Math-Net.Ru

Общероссийский математический портал

К. К. Ключников, Время исполнения программы в присутствии случайных сбоях, *Дискрет. матем.*, 2007, том 19, выпуск 4, 150–157

DOI: <https://doi.org/10.4213/dm984>

Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочитали и согласны с пользовательским соглашением

<http://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 3.80.253.173

26 апреля 2023 г., 18:18:55

Время исполнения программы в присутствии случайных сбоев

© 2007 г. К. К. Ключников

В данной работе рассматривается известный метод контрольных точек для восстановления работы программы при сбоях. Суть метода состоит в записи состояния программы в определенный момент времени, называемый контрольной точкой. Если сбой произойдет позже контрольной точки, то возвращаться надо не в нулевую точку, а в контрольную. Тем самым, избегается повторное исполнение программы от нулевой точки до контрольной. Предполагается, что время восстановления программы после сбоя равно нулю, а интервалы времени между двумя последовательными сбоями есть независимые случайные величины, распределенные экспоненциально. Построенная вероятностная модель позволяет получить математическое ожидание случайной величины времени исполнения программы в присутствии сбоев. Кроме того, в работе решается вопрос выбора оптимальной контрольной точки, минимизирующей математическое ожидание времени исполнения программы.

1. Время исполнения программы в присутствии сбоев

Исследования метода контрольных точек проведены в [1, 3–6]. В этих работах предполагается, что интервал времени между двумя последовательными сбоями в работе программы есть случайная величина. Случай произвольного распределения случайной величины интервала времени между сбоями рассматривается в [1]. В [5, 6] рассматриваются случаи трех различных распределений интервалов времени между сбоями, а также проводится сравнительный анализ эффективности метода контрольных точек при различных распределениях.

В данной работе рассматривается случай экспоненциального распределения интервалов времени между сбоями. Задача оптимизации заключается в выборе такого расположения контрольной точки, при котором математическое ожидание случайной величины времени исполнения программы достигает минимума. Случай экспоненциального распределения был также в подробностях изучен в [4]. Однако математическая модель, построенная в [4], не позволяет аналитически получить выражение для оптимальной контрольной точки, там представлены лишь численные результаты при различных параметрах модели.

Рассмотрим программу, время исполнения которой в отсутствие сбоев есть постоянная величина w . Если в процессе исполнения программы происходит сбой, то запуск программы происходит заново. Исполнение программы происходит до тех пор, пока сбой в процессе ее исполнения не произойдет. Пусть $(\Omega, \mathcal{A}, \mathbf{P})$ есть вероятностное пространство, где Ω есть множество элементарных исходов работы программы, $\mathcal{A}$ есть σ -алгебра подмножеств множества Ω .

Предположим, что имеется последовательность $X_1, X_2, \dots$ случайных величин, измеримых относительно σ -алгебры $\mathcal{A}$, независимых и одинаково экспоненциально распределенных с параметром λ . Мы будем интерпретировать случайные величины $X_i, i \geq 1$, как длины интервалов времени между сбоями с номерами $i - 1, i$, где $i \geq 1$ (считая нулевым сбоем начало работы программы), в случае, если бы программа работала подряд бесконечное число раз. С использованием случайных величин $X_1, X_2, \dots$ время выполнения программы T определяется следующим образом:

$$T(\omega) = \begin{cases} w, & \text{если } X_1(\omega) \geq w, \\ X_1(\omega) + w, & \text{если } X_1(\omega) < w, X_2(\omega) \geq w, \\ X_1(\omega) + X_2(\omega) + w, & \text{если } X_1(\omega) < w, X_2(\omega) < w, X_3(\omega) \geq w, \\ \dots & \end{cases} \quad (1)$$

Рассмотрим события

$$\begin{aligned} A_0 &= \{\omega: X_1(\omega) < w, X_2(\omega) < w, \dots\}, \\ A_1 &= \{\omega: X_1(\omega) \geq w\}, \\ A_2 &= \{\omega: X_1(\omega) < w, X_2(\omega) \geq w\}, \\ A_3 &= \{\omega: X_1(\omega) < w, X_2(\omega) < w, X_3(\omega) \geq w\}, \\ &\dots \end{aligned}$$

Предложение 1. *Справедливы следующие утверждения.*

(1) Множества $A_i, i \in \mathbf{N}$ попарно не пересекаются.

$$(2) \bigcup_{i=0}^{\infty} A_i = \Omega.$$

$$(3) \mathbf{P}(A_0) = 0.$$

Доказательство. Докажем утверждение 1. Возьмем любые два множества A_i и A_j , и пусть $i > j$. Тогда

$$\begin{aligned} A_i &= \{\omega: X_1(\omega) < w, X_2(\omega) < w, \dots, X_j(\omega) < w, X_{j+1}(\omega) < w, \dots, X_i(\omega) \geq w\}, \\ A_j &= \{\omega: X_1(\omega) < w, X_2(\omega) < w, \dots, X_j(\omega) \geq w\}. \end{aligned}$$

Оба множества являются подмножествами непересекающихся множеств: A_i есть подмножество множества $\{\omega: X_j(\omega) < w\}$, а A_j есть подмножество множества $\{\omega: X_j(\omega) \geq w\}$.

Докажем утверждение 2. Возьмем любое $\omega \in \Omega$. Возможны два случая.

В первом случае найдется $i \in \mathbf{N}$ такое, что $X_1(\omega) < w, X_2(\omega) < w, \dots, X_{i-1}(\omega) < w, X_i(\omega) \geq w$. Тогда $\omega \in A_i$.

Во втором случае для любого $i \in \mathbf{N}$ выполнено соотношение $X_i(\omega) < w$. В этом случае $\omega \in A_0$.

Докажем утверждение 3. По свойству непрерывности вероятностной меры,

$$\begin{aligned} \mathbf{P}(A_0) &= \mathbf{P}\left(\bigcap_{i=1}^{\infty} \{\omega: X_i(\omega) < w\}\right) \\ &= \lim_{n \rightarrow \infty} \mathbf{P}\left(\bigcap_{i=1}^n \{\omega: X_i(\omega) < w\}\right) = \lim_{n \rightarrow \infty} (1 - \exp(-\lambda w))^n = 0. \end{aligned}$$

Используя множества A_i , время исполнения программы T можно определить следующим образом:

$$T(\omega) = \begin{cases} w, & \text{если } \omega \in A_1, \\ w + \sum_{i=1}^{n-1} X_i(\omega), & \text{если } \omega \in A_n, n \geq 2. \end{cases} \quad (2)$$

Предложение 2. $T(\omega)$ есть случайная величина.

Доказательство. Для каждого $\omega \in \Omega$ величина $T(\omega)$ принимает ровно одно значение. Это следует из того, что множества A_i , $i \in \mathbf{N}$, попарно не пересекаются согласно предположению 1.

Далее, $T(\omega)$ есть измеримая функция. Действительно, для любого $x \in \mathbf{R}$

$$\begin{aligned} \{\omega: T(\omega) < x\} &= \{\omega: T(\omega) < x\} \cap \Omega = \{\omega: T(\omega) < x\} \cap \left(\bigcup_{i=0}^{\infty} A_i \right) \\ &= (\{\omega: T(\omega) < x\} \cap A_0) \cup \left(\bigcup_{i=1}^{\infty} (\{\omega: T(\omega) < x\} \cap A_i) \right). \end{aligned}$$

Заметим, что в силу определения $T(\omega)$

$$\begin{aligned} \{\omega: T(\omega) < x\} \cap A_1 &= \{\omega: w < x\}, \\ \{\omega: T(\omega) < x\} \cap A_n &= \left\{ \omega: w + \sum_{i=1}^{n-1} X_i(\omega) < x \right\}, \quad n \geq 2. \end{aligned} \quad (3)$$

В силу измеримости X_i , множества в правой части (3) будут измеримыми, а следовательно, и множество $\{\omega: T(\omega) < x\}$, как счетное объединение измеримых множеств, измеримо.

Предложение 3. Справедливо равенство

$$\mathbf{E}T = \frac{1}{\lambda}(e^{\lambda w} - 1).$$

Доказательство. Для вычисления математического ожидания случайной величины $T(\omega)$ представим ее в следующей форме:

$$\begin{aligned} T(\omega) &= wI(\omega: X_1(\omega) > w) + (X_1(\omega) + w)I(\omega: X_1(\omega) < w, X_2(\omega) \geq w) \\ &\quad + (X_1(\omega) + X_2(\omega) + w)I(\omega: X_1(\omega) < w, X_2(\omega) < w, X_3(\omega) \geq w) + \dots \end{aligned}$$

Рассмотрим следующую последовательность случайных величин:

$$\begin{aligned} T_1(\omega) &= wI(\omega: X_1(\omega) > w), \\ T_n(\omega) &= \left(\sum_{i=1}^{n-1} X_i(\omega) + w \right) I(\omega: X_1(\omega) < w, \dots, X_{n-1}(\omega) < w, X_n(\omega) \geq w), \quad n \geq 2. \end{aligned}$$

Тогда при $n \geq 2$

$$\begin{aligned}\mathbf{E}T_n(\omega) &= \mathbf{E} \left(\left(\sum_{i=1}^{n-1} X_i + w \right) I(X_1 < w, X_2 < w, \dots, X_i < w, \dots, X_{n-1} < w, X_n \geq w) \right) \\ &= (n-1) \left(\frac{1}{\lambda} (1 - e^{-\lambda w}) - w e^{-\lambda w} \right) (1 - e^{-\lambda w})^{n-2} e^{-\lambda w} + w (1 - e^{-\lambda w})^{n-1} e^{-\lambda w}\end{aligned}$$

и

$$\mathbf{E}T_1(\omega) = \mathbf{E}(wI(X_1 > w)) = w\mathbf{P}(X_1 > w) = w e^{-\lambda w}.$$

Отсюда

$$\mathbf{E}T = \mathbf{E} \sum_{n=1}^{\infty} T_n(\omega) = \sum_{n=1}^{\infty} \mathbf{E}T_n(\omega) = \frac{1}{\lambda} (e^{\lambda w} - 1).$$

Заметим, что

$$\lim_{\lambda \rightarrow 0} \mathbf{E}T = w,$$

то есть, если параметр λ близок к нулю (средний интервал между сбоями $1/\lambda$ близок к бесконечности), то $\mathbf{E}(T)$ стремится к величине времени при отсутствии сбоев w .

2. Влияние сбоев на исполнение программы

В [1, 2] описываются причины возникновения сбоев в ходе исполнения программы, а также рассматриваются методы, снижающие влияние сбоев на время работы программы. В [1] приводятся два способа снижения влияния сбоев, это перезапуск программы и способ, использующий запись состояния программы в определенный момент времени (метод контрольных точек).

В [1] рассматривается случай комбинированного использования методов обоих классов и проводится анализ численных результатов.

Процесс исполнения программы использует ресурсы вычислительной системы: оперативную память, жесткие диски. Исходный код большинства программ (в особенности это касается сложных многомодульных приложений) содержит участки, исполнение которых приводит к сбоям. Простейшим примером является программа, в исходном коде которой отсутствует процедура освобождения неиспользуемой памяти. Многократное исполнение этой программы приводит к накоплению неиспользуемых участков памяти и, в конце концов, влечет за собой ее зависание. Кроме накопления неиспользуемых участков памяти, существуют такие факторы, как увеличение количества неиспользуемых дескрипторов файлов, увеличение временных табличных пространств баз данных. В связи с этим, во многих случаях бывает полезным на некоторое время приостановить работу вычислительной системы, перезапустить ее или же провести ряд профилактических работ, способствующих обновлению ее работы.

Другим распространенным методом снижения влияния сбоев на исполнение программы является метод контрольных точек. В следующем разделе проводится исследование этого метода.

3. Метод контрольных точек

В данной работе рассматривается метод контрольных точек, идея которого состоит в следующем. Будем записывать состояние программы в одной из точек интервала $(0, w)$, назовем ее контрольной точкой. Тогда, если сбой произойдет позже момента записи состояния программы в контрольной точке, то возвращаться надо не в нулевую точку, а в контрольную. Тем самым избегается повторное исполнение программы от нулевой точки до контрольной. Если сбой происходит до момента записи контрольной точки, то исполнение программы переходит в начало. Предполагаем, что время возврата работы программы в контрольную точку равно нулю. Случай ненулевого времени возврата рассматривается в [4].

Обозначим контрольную точку через aw , $0 < a < 1$, а время исполнения программы при применении метода контрольных точек через $T_a(\omega)$. Ясно, что

$$T_a(\omega) = \begin{cases} w, & \text{если } \omega \in C, \\ w + \sum_{i=1}^{n-1} X_i(\omega), & \text{если } \omega \in B_n, n \geq 2, \\ w(1-a)w + \sum_{i=1}^{n-1} X_i(\omega), & \text{если } \omega \in A_n, n \geq 2, \end{cases} \quad (4)$$

где

$$\begin{aligned} C &= \{\omega: X_1(\omega) \geq w\}, \\ B_n &= \{\omega: X_1(\omega) < aw, \dots, X_{n-1}(\omega) < aw, X_n(\omega) \geq w\}, \quad n \geq 2, \\ A_n &= \{\omega: aw < X_1(\omega) < w, X_2(\omega) < (1-a)w, \dots, X_{n-1}(\omega) < (1-a)w, \\ &\quad X_n(\omega) \geq (1-a)w\} \cup \{\omega: X_1(\omega) < aw, aw < X_2(\omega) < w, \\ &\quad X_3(\omega) < (1-a)w, \dots, X_{n-1}(\omega) < (1-a)w, X_n(\omega) \geq (1-a)w\} \\ &\quad \cup \dots \cup \{\omega: X_1(\omega) < aw, X_2(\omega) < aw, \dots, X_{n-2}(\omega) < aw, \\ &\quad aw < X_{n-1}(\omega) < w, X_n(\omega) \geq (1-a)w\}, \quad n \geq 2. \end{aligned}$$

Положим

$$\begin{aligned} A_0 &= \{\omega: X_1(\omega) < aw, X_2(\omega) < aw, \dots\} \\ &\quad \cup \{\omega: aw < X_1(\omega) < w, X_2(\omega) < (1-a)w, X_3(\omega) < (1-a)w, \dots\} \\ &\quad \cup \left(\bigcup_{n=2}^{\infty} \{\omega: X_1(\omega) < aw, \dots, X_{n-1}(\omega) < aw, aw < X_n(\omega) < w, \right. \\ &\quad \left. X_{n+1}(\omega) < (1-a)w, X_{n+2}(\omega) < (1-a)w, \dots\} \right). \end{aligned}$$

Приведем без доказательства следующее утверждение, аналогичное предложению 1.

Предложение 4. *Справедливы следующие утверждения.*

- (1) Множества C , A_n , B_n , $n \geq 2$, попарно не пересекаются.

$$(2) A_0 \cup C \cup \left(\bigcup_{n=2}^{\infty} A_n \right) \cup \left(\bigcup_{n=2}^{\infty} B_n \right) = \Omega.$$

(3) Множества $C, A_n, B_n, n \geq 2$, измеримы.

В терминах множеств $C, A_n, B_n, n \geq 2$, случайную величину $T_a(\omega)$ можно определить следующим образом:

$$T_a(\omega) = wI(C) + \sum_{n=2}^{\infty} \left(w + \sum_{i=1}^{n-1} X_i(\omega) \right) I(B_n) + \sum_{n=2}^{\infty} \left((1-a)w + \sum_{i=1}^{n-1} X_i(\omega) \right) I(A_n). \quad (5)$$

Аналогично предложению 2 можно доказать следующее утверждение.

Предложение 5. $T_a(\omega)$ есть случайная величина.

В следующем утверждении выводится формула для математического ожидания $T_a(\omega)$.

Предложение 6. Справедливо равенство

$$\mathbf{E}T_a(\omega) = \frac{1}{\lambda}(e^{\lambda aw} - 1) + \frac{1}{\lambda}(e^{\lambda(1-a)w} - 1). \quad (6)$$

Доказательство. Нетрудно видеть, что

$$\begin{aligned} \mathbf{E}wI(C) &= we^{-\lambda w} \mathbf{E} \left(\sum_{n=2}^{\infty} \left(w + \sum_{i=1}^{n-1} X_i(\omega) \right) I(B_n) \right) \\ &= w(1 - e^{-\lambda aw})e^{-\lambda(1-a)w} + \left(\frac{1}{\lambda} - \frac{1}{\lambda}e^{-\lambda aw} - aw e^{-\lambda aw} \right) \\ &\quad \times \frac{e^{-\lambda(1-a)w}}{e^{-\lambda aw}} \mathbf{E} \left(\sum_{n=2}^{\infty} \left((1-a)w + \sum_{i=1}^{n-1} X_i(\omega) \right) I(A_n) \right) \\ &= (1-a)w(1 - e^{-\lambda(1-a)w}) \\ &\quad + \frac{1 - e^{-\lambda(1-a)w}}{e^{-\lambda aw}} \left(\frac{1}{\lambda} - \frac{1}{\lambda}e^{-\lambda aw} - aw e^{-\lambda aw} \right) \frac{e^{-\lambda(1-a)w}}{e^{-\lambda aw}}. \end{aligned}$$

В итоге,

$$\mathbf{E}T_a(\omega) = \frac{1}{\lambda}(e^{\lambda aw} - 1) + \frac{1}{\lambda}(e^{\lambda(1-a)w} - 1).$$

Заметим, что если $a = 0$ или $a = 1$, то

$$\mathbf{E}T_a(\omega) = \frac{1}{\lambda}(e^{\lambda w} - 1) = \mathbf{E}T(\omega).$$

Таким образом, если контрольная точка расположена на концах отрезка $[0, 1]$, то смысл метода контрольных точек, целью которого является минимизация среднего времени исполнения программы, теряется, и среднее время исполнения программы при применении контрольных точек становится равным среднему времени исполнения программы в отсутствие контрольных точек.

Используя результаты предложений 3 и 6, легко доказать следующее утверждение.

Теорема 1. *Справедливо равенство*

$$\operatorname{Arg} \min_{a \in [0;1]} \mathbf{E}T_a(\omega) = \frac{1}{2},$$

и при $0 \leq a \leq 1$

$$\mathbf{E}T_a(\omega) \leq \mathbf{E}T(\omega).$$

Доказательство. Положим

$$f(a) = \mathbf{E}T_a(\omega) = \frac{1}{\lambda}(e^{\lambda a w} - 1) + \frac{1}{\lambda}(e^{\lambda(1-a)w} - 1).$$

Ясно, что

$$f'(a) = w(e^{\lambda a w} - e^{\lambda(1-a)w}).$$

Поэтому, как легко видеть, $f'(a) = 0$ тогда и только тогда, когда $a = 1/2$, $f'(a) < 0$ при $a < 1/2$, и $f'(a) > 0$ при $a > 1/2$. Таким образом, функция $f(a)$ убывает на отрезке $[0, 1/2]$ и возрастает на отрезке $[1/2, 1]$, поэтому $a = 1/2$ есть точка минимума. Кроме того, ясно, что при $a \in [0, 1]$

$$\mathbf{E}T(\omega) = f(0) = f(1) \geq f(a) = \mathbf{E}T_a(\omega).$$

Данная теорема устанавливает тот факт, что оптимальный способ выбора контрольной точки в данной модели — это середина временного отрезка исполнения программы.

4. Заключение

В данной работе построена математическая модель работы программы в присутствии случайных сбоев и вычислены математические ожидания времени работы программы в присутствии и отсутствии контрольной точки. Одним из результатов данной работы является доказательство того факта, что при выборе контрольной точки в середине временного отрезка работы программы математическое ожидание времени работы достигает минимума.

В данной работе не учитывалось время, затраченное на запись состояния процесса исполнения программы в контрольной точке. Действительно, если речь идет о сложном программном приложении, то задача фиксации состояния процесса его исполнения занимает достаточно много времени. Если учитывать данное замечание, то дополненная модель, возможно, приведет к иным результатам. Вдобавок, контрольных точек может быть не одна, а несколько, и тогда оптимальный выбор контрольных точек определяется не только их расположением, но и их числом. Другой интересной задачей могла бы стать проблема поиска не только математического ожидания, но и вероятностного распределения случайной величины, равной времени исполнения программы при различном выборе контрольных точек.

Список литературы

1. Garg S., Huang Y., Kintala C., Trivedi K. S., Minimizing completion time of a program by checkpointing and rejuvenation. In: *Proc. 1996 ACM SIGMETRICS Conf.*, pp. 252–261.
2. Pfening A., Garg S., Puliafito A., Telek M., Trivedi K. S., Optimal software rejuvenation for tolerating software failures. *Performance Evaluation* (1996) **27–28**, 491–506.
3. Gelenbe E., On the optimum checkpoint interval. *J. Assoc. Comput. Mach.* (1979) **26**, 259–270.
4. Hong J., Kim S., Cho Y. Cost analysis of optimistic recovery model for forked checkpointing. In: *IEICE Trans. Inform. Syst.*, **E86-D**, № 3, pp. 1534–1541.
5. Nurmi D., Wolski R., Brevik J., *Model-based checkpoint scheduling for volatile resource environments*, UCSB Computer Sci. Techn. Rep. №2004-25.
6. Nurmi D., Wolski R., Brevik J., Minimizing the network overhead of checkpointing in cycle-harvesting cluster environments. In: *Proc. 2005 IEEE Intern. Conf. Cluster Computing*, pp. 1–10.
7. Колмогоров А. Н., Фомин С. В., *Элементы теории функций и функционального анализа*. Наука, Москва, 1976.

Статья поступила 15 июня 2007.