移动计算系统检查点迁移策略的性能评价

门朝光，徐振朋，李 香

（哈尔滨工程大学 高可信计算技术研究中心，哈尔滨 150001，menchaoguang@hrbeu.edu.cn）

摘 要：为了有效评估移动计算系统检查点迁移处理策略的性能，基于移动计算系统自身所具有的特性，给出了移动计算系统中的进程状态模型，并在此基础上提出了一种移动计算系统检查点迁移处理策略性能评价模型. 利用该性能评价模型对当前的3种日志检查点迁移策略进行了仿真实验，结果显示该模型与实际情况相吻合，从而验证了此性能评价模型的有效性. 该检查点迁移处理策略性能评价模型可用于确定不同移动计算环境下相对适用的检查点迁移处理策略.

关键词：移动计算；容错；检查点；握手迁移

中图分类号：TP302 **文献标志码**：A **文章编号**：0367－6234(2010)05－0806－05

The performance evaluation of checkpoint handoff scheme for the mobile computing system

MEN Chao-guang, XU Zhen-peng, LI Xiang

(R&D Center of High Dependability Computing Technology, Harbin Engineering University, Harbin 150001, China, menchaoguang@hrbeu.edu.cn)

Abstract: To effectively evaluate the performance of checkpoint handoff scheme for mobile computing system, a model of process state in the mobile system is presented, and then a performance evaluation model for the checkpoint handoff scheme is proposed, according to the characteristics of mobile computing system. The simulation experiments for three existing logging checkpoint recovery schemes have been implemented by the proposed performance evaluation model. The result shows that the performance evaluation model is consistent with practical cases, which proves its validity. By the proposed model of performance evaluation, the proper checkpoint handoff scheme can be determined for a specific mobile computing environment.

Key words: mobile computing; fault tolerance; checkpoint; handoff

检查点迁移处理策略是移动计算系统检查点卷回恢复策略中不可或缺的基本构成要素[1－7]. 目前移动主机检查点迁移处理策略性能评价的主要方法是使用简单数学量化或实际计算环境测试方法[8－9]. 简单数学量化法往往不能准确地反映检查点迁移策略的性能；实际计算环境中验证容错策略则需要周密复杂的系统规划和设计，不易实现. 目前，尚没有一种简单有效的检查点迁移策略性能评价方法，能够实现对移动计算系统中各检查点迁移处理策略的性能进行快速有效的评估. 为此，依据移动计算的进程状态模型，本文提出一种移动计算检查点迁移处理策略的性能评价方法.

1 移动计算系统模型

移动计算系统的系统模型可表示为 $MCS = \langle N, C \rangle$，是由节点集合 N 和信道集合 C 组成，如图1所示. 节点集合 N 包括两种主机（$N = M \cup S$），移动主机（MH）集合 $M = \{MH_1, MH_2, \cdots, MH_n\}$ 和移动支持站（MSS）集合 $S = \{MSS_1, MSS_2, \cdots, MSS_m\}$. 移动主机具有较小的处理能力和存储空间，移动支持站是静态节点，拥有较高的处理能力和可靠的存储器. 通信信道 C 由两部分

收稿日期：2008－09－25.
基金项目：国家自然科学基金资助项目(60873138).
作者简介：门朝光(1963—)，男，教授.

组成($C = W \cup W'$)，移动支持站间高速的有线通信信道 W($W = S \times S$) 和移动支持站与移动主机间相对低速的无线通信信道 W'($W' = S \times M$). MSS_i 为第 i 个移动支持站，MH_i 为第 i 个移动主机. 地理上，由一 MSS_i 覆盖的一个通信区域称作一个组(CELL)，在组 $CELL_i$ 中的移动支持站 MSS_i 被本组 MHs 称为本地移动支持站. 每个移动支持站(MSS_i) 上存在集合 $CL_i = \{MH_j \mid MH_j \in MSS_i, 0 < j < n+1\}$ 记录本组中的移动主机，子集 $Active\_MH\_List_i$ 记录活动的移动主机，子集 $Disconnected\_MH\_List_i$ 记录休眠的移动主机. 一个移动主机在某一时刻只属于一个组，即满足条件 $MH_j \in CL_i \Rightarrow MH_j \notin CL_k, \forall k \neq i$. 任一 MH_j 可以直接通过无线信道$\langle MSS_i, MH_j \rangle \in W'$ 连接到服务于该组的本地 MSS_i 上，并且通过本地的 MSS_i 与其它的 MH 或 MSS 通信.

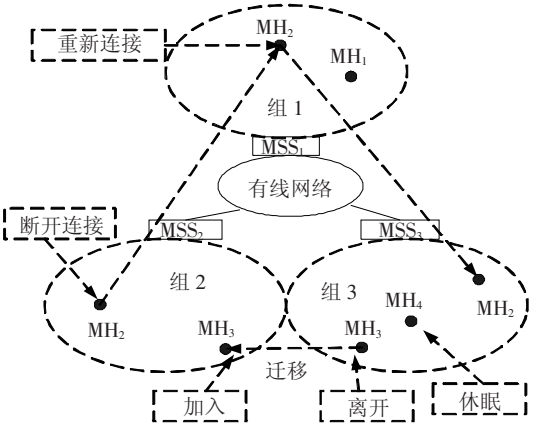

图 1　移动计算系统模型

移动计算系统包含 N 个独立计算进程 P_1，P_2，…，P_N. 每个进程周期性地存储其局部状态到稳定存储器中以产生局部检查点. 每个检查点被分配了唯一的检查点序号 CSN (checkpoint sequence number). 进程 P_k 的第 $i(i \geq 0)$ 个检查点被分配了序号 i，表示为 $C_{k,i}$. 任何存在于 $C_{k,i-1}$ 和 $C_{k,i}$ 之间的事件 $e_{k,x}$ 被称作“$e_{k,x}$ 属于 $C_{k,i}$”，表示为 $e_{k,x} \in C_{k,i}$. 故障为“fail-stop”形式，一旦进程失效，该进程将立即停止执行，不会产生任何恶意的行为. 当进程发生故障时，存储在 MH 或 MSS 内存中的内容可能会被破坏或丢失，而可靠存储器中的内容可以用来恢复主机进程. 网络连接支持双向的 FIFO 通信，且假定消息的传输是可靠的. 网络中消息的传输延迟在一定时间范围内是任意的.

2　移动计算系统检查点容错策略

移动计算检查点卷回恢复策略由 3 个基本部分构成：进程状态检查点创建、移动主机迁移处理与进程卷回恢复[10-12].

2.1　进程状态保存策略

由于移动主机存储器容量有限且不可靠，现有容错策略中都利用移动支持站上的可靠存储器存储系统进程状态. 两种最基本的进程状态保存方法是基于检查点和基于消息日志的检查点策略. 目前的移动计算检查点卷回恢复容错策略都使用基于消息日志的检查点策略.

基于消息日志的检查点容错策略同时使用了检查点和消息日志技术，进程除了在设定时刻创建检查点外，相应的消息事件也要保存到可靠存储介质上. 移动主机间不能直接通信，其接收或发送的消息需经过一个或多个移动支持站转发，因此利用移动支持站记录转发的消息不会引入过多的额外开销，且能保证日志记录的同步. 记录消息日志的方式可分为悲观、乐观和因果消息日志.

2.2　检查点迁移处理策略

由于目前检查点容错策略都是利用移动支持站的可靠存储器来存储主机进程状态和消息日志，移动主机在不同的移动支持站组间移动时，其检查点和相关消息日志会分散在不同的移动支持站上. 一旦移动主机计算进程发生故障，恢复策略必须定位其检查点和相关消息日志所在的移动支持站，即提供检查点迁移处理策略以保证故障主机及时正确地得到其全部恢复信息.

如图 1 所示，MH_1、MH_2、MH_3、MH_4 在共同执行一分布式计算程序. 某一时刻，MH_2 创建一检查点并保存在其本地移动支持站 MSS_2 上. 在随后执行过程中，MH_2 迁移到 MSS_1 组中，其后又迁移到 MSS_3 组中. 两次迁移分别发生在组 $CELL_2$ 与 $CELL_1$、$CELL_1$ 与 $CELL_3$ 的交界处. 在组 $CELL_3$ 中某时刻 MH_2 发生一计算故障. 由于 MSS_3 上未存储 MH_2 最近的检查点和消息日志. 所以检查点迁移策略必须保证进程故障主机能正确获得其检查点和相关消息日志. 目前 3 种检查点迁移处理策略为：急切(Eager)迁移策略、懒惰(Lazy)迁移策略和折衷(Trickle)迁移策略. 各检查点迁移处理策略会给移动计算系统引入不同程度的开销.

2.3　进程卷回恢复方式

移动主机计算进程发生故障后，进程重启并向本地移动支持站发送卷回恢复请求. 当移动主机从本地移动支持站获取到最新检查点后，重载进程检查点. 若采用基于消息日志的检查点卷回恢复策略，移动主机故障卷回进程将在移动支持站的协助下重放消息日志，最终恢复到故障前状态后继续运行. 恢复可分为同步恢复和异步恢复.

移动计算系统的低无线网络带宽、节点可移动、电池供给有限、移动存储空间有限及其易失等特性,使得整个移动计算系统的故障概率远远高于传统有线分布式计算系统. 因此,在容错策略中,选择高效适用的移动主机检查点迁移策略,使相应的移动计算检查点卷回恢复策略性能达到最优,对移动计算系统的可靠性和高效性尤为重要.

3 性能评价

3.1 系统进程的状态模型

在计算进程的运行过程中引入检查点操作后,计算任务的执行过程由一段段的进程检查点间隔组成. 进程的第 i 个检查点间隔由其检查点 $C_{k,i-1}$ 和 $C_{k,i}$ 之间的所有事件构成,包含第 i 个检查点 $C_{k,i-1}$,但不包含第 $i+1$ 个检查点 $C_{k,i}$. 在一个检查点间隔中,有两种意外的事件可能会发生:计算进程故障(f)和移动主机迁移(h)事件. 假定某个主机无迁移事件的情况下,其计算进程无故障执行时间为 Y. $Y^{(f,h)}$ 为在可能出现进程故障与主机迁移事件的情况下,此进程所需的执行时间. $G_Y(t)$ 为随机变量 Y 的累计分布函数,则其拉普拉斯变换为

$$\phi_Y(s) = \int_{t=0}^{\infty} \mathrm{e}^{-st}\mathrm{d}G_Y(t) = E(\mathrm{e}^{-sY}). \quad (1)$$

由式(1)可知,若 Y' 与 Y 是相互独立同分布的随机变量,则有 $\phi_{Y'}(s) = \phi_Y(s)$.

如图 2 所示,为简化求解进程检查点间隔的执行用时,区别于文献[13],采用 5 - 状态的离散马尔科夫链表示检查点间隔期间的进程状态. 状态 1 为检查点间隔开始时进程创建检查点的状态;状态 2 为计算进程正常执行状态,此状态中移动主机能处理接收消息和外部输入输出提交;状态3为移动主机迁移时的处理过程;状态4为移动主机进程发生故障后的卷回处理过程,此过程中移动主机获取最近的检查点信息,并重载检查点;状态 5 为移动主机计算进程卷回恢复过程,此过程将对该检查点间隔重新处理,重复损失的计算过程,一直恢复到计算进程故障前的正确状态. 模型中进程间消息率服从参数为 λ 的泊松分布,各移动主机计算进程发生故障概率满足参数为 γ 的泊松分布,并且移动主机连续两次迁移事件时间间隔满足参数为 ρ 的指数分布.

如果在一个进程检查点间隔执行过程中,没有出现任何进程故障或主机迁移事件,此检查点间隔会成功结束而进入下一个检查点间隔. 如果在此检查点间隔中,处在正常运行状态 2 的移动主机进程出现了迁移事件,则其计算进程会转入状态 3. 当相应的迁移处理过程结束后,进程会从状态 3 转入状态 2 继续运行. 如果在正常运行过程中有故障事件发生,则计算进程转入到状态 4. 当移动主机收集到所需的恢复信息并重载检查点完成后,进程会转入状态 5 进入恢复过程. 在状态 5 中,计算进程仍有可能发生故障转入到状态 4. 最终进程会恢复到故障前的时刻转入状态 2 继续正常运行. 在此检查点间隔计算过程成功结束时,计算进程会转入到状态 1 开始下一个检查点间隔.

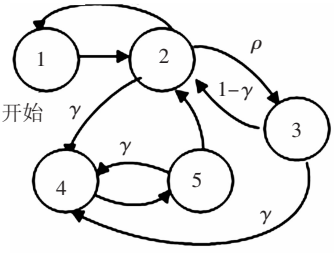

图 2 检查点间隔的马尔科夫链表示形式

计算进程在状态 4 和状态 5 的处理时间是由其故障损失的计算引起的. 在移动主机迁移处理过程中,移动支持站暂不会向其转发新的计算消息,只有当主机迁移处理过程完毕后,计算消息才被顺次转发到迁移移动主机. 本文的进程状态模型中,在移动主机迁移时,不会出现进程故障事件. 为了不失一般性,各计算进程初始时先创建一检查点(即各计算进程都从状态 1 开始).

3.2 性能评价模型

如图 2 所示,移动主机 MH 的计算进程进入到其第 N 个检查点间隔 I_N,n 为此检查点间隔内进程要处理的消息事件数,在不出现移动主机迁移和进程故障事件情况下该检查点间隔 I_N 需要的执行时间为 T_n,则 T_n 满足参数为 λ 和 n 的爱尔朗分布. 此间隔内有可能发生移动主机迁移事件,即从进程状态 2 到状态 3 的转化;H 为移动主机迁移处理过程的时间开销;R 为故障计算进程卷回时收集恢复信息并重载检查点所花费的时间开销,即是进程在状态 4 的停留时间;本文在消息传输时间上的分析基于两移动主机间距离的跳数,并假定相邻移动支持站间的跳数为 1. $T_n^{(f,h)}$ 为可能出现移动主机迁移和进程故障情况下完成该检查点间隔 I_N 所需要的时间.

X 为自检查点间隔 I_N 开始后计算进程发生故障时刻. Y 为自检查点间隔 I_N 开始后移动主机发生迁移事件的时刻. 如果 $T_n \leqslant X$ 且 $T_n \leqslant Y$,则此进程检查点间隔在执行期间没出现任何故障和移动主机迁移事件;如果 $T_n \leqslant X$ 且 $T_n > Y$,则在此

检查点间隔中移动主机出现了迁移事件，进程迁移处理时间为 $\rho T_n H$；如果 $T_n > X$，则表示在此检查点间隔中移动主机出现了故障，此时移动主机需要用时间 R 来收集相应的系统状态信息并重载检查点，然后从此检查点开始恢复到计算进程故障前状态，这意味着进程故障前的部分计算将会被重演. 在卷回恢复完成后，移动主机运行的进程转入到状态 2，继续正常执行. 假定在恢复重演过程中执行某一操作的故障概率与正常执行时的故障概率相同，则完成此检查点间隔的时间为 $R + X + X + (T_n^{(f,h)} - X)$. 由讨论可得检查点间隔 I_N 完成时间 $T_n^{(f,h)}$ 的分段函数为

$$T_n^{(f,h)}\big|_{T_n=t,X=x,Y=y} = \begin{cases} t, & t \leqslant x \& t \leqslant y; \\ t + \rho t H, & t \leqslant x \& t > y; \\ x + R + T_n^{(f,h)}, & t > x. \end{cases} \quad (2)$$

类似于文献[13]中利用拉普拉斯变换求解 $T_n^{(f,h)}$ 数学期望的方法，根据拉普拉斯变换定义式(1)，式(2)可变换为

$$\phi_{T_N^{(f,h)}}(s,n)\big|_{t,x,y} = \begin{cases} \mathrm{e}^{-st}, & t \leqslant x \& t \leqslant y; \\ \mathrm{e}^{-st}[\phi_H(s)]^{\rho t}, & t \leqslant x \& t > y; \\ \mathrm{e}^{-sx}\phi_R(s)\phi_{T_N^{(f,h)}}(s), & t > x. \end{cases} \quad (3)$$

为了求解检查点间隔 I_N 完成时间 $T_n^{(f,h)}$ 的数学期望，对式(3)中分布变量 t、x 和 y 进行积分处理为

$$\phi_{T_N^{(f,h)}}(s,n) = \int_0^\infty \int_0^\infty \int_0^\infty \phi_{T_N^{(f,h)}}(s,n)\big|_{t,x,y} P(t)P(x)P(y)\mathrm{d}y\mathrm{d}x\mathrm{d}t = \int_0^\infty \int_0^\infty \int_0^\infty \phi_{T_N^{(f,h)}}(s,n)\big|_{t,x,y} \frac{\lambda^n t^{n-1}\mathrm{e}^{-\lambda t}}{(n-1)!}\gamma \mathrm{e}^{-\gamma x}\rho \mathrm{e}^{-\rho y}\mathrm{d}y\mathrm{d}x\mathrm{d}t = \frac{\gamma\phi_H(s)\phi_{T_N^{(f,h)}}(s,n)}{(s+\gamma)}\left[1 - \frac{\lambda^n}{(s+\lambda+\gamma)^n}\right] + \frac{\lambda^n}{(s+\lambda+\gamma+\rho)^n} + \frac{\lambda^n}{(s+\lambda+\gamma-\rho\ln\phi_H(s))^n} - \frac{\lambda^n}{(s+\lambda+\gamma+\rho-\rho\ln\phi_H(s))^n}. \quad (4)$$

进一步整理式(4)可以得到此检查点间隔实际完成时间 $T_n^{(f,h)}$ 的拉普拉斯变换，即

$$\phi_{T_N^{(f,h)}}(s,n) = \frac{\dfrac{-(s+\gamma)\lambda^n}{(s+\lambda+\gamma+\rho-\rho\ln\phi_H(s))^n}}{s+\gamma-\gamma\phi_R(s)\left[1-\left(\dfrac{\lambda}{s+\lambda+\gamma}\right)^n\right]} + \frac{\left[\dfrac{(s+\gamma)\lambda^n}{(s+\lambda+\gamma+\rho)^n} + \dfrac{(s+\gamma)\lambda^n}{(s+\lambda+\gamma-\rho\ln\phi_H(s))^n}\right]}{s+\gamma-\gamma\phi_R(s)\left[1-\left(\dfrac{\lambda}{s+\lambda+\gamma}\right)^n\right]}. \quad (5)$$

利用拉普拉斯变换的性质[14]，在可能出现进程故障和移动主机迁移事件的情况下，此检查点间隔实际完成时间 $T_n^{(f,h)}$ 的数学期望为

$$E(T_n^{(f,h)}) = \frac{n\rho E(H)}{\lambda+\gamma}\left[1 - \left(1 - \frac{\rho}{\lambda+\gamma+\rho}\right)^{n+1}\right] + \left[\left(\frac{\lambda+\gamma}{\lambda}\right)^n - 1\right]\left[\frac{1}{\lambda} + E(R)\right]. \quad (6)$$

4　实例与仿真

在移动计算系统环境中，无线链路带宽为 $W' = 1$ M，有线链路带宽为 $W = 20$ M，计算消息和控制消息的大小分别为 0.5，0.1 M，$\lambda = 0.008$，$\rho = 0.005$，重载检查点的时间为 0.05 s，创建检查点时间的数学期望为 1.5 s，采用等消息间隔的检查点方式，每个检查点间隔处理 200 条消息事件. 检查点间隔 I_N 完成时间 $T_n^{(f,h)}$ 的数学期望可由式(6)得到. 为了更好地对比不同检查点迁移策略下检查点间隔 I_N 完成时间的差异，用差率 D 表示某一检查点迁移策略下 I_N 完成时间与 3 种策略下 I_N 完成时间均值之比.

结合移动计算系统的日志检查点卷回恢复策略，图 3 为不同检查点迁移策略和进程故障率情况下 I_N 完成时间差率 D 的对比情况.

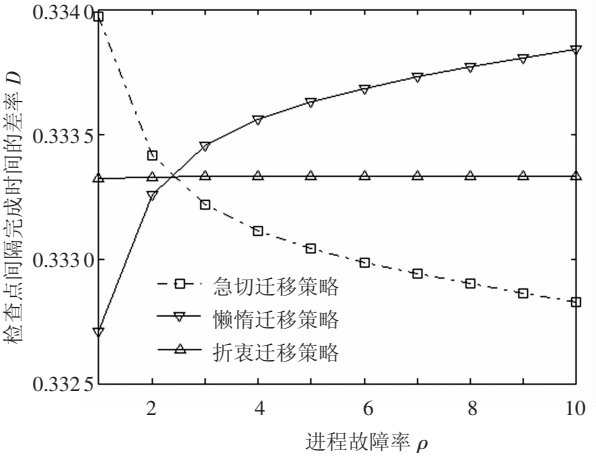

图 3　不同故障率下 3 种检查点迁移策略对比

如图 3 所示，在进程故障率 γ 较低的情况下，懒惰迁移策略具有相对较好的性能，这是由于懒惰迁移策略在迁移处理时传输的恢复信息量较小. 当进程故障率 γ 较高时，急切迁移策略则具有相对较好的性能，这是由于急切迁移策略能更好的保证故障进程恢复速度. 折衷迁移策略总能得到相对折衷的性能. 此结果正好符合了 3 种检查点迁移策略的实际情况，验证了本文性能评价方法的有效性.

在移动主机进程故障率 $\gamma = 0.001$ 的情况下，图 4 为不同检查点迁移策略和移动主机迁移

率 ρ 情况下进程检查点间隔 I_N 的完成时间差率 D. 如图 4 所示,在移动主机迁移率. ρ 较低的情况下,各检查点迁移策略的性能没有大的差别. 当移动主机迁移率 ρ 较高时,懒惰迁移策略具有相对较好的性能. 折衷迁移策略总能得到相对懒惰与急切之间折衷的性能. 此结果也正好符合了 3 种检查点迁移策略的实际情况,验证了本文性能评价方法的有效性.

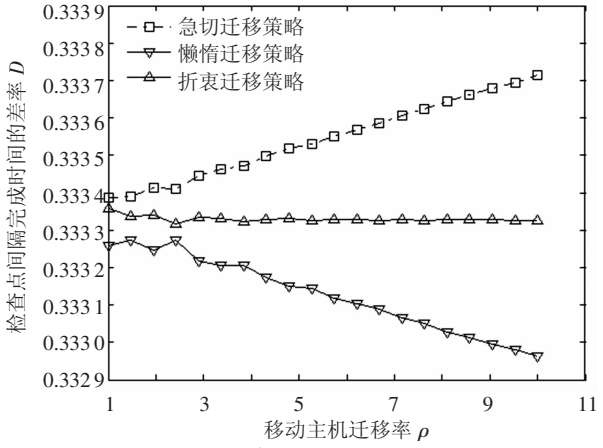

图 4 不同主机迁移率下 3 种检查点迁移策略对比

通过大量类似的仿真评估,在消息率 λ 较高、移动主机迁移率 ρ 较高和进程故障率 γ 较低的情况下,懒惰迁移处理策略会具有相对较好的性能;而在消息率 λ 较低、移动主机迁移率 ρ 较高、无线带宽较窄和进程故障率 γ 较高的情况下,折衷迁移处理策略则会具有相对较好的性能.

与具体的移动计算系统消息日志检查点卷回恢复策略相结合,利用所提出的检查点迁移策略性能评价方法,本文得到了不同移动计算环境下相对适用的检查点迁移处理策略,如表 1 所示.

表 1 适用检查点迁移处理策略表

进程消息率 λ	迁移率 ρ	进程错误率 γ	适用检查点迁移策略
高	高	低	懒惰迁移策略
高/低	低	高/低	急切迁移策略
高/低	高	高	折衷迁移策略
低	高	低	折衷迁移策略

5 结 论

1) 结合具体的日志检查点卷回恢复策略和系统参数,对各检查点迁移处理方式对系统检查点容错性能的影响进行了评估,结果符合 3 种检查点迁移策略的实际情况,从而验证了该模型的有效性.

2)利用该性能评价模型得出了不同移动计算环境下相对适用的检查点迁移处理策略.

参考文献:

[1] 杨金民, 张大方, 黎文伟. 一种可靠高效的回卷恢复实现方法[J]. 电子学报, 2006, 34(2):237 – 240.

[2] 汪东升, 邵明珑. 具有 $O(n)$ 消息复杂度的协调检查点设置算法[J]. 软件学报, 2003, 14(1)43 – 48.

[3] 李庆华, 蒋廷耀, 张红君. 一种面向移动计算的低代价透明检查点恢复协议[J]. 软件学报, 2005, 16(1):135 – 144.

[4] ELNOZAHY E N, ALVISI L, WANG Y M, *et al.* A survey of rollback-recovery protocols in message-passing systems[J]. ACM Computing Surveys, 2002, 34(3): 375 – 408.

[5] LI Guohui, WANG Hongya. A novel min-process checkpointing scheme for mobile computing systems[J]. Journal of Systems Architecture: the EUROMICRO Journal, 2005, 51(1): 45 – 61.

[6] CAO G H, SINGHAL M. Checkpointing with mutable checkpoints[J]. Theoretical Computer Science, 2003, 290(2):1127 – 1148.

[7] PARK T, WOO N, YEOM H Y. An efficient optimistic message logging scheme for recoverable mobile computing systems[J]. IEEE Transactions on Mobile Computing, 2002, 1(4):265 – 277.

[8] PRADHAN D K, KRISHNA P, VAIDAY N H. Recoverable mobile environment: design and trade-off analysis [C]//Proceedings of the The Twenty-Sixth Annual International Symposium on Fault-Tolerant Computing. Washington:IEEE Computer Society, 1996:16 – 25.

[9] CHEN I R, GU Baoshan, GEORGE S E, *et al.* On failure recoverability of client-server applications in mobile wireless environments[J]. IEEE Transactions on Reliability, 2005, 54(1):115 – 122.

[10] LI Guohui, SHU Lihchyun. Design and evaluation of a low-latency checkpointing scheme for mobile computing systems [J]. The Computer Journal,2006, 49(5): 527 – 540.

[11] KUMAR L, MISHRA M, JOSHI R C. Low overhead optimal checkpointing for mobile distributed systems [C]//The 19th International Conference on Data Engineering. [S. l.]: [s. n.], 2003: 686 – 688.

[12] HIRAKAWA T, HIGAKI H. Stable storage for wireless multi-hop access network[J]. IEIC Technical Report, 2006, 105(628):359 – 364.

[13] CHEN Xinyu, LYU M R. Performance and effectiveness analysis of checkpointing in mobile environments [C]//The Proceedings of the 22nd International Symposium on Reliable Distributed Systems. Washington: IEEE Computer Society, 2003:131 – 140.

[14] TANUSHEV M S, ARRATIA R. Central limit theorem for renewal theory for several patterns[J]. Journal of Computational Biology, 1997, 4(1): 35 – 44.

(编辑 张 红)