周报告

汇报人： 袁 娜

2017. 04. 28

本周进展

- 阅读容错经典论文

动态副本数的容错调度

•2010
•IEEE International Conference on High Performance Computing and Communications

2010 12th IEEE International Conference on High Performance Computing and Communications

Fault-Tolerant Scheduling with Dynamic Number of Replicas in Heterogeneous Systems

Laiping Zhao*, Yizhi Ren*‡, Yang Xiang†, and Kouichi Sakurai*
* Department of Informatics, Kyushu University, Fukuoka, Japan
Email: {zlp,ren}@itslab.csce.kyushu-u.ac.jp, sakurai@inf.kyushu-u.ac.jp
† School of Information Technology, Deakin University, Australia
Email: yang.xiang@deakin.edu.au
‡ School of Software, Dalian University of Technology, China

***Abstract*—In the existing studies on fault-tolerant scheduling, the active replication schema makes use of $\varepsilon + 1$ replicas for each task to tolerate ε failures. However, in this paper, we show that it does not always lead to a higher reliability with more replicas. Besides, the more replicas implies more resource consumption and higher economic cost. To address this problem, with the target to satisfy the user's reliability requirement with minimum resources, this paper proposes a new fault tolerant scheduling algorithm: *MaxRe*. In the algorithm, we incorporate the reliability analysis into the active replication schema, and exploit a dynamic number of replicas for different tasks. Both the theoretical analysis and experiments prove that the *MaxRe* algorithm's schedule can certainly satisfy user's reliability requirements. And the *MaxRe* scheduling algorithm can achieve the corresponding reliability with at most 70% fewer resources than the FTSA algorithm.**

***Index Terms*—Resource scheduling; Fault-tolerance; Reliability; Heterogeneous system**

Amazon, for example, claims that its S3 service stores three replicas of each file. That means, to store x gigabytes data, Amazon has to supply $3x$ gigabytes storage space located on three different drives, with x gigabytes corresponding to each drive. Assuming the economic cost of each drive is y, then 3 drives will cost $3y$, including extra $2y$ cost. It is believed that these extra cost will be passed on to the customers eventually. Not only the storage service, the active replication scheme in computing service also consume much extra resources and economic cost.

How to achieve a higher reliability with minimum resources is a challenge for the scheduling algorithm. In this study, specifically, *reliability* is interpreted as a probability value of the successful completion of a job. Our objective is to design a fault tolerant scheduling algorithm to satisfy the user's reliability requirement with minimum resources.

- Fault-Tolerant Scheduling with Dynamic Number of Replicas in Heterogeneous Systems

背景：云计算中各种服务的高可靠性且资源冗余最小化的需求

挑战：⑴最小化冗余→每个任务的副本数尽可能最小→如何确定每个任务的副本数？
⑵冗余最小化的同时该算法应保证用户要求的可靠性
⑶满足可靠性要求下，算法在执行时间这一性能上应是可以可接受的

模型：⑴单位时间内的故障分布 $f(k,\lambda)=\frac{\lambda^k e^{-\lambda}}{k!}$
⑵任务优先级模型 *the upward rank value*
⑶工作流模型 DAG模型

- Fault-Tolerant Scheduling with Dynamic Number of Replicas in Heterogeneous Systems

提出的调度策略：

①定义每个任务的可靠性值r，为用户需求可靠性R的的几何均值

$$r = \sqrt[n]{\Re}$$

定义执行时间为已调度任务时间与当前任务时间之和

$$TT(p_j) = ET(\tau_i, p_j) + \sum_{\tau_k \in on(p_j)} ET(\tau_k, p_j)$$

定义当前可靠性

$$\begin{aligned} & CR(p_j) \\ & = e^{-\lambda_j TT(p_j)} \\ & = e^{-\lambda_j ET(\tau_i, p_j)} \times e^{-\lambda_j \sum_{\tau_k \in on(p_j)} ET(\tau_k, p_j)} \\ & = R(\tau_i, p_j) \times \prod_{\tau_k \in on(p_j)} R(\tau_k, p_j) \end{aligned}$$

- Fault-Tolerant Scheduling with Dynamic Number of Replicas in Heterogeneous Systems

提出的调度策略：

②动态副本数

在副本数小于处理器个数条件下，不断增加副本数直至满足任务可靠性需求，得到每个任务的副本数

Algorithm 2 Decide the number of replicas for task τ_i: $\xi \leftarrow replica\_num\,(r, \tau_i, CR)$

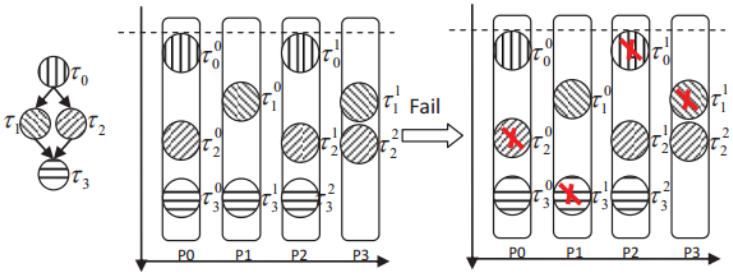
Require:
$r, \tau_i, CR.$

Ensure:
The number of replicas for task τ_i.
//Variable *counter* stores the number of replicas
1: $counter = 0;$
//Variable *fail* represents the probability of all scheduled processors failing
2: $fail = 1 - CR\,(\tau_i, p_0);$
3: **while** $(1 - fail) < r \&\& counter < m$ **do**
4: $counter = counter + 1;$
5: $fail = fail \times (1 - CR\,(\tau_i, p_{counter}));$
6: **end while**
7: return counter;

- Fault-Tolerant Scheduling with Dynamic Number of Replicas in Heterogeneous Systems

提出的调度策略：

③调度



$$\Re \leq \Psi \leq \prod_{i=1}^{i<n} (1 - \prod_{p_j \in sche(\tau_i)} (1 - R(\tau_i, p_j)))$$

Require:
$G = (V, E)$, $\Re$, and $\Lambda = \{\lambda_1, \lambda_2, \lambda_3, ..., \lambda_m\}$.
Ensure:
To what processors the tasks will be scheduled.
1: **for** each task $\tau_i \in V$ **do**
2: **for** each processor $p_j \in P$ **do**
3: $ET(\tau_i, p_j) \leftarrow$ compute the execution time using $(\tau_i.load, p_j.speed)$;
4: $R(\tau_i, p_j) \leftarrow$ compute the reliability using $(ET(\tau_i, p_j), \lambda_j)$;
5: **end for**
6: **end for**
7: $C \leftarrow$ compute the average communication time using (G, P);
8: $TP \leftarrow$ compute the priority value for all tasks using Formula 5;
9: $sort(V, TP)$; (Sort all tasks according to the task priority value)
10: $r \leftarrow root(\Re)$; (Compute the geometric mean of $\Re$ using Formula 6)
11: $\Theta = \emptyset$, $U = V$;
//Start scheduling
12: **while** $U \neq \emptyset$ **do**
13: $\tau_i = head(U)$;
14: **for** each processor $p_j \in P$ **do**
15: $TT(\tau_i, p_j) \leftarrow$ compute the total execution time using Formula 7;
16: $CR(\tau_i, p_j) \leftarrow (TT(\tau_i, p_j), \lambda_j)$; (Compute the CR for each processor using Formula 8)
17: **end for**
18: $sort(P, CR)$; (Sort all processors according to $CR(\tau_i, p_j)$)
19: $\xi \leftarrow replica\_num(r, \tau_i, CR)$; (Compute the number of replicas)
20: $S \leftarrow$ select the first ξ maximum CR value processors from sorted P;
21: Schedule task τ_i on processors in S;
22: Put τ_i into Θ;
23: $U \leftarrow U \backslash \{\tau_i\}$;
24: **end while**

与对比算法FTSA相比，该算法能有效的减少资源使用，尤其是在处理器数量多的情况下能减少70%的资源，并且能保证其性能可接受。

下一步计划

- 继续阅读基于主动复制的近期相关研究