности.

Обозначим  $\mathbb{F}_t$  —  $\sigma$ -алгебру вероятностных событий, порожденную случайными элементами  $x_0^i, w_0^{i,j}, w_1^{i,j}, \dots, w_{t-1}^{i,j}, f_0^i, f_1^i, \dots, f_{t-1}^i, b_0^{i,j}, b_1^{i,j}, \dots, b_t^{i,j}$ ,  $i, j \in N, A_0, \dots, A_t$  и рассмотрим условное математическое ожидание от квадрата нормы  $D_{t+1}$

$$\begin{aligned} \mathbb{E}_{\mathbb{F}_t} \|D_{t+1}\|^2 &= \|(I - \gamma\mathbb{L}(A_{\max}))D_t\|^2 + \\ &+ 2D_t^T (I - \gamma\mathbb{L}(A_{\max}))^T (\gamma(\mathbb{L}(A_{\max})D_t - \mathbb{L}(B_t))D_t + \end{aligned}$$

$$\begin{aligned}
& +2D_t^T(I - \gamma\mathbb{L}(B_t))^T (\gamma\mathbb{E}_{\mathbb{F}_t} \mathbf{w}_t + \mathbb{E}_{\mathbb{F}_t}(\mathbf{f}_t - \bar{f}\mathbf{1})) + 2\gamma\mathbb{E}_{\mathbb{F}_t} \mathbf{w}_t^T(\mathbf{f}_t - \bar{f}\mathbf{1}) + \\
& \quad (13.26) \\
& +\gamma^2 D_t^T(\mathbb{L}(A_{\max}) - \mathbb{L}(B_t))^T(\mathbb{L}(A_{\max}) - \mathbb{L}(B_t)) + \gamma^2 \mathbb{E}_{\mathbb{F}_t} \|\mathbf{w}_t\|^2 + \\
& \quad + \mathbb{E}_{\mathbb{F}_t} \|\mathbf{f}_t - \bar{f}\mathbf{1}\|^2.
\end{aligned}$$

В силу выполнения условия **A2в** и независимости  $\mathbf{f}_t$  от  $\sigma$ -алгебры  $\mathbb{F}_t$  имеем

$$\mathbb{E}_{\mathbb{F}_t}(\mathbf{f}_t - \bar{f}\mathbf{1}) = \mathbb{E}(\mathbf{f}_t - \bar{f}\mathbf{1}) = 0, \quad \mathbb{E}_{\mathbb{F}_t} \|\mathbf{f}_t - \bar{f}\mathbf{1}\|^2 = \mathbb{E} \|\mathbf{f}_t - \bar{f}\mathbf{1}\|^2 = n\sigma_f^2. \quad (13.27)$$

В силу выполнения условия **A2а** и независимости  $w_t^{i,j}$ ,  $i, j \in N$  от  $\mathbf{f}_t$ ,  $\sigma$ -алгебры  $\mathbb{F}_t$  и между собой получаем

$$\mathbb{E}_{\mathbb{F}_t} \sum_{j \in N_t^1} b_t^{i,j} (w_t^{i,i} - w_t^{i,j}) = \sum_{j \in N_t^1} b_t^{i,j} \mathbb{E}(w_t^{i,i} - w_t^{i,j}) = 0, \quad (13.28)$$

$$\mathbb{E}_{\mathbb{F}_t} \sum_{j \in N_t^1} b_t^{i,j} (w_t^{i,i} - w_t^{i,j})(\mathbf{f}_t - \bar{f}\mathbf{1}) = \sum_{j \in N_t^1} b_t^{i,j} \mathbb{E}(w_t^{i,i} - w_t^{i,j}) \mathbb{E}(\mathbf{f}_t - \bar{f}\mathbf{1}) = 0, \quad (13.29)$$

и

$$\begin{aligned}
\mathbb{E}_{\mathbb{F}_t} \left( \sum_{j \in N_t^i} b_t^{i,j} (w_t^{i,i} - w_t^{i,j}) \right)^2 &= \sum_{j \in N_t^i} (b_t^{i,j})^2 (\mathbb{E}(w_t^{i,i})^2 + \mathbb{E}(w_t^{i,j})^2) = \\
&= 2\sigma_w^2 \sum_{j \in N_t^i} (b_t^{i,j})^2,
\end{aligned} \quad (13.30)$$

Учитывая полученные соотношения (13.27)–(13.30) и обозначив  $\bar{\mathbf{b}}_t$  вектор составленный из компонент  $\sum_{j \in N_t^1} (b_t^{1,j})^2, \dots, \sum_{j \in N_t^n} (b_t^{n,j})^2$ , из (13.26) выводим

$$\begin{aligned}
\mathbb{E}_{\mathbb{F}_t} \|D_{t+1}\|^2 &= \|(I - \gamma\mathbb{L}(A_{\max}))D_t\|^2 + 2D_t^T(I - \gamma\mathbb{L}(A_{\max}))^T \times \\
& \quad (\gamma(\mathbb{L}(A_{\max})D_t - \mathbb{L}(B_t))D_t + \gamma^2 D_t^T(\mathbb{L}(A_{\max}) - \mathbb{L}(B_t))^T \times \\
& \quad (\mathbb{L}(A_{\max}) - \mathbb{L}(B_t))) + 2\sigma_w^2 \gamma^2 \bar{\mathbf{b}}_t + n\sigma_f^2.
\end{aligned}$$

Обозначим  $\tilde{\mathbb{F}}_t$  —  $\sigma$ -алгебру вероятностных событий, порожденную всеми случайными элементами  $x_0^i, w_0^{i,j}, w_1^{i,j}, \dots, w_{t-1}^{i,j}, f_0^i, f_1^i, \dots, f_{t-1}^i, b_0^{i,j}, b_1^{i,j}, \dots, b_{t-1}^{i,j}$ ,  $i, j \in N, A_0, \dots, A_{t-1}$  реализовавшихся до момента времени  $t$ , и рассмотрим условные математические ожидания от обеих частей последнего

соотношения. В силу стохастических свойствах неопределенностей **A26** и независимости  $B_t$  и  $\bar{\mathbf{b}}_t$  от  $\sigma$ -алгебры  $\tilde{\mathbb{F}}_t$  получаем

$$\begin{aligned} \mathbb{E}_{\tilde{\mathbb{F}}_t} \|D_{t+1}\|^2 &= \|(I - \gamma \mathbb{L}(A_{\max}))D_t\|^2 + \gamma^2 D_t^T Q D_t + 2\sigma_w^2 \gamma^2 \mathbb{E}_{\tilde{\mathbb{F}}_t} \bar{\mathbf{b}}_t + n\sigma_f^2 \leq \\ &\leq (1 - \gamma \operatorname{Re}(\lambda_2(A_{\max})) + \gamma^2 \lambda_{\max}(Q)) \|D_t\|^2 + \\ &\quad + 2\sigma_w^2 \gamma^2 (n^2 \sigma_b^2 + \sum_{i=1}^n \sum_{j=1}^n (b^{i,j})^2) + n\sigma_f^2 = \\ &= (1 - \rho) \|D_t\|^2 + 2\sigma_w^2 \gamma^2 (n^2 \sigma_b^2 + \sum_{i=1}^n \sum_{j=1}^n (b^{i,j})^2) + n\sigma_f^2 = (1 - \rho) \|D_t\|^2 + \Delta. \end{aligned}$$

Перейдя к безусловным математическим ожиданиям, получаем оценки

$$\mathbb{E} \|D_{t+1}\|^2 \leq (1 - \rho) \mathbb{E} \|D_t\|^2 + \Delta,$$

из которых по лемме 1 из главы 2 [13.13] следует неравенство (13.24), являющееся первой частью заключения Теоремы 1.

Второе заключение об асимптотическом среднеквадратичном  $\varepsilon$ -консенсусе получается из неравенства (13.24) при стремлении  $t \rightarrow \infty$ , так как из условия **A4** следует, что  $|1 - \rho| < 1$ , и, следовательно, второе слагаемое в (13.24) экспоненциально стремится к нулю.  $\square$

### 13.3 Имитационное моделирование

#### 13.3.1 Случай с шестью узлами

Для иллюстрации теоретических результатов приведем пример имитационного моделирования для задачи балансировки загрузки децентрализованной вычислительной сети, топология которой приведена на Рис. 1.

Предварительно были заданы начальные состояния  $q_0^1 = 5000$ ;  $q_0^2 = 3500$ ;  $q_0^3 = 2300$ ;  $q_0^4 = 3150$ ;  $q_0^5 = 7400$ ;  $q_0^6 = 1100$  и производительности узлов  $r^1 = 2$ ;  $r^2 = 0,75$ ,  $r^3 = 1,2$ ;  $r^4 = 1,7$ ;  $r^5 = 3,5$ ;  $r^6 = 2,1$ . Производительности узлов со временем не изменялись.

Наблюдения текущих длин очередей проводились на фоне централизованных независимых одинаково равномерно распределенных помех из интервала  $[-500; 500]$ .

Рассмотрим два случая: стационарный и нестационарный (когда система получает новые задания в процессе работы). В первом случае все узлы имеют определенную начальную загрузку, и новые задания

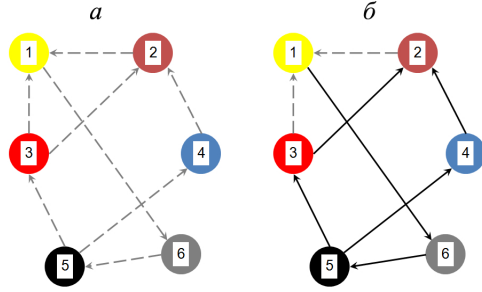


Рис. 13.1: а – максимальное множество каналов связи  $E_{\max}$ ; б – топология сети.

не поступают. Во втором случае в процессе работы алгоритма в систему поступают новые заказы. Используем постоянный размер шага  $\alpha_t = \alpha = 0.1$ . Динамика состояний узлов  $x_t^i$  в результате работы алгоритма (13.12) в обоих случаях показаны на Рис. 13.2 и 13.4. Видно, что поступление новых заказов в систему не меняет качество работы системы. Это большое преимущество алгоритма.

На Рис. 13.2 показано, как работает система в стационарном случае, когда новые задания не поступают в систему с течением времени (есть только начальная загрузка). Каждая линия, соответствующая одному узлу, показывает как меняется загрузка узла  $x_t^i$  с течением времени. Эти линии также показывают, как в системе достигается балансировка загрузки или консенсус.

Теперь вычислим время достижения консенсуса. Мы вычисляем собственные числа и получаем:  $|Re(\lambda_2)| = 0.7737$ . По формуле

$$T(\varepsilon) = \frac{1}{2Re(\lambda_2)} \ln \left( \frac{(n-1)\|x_0 - x^*\mathbf{1}\|^2}{\varepsilon} \right), \quad (13.31)$$

вычисляем время достижения  $\varepsilon$ -консенсуса  $T(\varepsilon)$  для соответствующей непрерывной системы. Если  $\varepsilon = 0.1$ , тогда  $T(\varepsilon) = 12.8883$ . Если  $\varepsilon = 1$  тогда  $T(\varepsilon) = 11.4003$ . Соответствующие значения отмечены на Рис. 13.2.

Результаты сравнения траекторий дискретной стохастической системы (линии из точек) и соответствующей усредненной дискретной модели, описанной выше (пунктирные линии), при постоянном размере шага  $\alpha_t = 0,1$  приведены на Рис. 3.

На Рис. 13.3 приведена иллюстрация того, что мы можем использовать усредненную модель для изучения исходной стохастической

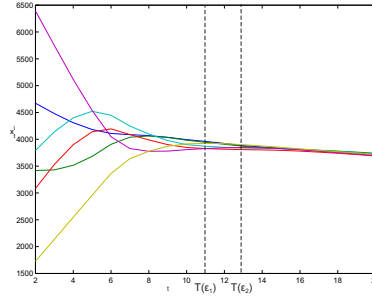


Рис. 13.2. Динамика агентов  $x_t^i$  и время достижения консенсуса

системы. Рис. 13.3 показывает, что траектории дискретной стохастической системы (линии из точек) близки к предельным траекториям соответствующей усредненной дискретной модели (пунктирные линии).

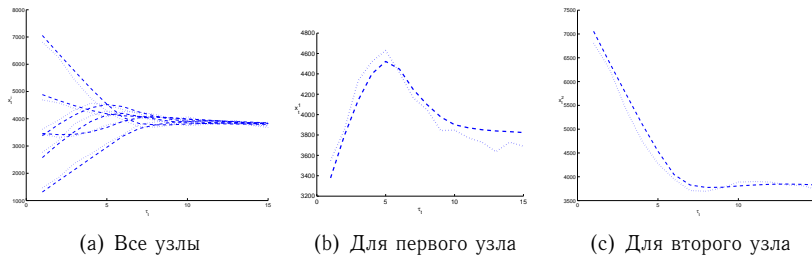


Рис. 13.3: Сравнение траекторий стохастической дискретной системы и соответствующей усредненной модели

Качество работы протокола (13.12) (сходимость траекторий к консенсусу  $x^*$ ) характеризуется средней невязкой  $Err = \sqrt{\sum_i \frac{(x_t^i - x^*)^2}{n}}$ .

На Рис. 2 приведены графики для средних невязок при использовании разных постоянных параметров шага  $\alpha$ . Видно, что чем больше  $\alpha$ , тем состояния узлов быстрее сходятся к консенсусу, но с определенного момента поведение средних невязок становится в значительной степени колебательным.

На Рис. 13.4 показана динамика системы в нестационарном случае, когда в систему могут поступать новые задания с течением времени. Новые задания поступают в случайные моменты времени на случайные узлы. На Рис. 13.4.(а) показано, как система пытается до-



стигнуть консенсуса при использовании протокола локального голосования (13.12), когда с течением времени поступают новые задания. Кроме того, качество работы протокола (13.12) проиллюстрировано на Рис. 13.4.(b), где построена соответствующая средняя невязка. Видно, что средняя невязка меняется с течением времени: он быстро уменьшается и сохраняется на низком уровне до тех пор, пока новые задания не поступают в систему, затем она возрастает и снова начинает снижаться. Результаты имитационного моделирования демонстрируют хорошую производительность системы с протоколом (13.12) в нестационарном случае. Это объясняется свойствами алгоритма стохастической аппроксимации с неубывающим шагом, и каждый момент времени, когда в систему поступают новые задания, может быть рассмотрен как начальный момент времени. В ряде подобных случаев применение стохастической аппроксимации с неубывающим до нуля размером шага при решении нестационарных задач может быть теоретически обоснованы (см., например, [13.6]).

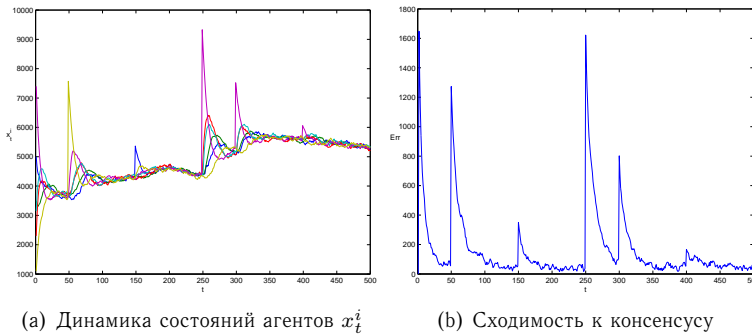


Рис. 13.4. Нестационарный случай для 6 агентов

На Рис. 13.5 показаны графики средней невязки при использовании разных параметров шага  $\alpha$ . На первых четырех графиках использовался постоянный шаг. Видно, что, если увеличивать размер шага, то время до консенсуса уменьшится, но до определенного момента. Однако, если использовать убывающий шаг ( $\alpha_t = 1/t$ ), то скорость сходимости уменьшается со временем.

### 13.3.2 Случай с 1024 узлами

Для того, чтобы показать преимущество перераспределения заданий в сети, а также то, что предложенный нами подход для балансировки загрузки работает и в больших сетях, рассмотрим сеть из

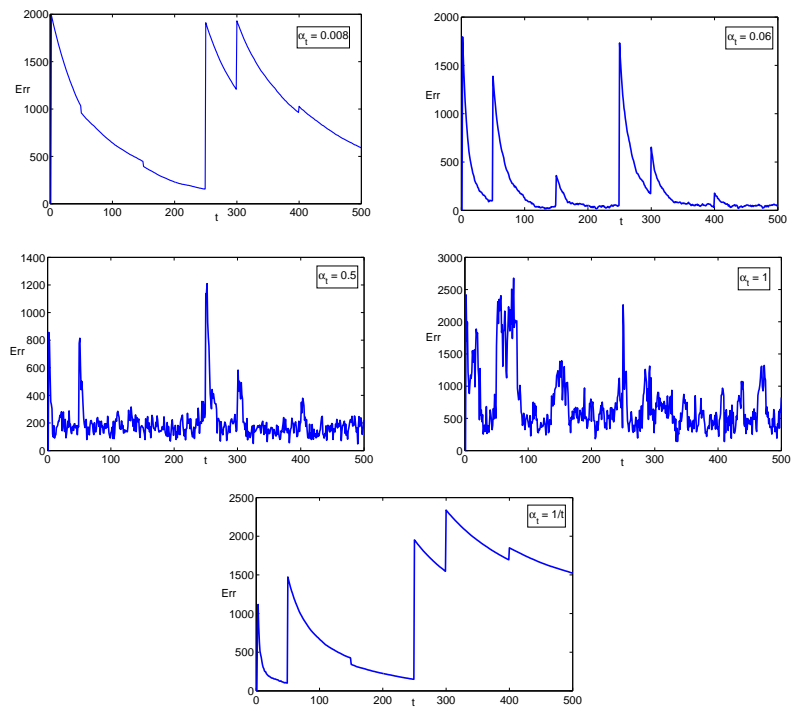


Рис. 13.5. Сходимость к консенсусу при разных параметрах шага.

1024 агентов. Мы сфокусируемся на сравнении производительности системы с перераспределением нагрузки и без перераспределения.

Предполагается, что среднее время между событиями во входном потоке распределено экспоненциально с параметром  $d_{in} = 1/3000$ , а нормированные «трудоемкости» заданий также распределены экспоненциально с параметром  $d_p = 1$  (нормированная «трудоемкость» — время необходимое для выполнения задание на одном узле с производительностью  $p = 1$ ). Количество поступающих заданий равно  $10^6$ . Выбор узла, на который поступает очередное задание выполняется случайно по равномерному распределению из 1024 узлов.

Агенты связаны между собой по кругу, а также на каждой итерации между агентами устанавливается  $n$  случайных связей, которые переключаются с течением времени. Пример такой сети приведен на Рис. 13.6.

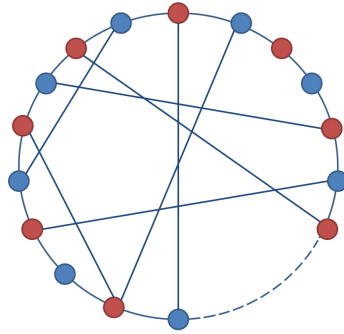


Рис. 13.6. Пример топологии сети.

Рассмотрим случай, когда все задания поступают в разные моменты времени на интервале от 1 до 2000. На Рис. 13.7 и 13.8 показаны типичные результаты моделирования, в которых задания в момент поступления посылаются случайно на какой-то из узлов. На этих рисунках сплошные линии соответствуют варианту с перераспределением заданий по протоколу локального голосования, а пунктирные — случаю без перераспределения, где  $|D(t)|$  — максимальное отклонение от средней загрузки в сети. Видно, что адаптивная мультиагентная стратегия с перераспределением заданий между «связанными соседями» существенно лучше справляется с распределением заданий, чем стратегия без перераспределения.

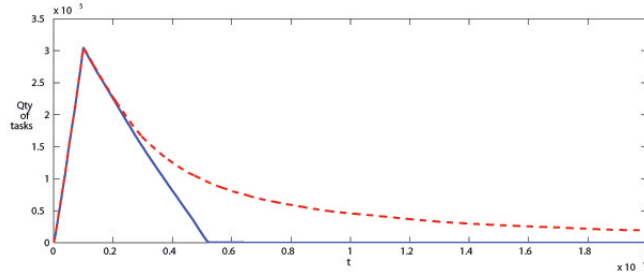


Рис. 13.7: Количество заданий в очереди в случае, когда задания поступают в разные моменты времени.

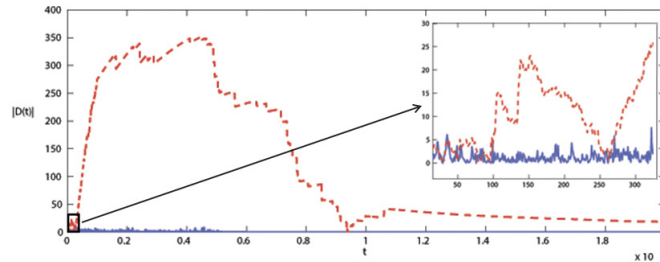


Рис. 13.8. Максимальное отклонение от средней загрузки в сети.

### 13.4 Заключение

В статье была рассмотрена задача балансировки загрузки мульти-агентной системы в условиях стохастических неопределенностей для агентов с нелинейной динамикой, помехами и задержками в измерениях и переключающейся топологией. Для решения был предложен консенсусный протокол локального голосования и проведено имитационное моделирование работы алгоритма для вычислительной сети. Для более общей постановки задачи в [13.11] для балансировки загрузки сети в нестационарном случае предложен робастный протокол локального голосования, установлены условия достижения приближенного баланса загрузки сети, получена оценка асимптотического уровня субоптимальности протокола.

Работа выполнена при финансовой поддержке Российского Фонда Фундаментальных Исследований (проекты № 13-07-00250-а, 11-08-01218-а).

### 13.5 Литература к главе 13

- [13.1] *Armbruster D., Mikhailov A.S., Kaneko K. (eds.) Networks of Interacting Machines: Production Organization in Complex Industrial Systems and Biological Cells — World Scientific. Singapore. 2005. 267 p.*
- [13.2] *Glashenko A., Inozemtzev S., Grachev I., Skobelev P. Magenta Technology: case studies of Magenta i-scheduler for road transportation // Proc. Int. Conf. Autonom. Agents Multi Agent Syst. (AAMAS-6). Hawaii. 2007. P. 1385–1392.*
- [13.3] *Граничин О.Н., Поляк Б.Т. Рандомизированные алгоритмы оценивания и оптимизации при почти произвольных помехах — М.: Наука. 2003. 291 с.*
- [13.4] *Граничин О.Н. Процедура стохастической аппроксимации с возмущением на входе // Автоматика и телемеханика. 1992. №2. С. 97–104.*
- [13.5] *Граничин О.Н. Рандомизированные алгоритмы стохастической аппроксимации при произвольных помехах // Автоматика и телемеханика. 2002. №2. С. 44–55.*
- [13.6] *Вахитов А.Т., Граничин О.Н., Гуревич Л.С. Алгоритм стохастической аппроксимации с пробным возмущением на входе в нестационарной задаче оптимизации // Автоматика и телемеханика. 2009. №11. С. 70–79.*
- [13.7] *Амелина Н.О. Диспетчеризация сети с переменной топологией при помехах и задержках в измерениях // Вестник СПбГУ. Сер. 1: Математика. Механика. Астрономия. 2012. №2. С. 11–15.*
- [13.8] *Амелин К.С., Амелина Н.О., Граничин О.Н., Корявко А.В. Применение алгоритма локального голосования для достижения консенсуса в децентрализованной сети интеллектуальных агентов // Нейрокомпьютеры: разработка, применение. №11. 2012. С. 039-047.*
- [13.9] *Амелина Н., Лада А., Майоров И. и др. Исследование моделей организации грузовых перевозок с применением мультиагентной системы адаптивного планирования грузовиков в реальном времени // Проблемы управления. 2011. №6. С. 31–37.*

- [13.10] *Амелина Н.О., Фрадков А.Л.* Приближенный консенсус в стохастической динамической сети с неполной информацией и задержками в измерениях // Автоматика и телемеханика. 2012. №11. С. 6-29.
- [13.11] *Амелина Н.О.* Применение протокола локального голосования для децентрализованной балансировки загрузки сети с переменной топологией и помехами в измерениях // Вестник СПбГУ. Сер. 1: Математика. Механика. Астрономия. 2013. №3. С. 12-20.
- [13.12] *Амелина Н.О.* Мультиагентные технологии, адаптация, самоорганизация, достижение консенсуса // Стохастическая оптимизация в информатике. Вып. 7. 2011. С. 149–185.
- [13.13] *Поляк Б.Т.* Введение в оптимизацию. М.: Наука, 1983, 384 с.
- [13.14] *Амелина Н.О.* Применение протокола локального голосования для децентрализованной балансировки загрузки сети с переменной топологией и помехами в измерениях // Вестник СПбГУ. Сер. 1: Математика. Механика. Астрономия. 2013. №3. С. 12-20.
- [13.15] *Amelina N., Granichin O., and Kornivets A.* Local Voting Protocol in Decentralized Load Balancing Problem with Switched Topology, Noise, and Delays // Proc. of 52st IEEE Conference on Decision and Control. December 10-13, 2013. Firenze, Italy. P. 4613-4618.