УДК 519.2

АЛГОРИТМ РАСПАРАЛЛЕЛИВАНИЯ ПРОЦЕССА ОПТИМИЗАЦИИ ФУНКЦИИ КВАНТИЛИ

Андрей Иванович КИБЗУН родился в 1951 г. в городе Хабаровске. Профессор МАИ. Доктор физико-математических наук, профессор. Основные научные интересы — в области стохастического программирования. Автор более 200 научных работ.

Andrey I. KIBZUN, D.Sci, was born in 1951, in Khabarovsk. He is a Professor at the MAI. His research interests are in stochastic programming. He has published over 200 technical papers.

Евгений Леонидович МАТВЕЕВ родился в 1984 г. в городе Москве. Аспирант МАИ. Основные научные интересы — в области стохастического программирования. Автор одной научной работы.

Eugeny L. MATVEYEV, was born in 1984, in Moscow. He is a Postgraduate Student at the MAI. His research interests are in stochastic programming. He has published 1 technical paper.

Предлагается алгоритм распараллеливания процесса минимизации функции квантили, базирующийся на использовании децентрализованного алгоритма оценивания квантили. Доказывается сходимость алгоритма с вероятностью единица.

Введение

В многочисленных задачах экономики и авиационной техники возникает функция квантили [1], которая характеризует гарантированные с заданной вероятностью потери. Квантильные постановки являются полезными в некоторых экономических приложениях, например при оптимизации портфеля ценных бумаг при вероятностных ограничениях [2]. Функция квантили с математической точки зрения оказывается довольно сложным объектом, в отличие, скажем, от математического ожидания. Поэтому трудно предложить эффективные вычислительные процедуры для решения задачи квантильной оптимизации.

Для решения задач квантильной оптимизации используются различные методы. Алгоритмы типа стохастической аппроксимации предложены в [1, 3]. Данные алгоритмы основаны на выборочных оценках функции квантили и требуют дополнительной процедуры сглаживания, что приводит к дополнительным вычислениям и потерям точности. В [4] предложен алгоритм, а в [5] доказана сходимость алгоритма, базирующегося на неантагонистической игре двух лиц и сводящегося к минимизации функции квантили с помощью стохастического квазиградиентного алгоритма типа Эрроу—Гурвица. Однако применение данного алгоритма затруднено тем, что его сходимость доказана при труднопроверяемых условиях. Задачи квантильной оптимизации исследовались в рамках доверительного подхода [1, 3], сводящего задачу квантильной оптимизации к обобщенной минимаксной задаче. Такие алгоритмы позволяют решить задачу квантильной оптимизации приближенно, однако на практике трудно оценить погрешность такого приближения. В основе изложенных алгоритмов квантильной оптимизации лежит построение стохастического квазиградиента вида, предложенного в [6], а сами алгоритмы являются по сути рекуррентными, очень медленно сходящимися.

В данной статье предлагается алгоритм распараллеливания процесса минимизации функции квантили, базирующийся на использовании децентрализованного алгоритма оценивания квантили. Доказывается сходимость алгоритма с вероятностью единица.

1. Постановка задачи

Пусть X — случайная величина с функцией распределения $F_X(x) \triangleq P\{X \le x\}$. Для $\alpha \in (0,1)$ квантиль уровня α определим следующим образом:

$$x_\alpha \triangleq \min\{x \in R^1 : F_X(x) \ge \alpha\}. \quad (1)$$

Если $F_X(x)$ непрерывна, то $F_X(x_\alpha) = \alpha$.

Пусть X_i, $i = \overline{1,n}$ — выборка, образованная из независимых одинаково распределённых случайных величин с функцией распределения $F_X(x)$. Предположим, что X имеет плотность вероятности $f_X(x)$ такую, что $f_X(x)>0$ для всех $x \in R^1$. Пусть $X_{(1)}, X_{(2)}, \ldots, X_{(n)}$ — порядковые статистики, образованные выборкой X_i, $i=\overline{1,n}$. Тогда выборочной оценкой квантили называется величина

$$\hat{X}_\alpha(n) \triangleq X_{([n\alpha]+1)}. \qquad (2)$$

Известно [7], что при сделанных предположениях выборочная оценка квантили будет сильно состоятельна, т.е. $\hat{X}_\alpha(n) \xrightarrow{\text{п.н.}} x_\alpha$. Более того, имеется сходимость по распределению

$$\sqrt{n}\left(\hat{X}_\alpha(n) - x_\alpha\right) \xrightarrow{d} \mathrm{N}\left(0, \frac{\alpha(1-\alpha)}{f_X^2(x_\alpha)}\right). \qquad (3)$$

Рассмотрим последовательность случайных величин, определяемую рекуррентным соотношением

$$Z_{n+1} = Z_n + \rho_i H(Z_n, X_{n+1}), \; z_0 \in R^k, \; n = 0,1,\ldots, \qquad (4)$$

где числовая последовательность ρ_n, $n=1,2,\ldots$, удовлетворяет следующим условиям:

$$\rho_n > 0, \; \sum_{n=0}^{\infty} \rho_n = \infty, \; \sum_{n=0}^{\infty} \rho_n^2 < \infty, \qquad (5)$$

а $H(z,x): R^k \times R^k \to R^k$ — некоторая вектор-функция. Введём обозначения

$$h(z) \triangleq \mathbf{M}[H(z,X) \,|\, z], \; R(z) \triangleq \mathbf{cov}[H(z,X) \,|\, z]. \qquad (6)$$

Рассмотрим следующие условия, которым должна удовлетворять функция $H(z,x)$, и последовательность случайных величин (условия «псевдоградиентности» и ограниченности):

$$\exists! z^* \text{ такое, что } \sup_{\varepsilon \le |z-z^*| \le \frac{1}{\varepsilon}} (z-z^*)h(z) < 0 \; \forall \varepsilon > 0; \qquad (7)$$

$$\exists C > 0 \text{ такое, что}$$
$$\int_{-\infty}^{\infty} |H(z,x)|^2 \, dF_X(x \,|\, z) \le C(1+|z|^2). \qquad (8)$$

Известен следующий результат [8, 9].

Теорема 1. Пусть выполнены условия (7) и (8), а числовая последовательность ρ_n удовлетворяет условию (5). Тогда последовательность z_n, образованная согласно (4), сходится почти наверное (п.н.) к z^*.

Если условие (8) заменить условием

$$\sup_{|z| \le r} \int_{-\infty}^{\infty} |H(z,x)|^2 \, dF_X(x \,|\, z) < \infty \quad \forall r > 0, \qquad (9)$$

а условие (7) оставить без изменений, то z_n сходится почти наверное к z^* на множестве $\{\sup_n |z_n| < \infty\}$ (см. [8]). Известен следующий результат [8].

Теорема 2. Пусть $n\rho_n \to \text{const}$ при $n \to \infty$ и все собственные значения матрицы $\left.\frac{1}{2}I + \frac{dh(z)}{dz}\right|_{z=z^*}$ имеют отрицательные действительные части, где $\frac{dh(z)}{dz}: R^k \to R^k$ — матрица Якоби отображения $h(z): R^k \to R^k$, а I — единичная матрица размера $k \times k$. Тогда

$$\sqrt{n}(Z_n - z^*) \xrightarrow{d} \mathrm{N}(0, K), \qquad (10)$$

где K — единственное решение матричного уравнения

$$\left(\frac{I}{2} + \left.\frac{dh(z)}{dz}\right|_{z=z^*}\right)K + K\left(\frac{I}{2} + \left.\frac{dh(z)}{dz}\right|_{z=z^*}\right)^T + $$
$$+ R(z^*) = 0. \qquad (11)$$

2. Децентрализированный алгоритм оценивания квантили

Предложим рекуррентный децентрализованный алгоритм оценивания квантили x_α.

Предположим, что имеющаяся в нашем распоряжении вычислительная сеть (рис. 1) состоит из m процессоров, которые накапливают информацию о выборке, и вычислительного блока, который обрабатывает данные, поступающие от процессоров.

Более точно, предположим, что каждый процессор $i \in \{1,\ldots,m\}$ содержит независимые наблюдения $x(i)$ случайной величины X с функцией распределения $F_X(x) = P\{X \le x\}$. На каждом шаге $n = 0,1,\ldots$ каждый процессор $i = 1,2,\ldots,m$ передаёт 1 бит

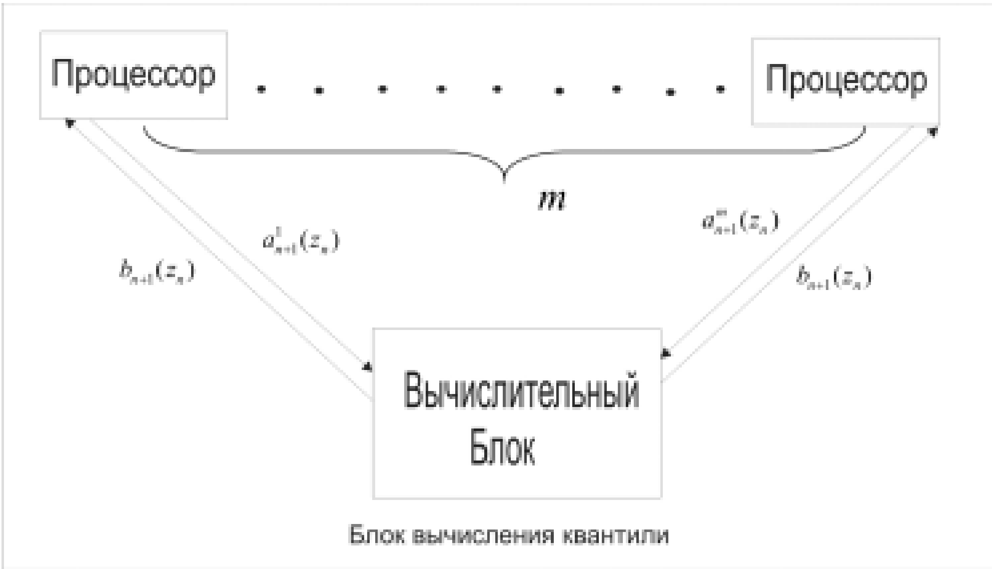

Рис. 1. Схема оценивания квантили

$a^i_{n+1}(z_n)$, вычисленный по определённому алгоритму, описанному ниже, в вычислительный блок. Вычислительный блок обрабатывает полученную информацию и передает 1 бит $b_{n+1}(z_n)$, вычисляемый определённым образом, назад каждому из процессоров, в которых уточняется текущая оценка квантили.

Начальное значение всех процессоров одинаковое. Пусть $\eta_m > 0$ — некоторая нормирующая константа.

Рассмотрим следующий алгоритм:

1. Каждый i-й процессор формирует реализацию $x_{n+1}(i)$ случайной величины X_{n+1} и вычисляет значение индикаторной функции

$$a^i_{n+1}(z_n) \triangleq \begin{cases} 1, \text{ если } x_{n+1}(i) \le z_n; \\ 0, \text{ иначе}, \end{cases} \tag{12}$$

а затем передаёт её в вычислительный блок.

2. Вычислительный блок вычисляет значение индикаторной функции по усреднённой величине, характеризующей полученную информацию от всех процессоров:

$$b_{n+1}(z_n) \triangleq \begin{cases} 1, \text{если } \dfrac{1}{m}\sum_{i=1}^{m} a^i_{n+1}(z_n) \le \alpha; \\ 0, \text{иначе}, \end{cases} \tag{13}$$

и передаёт её обратно каждому процессору.

3. Каждый процессор вычисляет новое значение z_{n+1} по формуле

$$z_{n+1} = z_n + \rho_n \eta_m \left(b_{n+1}(z_n) - \beta\right), \tag{14}$$

где

$$\beta \triangleq \sum_{i=0}^{[m\alpha]} C^i_m \alpha^i (1-\alpha)^{m-i}. \tag{15}$$

Теорема 3. Рассмотрим последовательность Z_n, образованную алгоритмом. Тогда:

1) для произвольного начального состояния z_0

$$Z_n \xrightarrow{\text{п.н.}} x_\alpha;$$

2) если нормирующая константа η_m выбрана таким образом, что

$$\eta_m > \frac{1}{2 f_X(x_\alpha) m C^{[m\alpha]}_{m-1} \alpha^{[m\alpha]} (1-\alpha)^{m-[m\alpha]-1}}, \tag{16}$$

то

$$\sqrt{n}\left(Z_n - x_\alpha\right) \xrightarrow{d} \xrightarrow{d} \mathrm{N}\left(0, \frac{\eta_m^2 \beta(1-\beta)}{2\eta_m f_X(x_\alpha) m C^{[m\alpha]}_{m-1} \alpha^{[m\alpha]} (1-\alpha)^{m-[m\alpha]-1} - 1}\right), \tag{17}$$

где β определена в (15).

Доказательство теоремы 3.

Введём вспомогательное обозначение:

$$g_m(x) \triangleq \sum_{i=0}^{[m\alpha]} C^i_m x^i (1-x)^{m-i},\ x \in (0,1). \tag{18}$$

Докажем техническую лемму.

Лемма 1. Функция $g_m(x)$, введённая в (18), является неотрицательной, дифференцируемой и монотонно убывающей по *x*. При этом

$$\frac{dg_m(x)}{dx} = -mC_{m-1}^{[m\alpha]}x^{[m\alpha]}(1-x)^{m-[m\alpha]-1}. \quad (19)$$

Доказательство леммы 1. Неотрицательность и дифференцируемость следуют из определения (18). Докажем, что $g_m(x)$ монотонно убывает по $x \in (0,1)$. Учитывая следующие свойства биномиальных коэффициентов:

$$iC_m^i = mC_{m-1}^{i-1} \text{ и } (m-i)C_m^{m-1} = mC_{m-1}^{m-i-1},$$

вычислим $\frac{dg_m(x)}{dx}$:

$$\begin{aligned}\frac{dg_m(x)}{dx} &= \sum_{i=0}^{[m\alpha]} C_m^i [ix^{i-1}(1-x)^{m-i} - (m-i)x^i(1-x)^{m-i-1}] = \\ &= \sum_{i=0}^{[m\alpha]} \left[C_m^i ix^{i-1}(1-x)^{m-i} - C_m^{m-i}(m-i)x^i(1-x)^{m-i-1}\right] = \\ &= \sum_{i=0}^{[m\alpha]} \left[mC_{m-1}^{i-1}x^{i-1}(1-x)^{m-i} - mC_{m-1}^{m-i-1}x^i(1-x)^{m-i-1}\right] = \\ &= \sum_{i=0}^{[m\alpha]} m\left(C_{m-1}^{i-1}x^{i-1}(1-x)^{m-i} - C_{m-1}^{i}x^i(1-x)^{m-i-1}\right) = \\ &= \sum_{i=0}^{[m\alpha]} -m\left(C_{m-1}^{i}x^{i}(1-x)^{m-i-1} - C_{m-1}^{i-1}x^{i-1}(1-x)^{m-i}\right) = \\ &= -mC_{m-1}^{[m\alpha]}x^{[m\alpha]}(1-x)^{m-[m\alpha]-1} < 0.\end{aligned} \quad (20)$$

■

Для доказательства сходимости алгоритма (14) воспользуемся результатом теоремы 1. Для это вычислим значение функции $h(z)$. Учитывая (14), получим

$$\begin{aligned}h(z) &= \eta_m \mathbf{M}[b_{n+1}(z) - \beta \mid z] = \\ &= \eta_m \left(g_m(F_X(z)) - g_m(F_X(x_\alpha))\right).\end{aligned} \quad (21)$$

Проверим выполнение условия (7). Условие (7) будет выполнено, если для всех $z \neq x_\alpha$

$$h(z)\left(z - x_\alpha\right) < 0. \quad (22)$$

С учетом (21) соотношение (22) примет следующий вид:

$$\eta_m \left(g_m(F_X(z)) - g_m(F_X(x_\alpha))\right)\left(z - x_\alpha\right) < 0, \quad (23)$$

так как, функция распределения $F_X(x)$ — монотонно возрастающая функция, а функция $g_m(x)$ — монотонно убывающая по *x* (в силу леммы 1).

Покажем, что $R(z)$, введённая в (6), ограничена:

$$\begin{aligned}R(z) &= \mathbf{D}\left[\eta_m\left(b_{n+1}(z) - \beta\right) \mid z\right] = \eta_m^2 \mathbf{D}\left[b_{n+1}(z) \mid z\right] = \\ &= \eta_m^2 g_m(F(z))\left(1 - g_m(F(z))\right) \le \frac{\eta_m^2}{4} < \infty.\end{aligned} \quad (24)$$

Таким образом, все условия теоремы 1 выполнены. Поэтому последовательность, сгенерированная алгоритмом (14), сходится п.н., т.е. $z \xrightarrow{\text{п.н.}} x_\alpha$.

Для доказательства (17) проверим выполнение условия теоремы 2. В данном случае условие отрицательности собственных значений эквивалентно условию

$$\left.\frac{dh(z)}{dz}\right|_{z=x_\alpha} < -\frac{1}{2}. \quad (25)$$

Учитывая, что

$$h(z) = \eta_m\left[g_m(F_X(z)) - g_m(F_X(x_\alpha))\right],$$

получим

$$\left.\frac{dh(z)}{dz}\right|_{z=x_\alpha} = \eta_m \left.\frac{dg_m(x)}{dx}\right|_{x=\alpha} f_X(x_\alpha). \quad (26)$$

Заметим, что согласно (19)

$$\left.\frac{dg_m(x)}{dx}\right|_{x=\alpha} = -mC_{m-1}^{[m\alpha]}\alpha^{[m\alpha]}(1-\alpha)^{m-[m\alpha]-1}.$$

Таким образом, если выбирать η_m из условия

$$\eta_m > \frac{1}{2f_X(x_\alpha)mC_{m-1}^{[m\alpha]}\alpha^{[m\alpha]}(1-\alpha)^{m-[m\alpha]-1}}, \quad (27)$$

то условие (25) будет выполнено.

В одномерном случае решение уравнения (11) определяется следующим образом:

$$K = \frac{R(x_\alpha)}{-2\left.\frac{dh(z)}{dz}\right|_{z=x_\alpha} - 1}.$$

Заметим, что $\left.\frac{dh(z)}{dz}\right|_{z=x_\alpha} = \eta_m \left.\frac{dg_m(x)}{dx}\right|_{x=\alpha} f_X(x_\alpha)$.

Тогда окончательно получим

$$K = \frac{\eta_m^2 g_m(\alpha)(1-g_m(\alpha))}{2\eta_m f_X(x_\alpha) m C_{m-1}^{[m\alpha]} \alpha^{[m\alpha]} (1-\alpha)^{m-[m\alpha]-1} - 1}. \quad (28)$$

Учитывая, что $g_m(\alpha)=\beta$, окончательно получим

$$K = \frac{\eta_m^2 \beta(1-\beta)}{2\eta_m f_X(x_\alpha) m C_{m-1}^{[m\alpha]} \alpha^{[m\alpha]} (1-\alpha)^{m-[m\alpha]-1} - 1}, \quad (29)$$

где β определена в (15).

■

3. Сравнение оценок

Рассмотрим оценку квантили, полученную с помощью алгоритма (14) при $m=1$, и сравним её с выборочной оценкой квантили (3).

Для этого найдём решение уравнения (11) в одномерном случае. Учитывая (6), получим

$$K = \frac{R(x_\alpha)}{-2\left.\frac{dh(z)}{dz}\right|_{z=x_\alpha} - 1}. \quad (30)$$

При $m=1$ последовательность, образованная алгоритмом (14), будет выглядеть следующим образом:

$$z_{n+1} = z_n + \rho_n \eta\left(b_{n+1}(z_n) - \beta\right), \quad (31)$$

где $\beta = 1-\alpha$;

$$b_{n+1}(z_n) = \begin{cases} 1, \text{ если } a_{n+1}(z_n) \le \alpha; \\ 0, \text{иначе}, \end{cases} \quad (32)$$

где $a_{n+1}(z_n) \sim \mathrm{Bi}(1, F_X(z_n))$.

Таким образом, учитывая что $\alpha \in (0,1)$, получим $b_{n+1}(z_n) \sim \mathrm{Bi}(1, 1-F_X(z_n))$.

Поэтому согласно (6) находим

$$h(z) = \eta M\left[b_{n+1}(z) - 1 + \alpha \mid z\right] = \eta\left(\alpha - F_X(z)\right),$$

$$R(z) = \eta^2 D[b_{n+1}(z) - \beta \mid z] = \eta^2 D[b_{n+1}(z) \mid z] =$$
$$= \eta^2 F_X(z)(1-F_X(z)).$$

Таким образом, $R(x_\alpha) = \eta^2\alpha(1-\alpha)$, так как $F_X(x_\alpha) = \alpha$.

При этом $\frac{dh(z)}{dz} = -\eta f_X(z)$. Тогда выражение (30) примет вид

$$K = \frac{\eta^2\alpha(1-\alpha)}{2\eta f_X(x_\alpha) - 1} \quad (33)$$

при условии $\left.\frac{dh(z)}{dz}\right|_{z=x_\alpha} < -1/2$, которое эквивалентно следующему: $\eta f_X(x_\alpha) > 1/2$. Таким образом, получим

$$\sqrt{n}\left(z_n - x_\alpha\right) \xrightarrow{d} \mathrm{N}\left(0, \frac{\eta^2\alpha(1-\alpha)}{2\eta f_X(x_\alpha) - 1}\right). \quad (34)$$

Представим η в следующем виде:

$$\eta = \frac{\gamma}{f_X(x_\alpha)},\ \gamma \in R^1.$$

Найдём такое γ, при котором будет достигаться наименьшее отклонение оценки от истинного значения квантили. Для этого решим следующую задачу:

$$K \to \min \text{ при условии } 2\eta f_X(x_\alpha) > 1. \quad (35)$$

Данная задача эквивалентна следующей:

$$\frac{\alpha(1-\alpha)}{f_X^2(x_\alpha)} \frac{\gamma^2}{2\gamma - 1} \to \min_{\gamma:\gamma>1/2}. \quad (36)$$

Для нахождения минимума найдём производную по γ:

$$\frac{d}{d\gamma}\left(\frac{\gamma^2}{2\gamma-1}\right) = \frac{2\gamma(\gamma-1)}{(2\gamma-1)^2} = 0. \quad (37)$$

Учитывая знакопостоянность производной на интервалах (1/2,1) и (1, ∞), можно сделать вывод, что минимум достигается при $\gamma = 1$. Таким образом, минимальное среднеквадратическое отклонение достигается при $\eta = 1/f_X(x_\alpha)$. При этом получим

$$\sqrt{n}\left(z_n - x_\alpha\right) \xrightarrow{d} \mathrm{N}\left(0, \frac{\alpha(1-\alpha)}{f_X^2(x_\alpha)}\right). \quad (38)$$

Заметим, что найденное выражение для скорости сходимости оценки квантили, полученной с помощью алгоритма (14) при $m = 1$, совпадает со скоростью сходимости оценки квантили, полученной при использовании выборочной оценки (3).

Рассмотрим поведение дисперсии оценки, полученной при использовании алгоритма (14), при $m \gg 1$. Используя формулу Стирлинга, заметим, что

$$\frac{1}{mC_{m-1}^{[m\alpha]}\alpha^{[m\alpha]}(1-\alpha)^{m-[m\alpha]-1}} = \frac{1}{m\alpha^{m\alpha}(1-\alpha)^{m-m\alpha-1}} \times$$

$$\times\sqrt{2\pi m\alpha}(m\alpha)^{m\alpha}e^{-m\alpha}\sqrt{2\pi(m-1-m\alpha)}\times$$

$$\times\frac{(m-m\alpha-1)^{m-m\alpha-1}e^{-m+m\alpha+1}}{\sqrt{2\pi(m-1)}(m-1)^{m-1}e^{-m+1}} + O\left(\frac{1}{m}\right) =$$

$$= \frac{\sqrt{2\pi\alpha(1-\alpha)m}}{m\alpha^{m\alpha}(1-\alpha)^{m-m\alpha-1}}\left(\frac{m-m\alpha-1}{m-1}\right)^{m-1}\left(\frac{m\alpha}{m-m\alpha-1}\right)^{m\alpha} +$$

$$+O\left(\frac{1}{m}\right) = \sqrt{\frac{2\pi\alpha(1-\alpha)}{m}} + O\left(\frac{1}{m}\right) = O\left(\frac{1}{\sqrt{m}}\right). \quad (39)$$

Рассмотрим поведение величины β, определённой в (15), при $m \gg 1$. Так как

$$\beta = \sum_{i=0}^{[m\alpha]} C_m^i \alpha^i (1-\alpha)^{m-i} = P\{Y \le [m\alpha]\},$$

где $Y \sim \mathrm{Bi}(m,\alpha)$, то согласно теореме Муавра—Лапласа [10] получим, что

$$\beta \to \Phi\left(\frac{m\alpha - [m\alpha]}{\sqrt{m\alpha(1-\alpha)}}\right) \to \frac{1}{2}. \quad (40)$$

С учетом (39) и (40) выражение для дисперсии оценки, полученной при помощи алгоритма (14), при $m \gg 1$ примет следующий вид:

$$K = \frac{\eta_m^2}{8\eta_m f_X(x_\alpha)\sqrt{\frac{m}{2\pi\alpha(1-\alpha)}} - 4} + O\left(\frac{1}{m}\right) = O\left(\frac{1}{\sqrt{m}}\right). \quad (41)$$

Таким образом, можно сделать вывод, что при увеличении числа процессоров среднеквадратическое отклонение оценки, полученной с помощью алгоритма (14), будет стремиться к нулю.

Интересным представляется вопрос об оптимальном (в смысле минимума среднеквадратического отклонения) выборе нормирующей константы η_m в общем случае. Так как значение η_m зависит от $f_X(x_\alpha)$, которое неизвестно, то нельзя точно вычислить значение нормирующей константы, используя соотношение (16). Возможно использование адаптивной схемы оценивания, которая автоматически оценивает $f_X(x_\alpha)$ (и η_m соответственно), обеспечивая минимум среднеквадратического отклонения. Мы оставляем этот вопрос открытым для будущих работ.

4. Алгоритм минимизации функции квантили

Пусть X — случайный вектор с известной функцией распределения, а $\Phi(u,x): R^r \times R^s \to R^1$ — непрерывная по совокупности аргументов действительная функция. Определим функции вероятности и квантили следующим образом:

$$P_\varphi(u) \triangleq P\{X : \Phi(u,X) \le \varphi\}, \varphi \in R^1; \quad (42)$$

$$\varphi_\alpha(u) \triangleq \min\{\varphi : P_\varphi(u) \ge \alpha\}, \quad \alpha \in (0,1). \quad (43)$$

Рассмотрим задачу безусловной минимизации функции квантили:

$$u_\alpha = \arg\min_{u \in U} \varphi_\alpha(u), \quad (44)$$

где U — множество допустимых решений, которое предполагается выпуклым компактом в R^r. Пусть для любого $u \in U$ есть возможность получить выборку $\{\Phi(u,X_i)\}_{i=1}^{t_k}$ значений функции $\Phi(u,X)$. Обозначим через $\{\Phi_{(i)}(u)\}_{i=1}^{t_k}$ вариационный ряд выборки, т.е.

$$\Phi_{(1)}(u) \le \Phi_{(2)}(u) \le \ldots \le \Phi_{(t_k)}(u).$$

Тогда, аналогично (2), выборочной оценкой функции квантили называется величина

$$\hat{\varphi}_{t_k}(u) \triangleq \Phi_{([t_k\alpha]+1)}(u). \quad (45)$$

Назовем стохастическим квазиградиентом функции квантили случайный вектор

$$\xi_k(u,\delta_k) \triangleq \frac{1}{2\delta_k}\sum_{j=1}^{r}\Big[\hat{\varphi}_{t_k}(\tilde{u}_1,\ldots,u_j+\delta_k,\ldots,\tilde{u}_r) - \\ -\hat{\varphi}_{t_k}(\tilde{u}_1,\ldots,u_j-\delta_k,\ldots,\tilde{u}_r)\Big]e_j, \quad (46)$$

где $\{\delta_k\}$ — некоторая детерминированная последовательность ($\delta_k \to 0$, $\delta_k > 0$), $\hat{\varphi}_{t_k}(u)$ — выборочная оценка вида (45), $\tilde{u}_l$, $l=\overline{1,r}$, $l \neq j$ — независимая случайная величина и $\tilde{u}_j \sim \mathbf{R}[u_i - \delta_k, u_i + \delta_k]$, e_j — единичный орт. Пусть последовательности ρ_k, δ_k, t_k удовлетворяют следующим условиям:

$$\rho_k > 0, \quad \sum_{k=1}^{\infty} \rho_k = \infty, \quad \sum_{k=1}^{\infty} \rho_k^2 < \infty,$$

$$\delta_k \to 0, \quad t_k \to \infty, \quad \delta_k t_k \to \infty, \quad \frac{k}{\delta_k t_k^2} \to 0. \qquad (47)$$

Например, последовательности

$$\rho_k = \frac{\rho_0}{k}, \; \delta_k = \delta_0 k^{-1/5}, \; t_k = t_0 + k^{\frac{3}{2}} \qquad (48)$$

удовлетворяют ограничениям (47).

Обозначим через Π_U оператор проецирования вектора $u \in R^r$ на область U. Рассмотрим последовательность вида

$$u_{k+1} = \begin{cases} \Pi_U\left(u_k - \rho_k \xi_k(u_k, \delta_k)\right), & \|\xi_k(u_k, \delta_k)\| \leq L; \\ u_1, & \|\xi_k(u_k, \delta_k)\| > L, \end{cases} \qquad (49)$$

где $L \in R^1$ — некоторое большое число.

Известно [1], что последовательность, образованная согласно (49), сходится п.н. к u_α.

Рассмотрим алгоритм поиска минимума функции квантили на основе децентрализованной оценки квантили, описанной в разд. 3, на примере структуры, изображенной на рис. 2.

Пусть вычислительная сеть состоит из двух основных блоков: блока обработки результатов и блока вычисления квантили, состоящего из m процессоров и вычислительного блока. В блоке обработки результатов происходит уточнение текущей оценки минимума функции квантили, а в блоке вычисления квантили по текущему значению u_k происходит вычисление новой оценки квантили, согласно алгоритму (14).

Более точно, пусть числовые последовательности t_k, ρ_k, δ_k удовлетворяют условиям (47).

Рассмотрим следующий алгоритм:

1. Блок обработки результатов формирует вектор

$$u_{k,j}^1 \triangleq col\left(\tilde{u}_{k1}, \ldots, u_{kj} + \delta_k, \ldots, \tilde{u}_{kr}\right),$$

где $\tilde{u}_{kl} \sim \mathbf{R}[u_{kl} - \delta_k, u_{kl} + \delta_k]$, $l = \overline{1,r}$, $l \neq j$, и передаёт его в процессоры. Положим $\hat{u}_{k,j} = u_{k,j}^1$.

2. Осуществляется итерационный процесс уточнения текущего значения квантили. При этом на каждой n-й итерации, $n = \overline{0, t_k - 1}$, каждый i-й процессор формирует реализацию $x_{n+1}(i)$ случайного вектора X_{n+1}, вычисляет значение индикаторной функции

$$a_{n+1}^i(\tilde{\varphi}_n^i(\hat{u}_{k,j})) \triangleq \begin{cases} 1, \text{ если } \Phi\left(\hat{u}_{k,j}, x_{n+1}(i)\right) \leq \tilde{\varphi}_n^i(\hat{u}_{k,j}); \\ 0, \text{ иначе} \end{cases} \qquad (50)$$

и передаёт её в вычислительный блок.

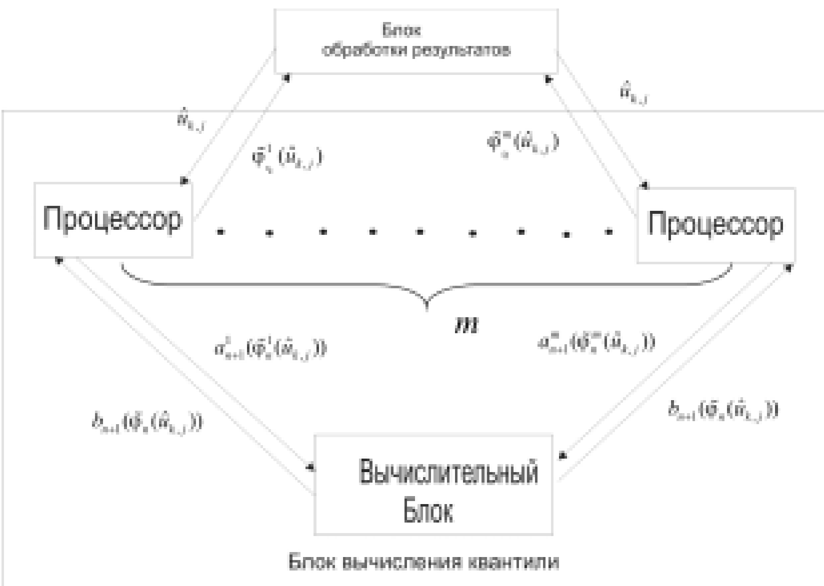

Рис. 2. Схема поиска минимума функции квантили

3. Вычислительный блок вычисляет значение индикаторной функции по усреднённой величине, характеризующей полученную информацию от всех процессоров:

$$b_{n+1}(\tilde{\varphi}_n(\hat{u}_{k,j})) \triangleq \begin{cases} 1, \text{ если } \dfrac{1}{m}\sum_{i=1}^{m} a_{n+1}^i(\tilde{\varphi}_n^i(\hat{u}_{k,j})) \le \alpha; \\ 0, \text{ иначе} \end{cases} \quad (51)$$

и передаёт её обратно каждому процессору.

4. Каждый процессор вычисляет новое значение $\tilde{\varphi}_{n+1}^i(\hat{u}_{k,j})$ по формуле

$$\tilde{\varphi}_{n+1}^i(\hat{u}_{k,j}) = \tilde{\varphi}_n^i(\hat{u}_{k,j}) + \rho_n \eta_m \left(b_{n+1}(\tilde{\varphi}_n(\hat{u}_{k,j})) - \beta\right), \quad (52)$$

где β определена в (15).

5. Каждый процессор передаёт в блок обработки информации оценку квантили $\tilde{\varphi}_{t_k}^i(\hat{u}_{k,j})$. Блок обработки информации вычисляет усреднённое значение квантили

$$\tilde{\varphi}_{t_k}(\hat{u}_{k,j}) = \frac{1}{m}\sum_{i=1}^{m} \tilde{\varphi}_{t_k}^i(\hat{u}_{k,j}).$$

6. Блок обработки информации формирует вектор

$$u_{k,j}^2 \triangleq col\left(\tilde{u}_{k1}, \ldots, u_{kj} - \delta_k, \ldots, \tilde{u}_{kr}\right),$$

где $\tilde{u}_{ki} \sim \mathbf{R}[u_{kj} - \delta_k, u_{kj} + \delta_k]$, $i = \overline{1,r}$, $i \ne j$.

После этого шаги 2—5 повторяются для $\hat{u}_{k,j} = u_{k,j}^2$.

7. Блок обработки информации вычисляет стохастический квазиградиент по формуле

$$\xi_k(u_k, \delta_k) = \frac{1}{2\delta_k}\sum_{j=1}^{r}\left[\tilde{\varphi}_{t_k}(u_{k,j}^1) - \tilde{\varphi}_{t_k}(u_{k,j}^2)\right] e_j$$

и находит новое приближение u_{k+1} к оптимальному значению u_α:

$$u_{k+1} = \begin{cases} \Pi_U(u_k - \rho_k \xi_k(u_k, \delta_k)), & \|\xi_k(u_k, \delta_k)\| \le L; \\ u_1, & \|\xi_k(u_k, \delta_k)\| > L. \end{cases} \quad (53)$$

Учитывая, что оценка квантили, полученная с помощью алгоритма (14), сходится п. н. к истинному значению квантили, а среднеквадратическое отклонение убывает с ростом числа процессоров, можно ожидать, что последовательность, образованная согласно (53), также будет сходится п.н. к минимальному значению функции квантили u_α.

Выводы

Построен алгоритм, базирующийся на использовании многопроцессорной архитектуры, который позволяет находить минимум функции квантили при достаточно общих ограничениях. Полученная методика может быть использована для решения задач минимизации с квантильным критерием качества.

Summary

A parallel algorithm is suggested to minimize a quantile function. The algorithm is based on decentralized algorithm for the quantile estimation. The almost sure convergence is proved for the algorithm.

Библиографический список

1. *Kibzun A.I., Kan Yu. S.* Stochastic programming problems with probability and quantile functions. Chichester, J. Willey and Sons, 1996.

2. *Kibzun A.I., Kuznetsov E.A.* Analysis of criteria VaR and CVaR //J. of Banking and Finance, 2006, V. 30, P. 779-796.

3. *Kibzun A.I., Kurbakovskiy V.Yu.* Guaranteeing approach to solving quantile otimization problems// Ann. Oper. Research. 1991, V. 30, P. 81-93.

4. *Кан Ю.С.* Квазиградиентый алгоритм минимизации функции квантили // Изв. РАН. Теория и системы управления. 1996. №2. С. 81-86.

5. *Кан Ю.С.* О сходимости одного стохастического квазиградиентного алгоритма квантильной оптимизации // АиТ. 2003. №2. С. 100-116.

6. *Ермольев Ю.М.* Методы стохастического программирования. — М.: Наука, 1976.

7. *Дэйвид Г.* Порядковые статистики. — М.: Наука, 1979.

8. *Benveniste A., Metivier M., Priouret P.* Adaptive Algorithms and Stochastic Approximations. Springer-Verlag, New York, NY, 1990.

9. *Kushner H.J., Yin G.G.* Stochastic Approximation and Recursive Algoritms and Applications. Springer-Verlag, New York, NY, 2003.

10. *Ширяев А.Н.* Вероятность—1. — М.: МЦНМО, 2004.

Работа поддержана грантом РФФИ №05-08-17963.

Московский авиационный институт
Статья поступила в редакцию 15.12.2007