

An IMA Static Load Balancing Strategy Optimization Method Based on Graph Theory

Qing Zhou^{1,a}, Zhijuan Zhan^{*1,b}, Siyuan Zhou^{2,c} and Shihai Wang^{2,d}

¹Science and Technology on Avionics Integration Laboratory, Avic Radio Electronics Research Institute Shanghai, China

²School of Reliability and Systems Engineering, Beihang University, China

^azhou_qing@careri.com, ^bzzjaldm@aliyun.com, ^czhousiyuan@buaa.edu.cn,

^dwangshihai@buaa.edu.cn

Keywords: integrated avionics system; reconfiguration strategy; load balance; graph theory.

Abstract: Aiming at the problem of reconfiguration strategy of integrated avionics system, a method of static load balancing strategy optimization based on graph theory is proposed after satisfying system performance and reliability requirements. This method establishes the IMA static load balancing strategy analysis model based on graph theory and puts forward the load balance evaluation indexes of IMA system: The maximum load value of the system partition obtained by allocating the maximum load value of the processor according to the time slice length; the longest communication link for the transmission of information between tasks; and the maximum traffic volume on the communication line. With the three evaluation indexes synthesizing, the unitary evaluation index of the IMA system load balancing strategy is put forward. On the basis of this method, the future research direction of load balancing of IMA system is prospected.

1. Introduction

The IMA system reconfigures the operating system and application software through refactoring. Most of the existing IMA software refactoring analytical methods are merely generating simply configuration files through a model containing configuration information and directly analyzing the schedulability and resource constraints, while lacking of an overall assessment method to IMA system availability. With reference to the analysis of load balancing in distributed system, this paper proposes an optimization method of IMA system reconfiguration strategy based on load balancing. This method establishes the IMA static load balancing strategy analysis model based on graph theory and puts forward the load balance evaluation indexes of IMA system.

2. Methodology

In IMA system, the tasks on module and partition are fixed after system reconstruction determined. Only static load balancing analysis can be performed. In this paper, we focus on the reconstruction strategy optimization of IMA system and carry out load balancing analysis on the basis of the reconfiguration scheme state space generated by exhaustive method. Therefore, it is not necessary to

find the solution of the load balancing distribution. Only according to the load balance evaluation indexes which conform to the IMA system characteristics, the load balance indexes in different reconstruction schemes are calculated and the different reconstruction schemes are compared. Based on the classic analysis method of load balance in distributed system, this paper builds up the static load balance analysis graph model of IMA system and puts forward a comprehensive evaluation index.

2.1 The IMA static load balancing strategy analysis model based on graph theory.

IMA system is divided into multiple modules; each module is divided into multiple partitions. Each partition obtains the module processor computing resource by way of time-slice rotation, and performs processing on the task. The communication efficiency of different partitions in the same module and different partitions in different modules is different, and the communication cost in the same partition is negligible. Communication efficiency within the same module and between the disparate modules is different; and the communication cost within the same partition is negligible.

Use a planar point set $P_r = \{p_{r1}, p_{r2}, \dots, p_{rk}\}$ represents k partitions within module r. Use the connection of these points $F_r = \{f_{r1}, f_{r2}, \dots, f_{rk}\} \subseteq P_r \times P_r$ represents the communication line between the nodes. Node weight $\delta: P_r \rightarrow R$ represents the module processor time slice length that the partition occupies. Assuming that within the same module, the message transfer capabilities between partitions are same and have nothing to do with the direction. Without regard to the directionality and weight of edges, the module r can be represented by the weighted graph $M_r = (P_r, F_r, \delta)$ (Fig. 1).

Use a point set $M = \{m_1, m_2, \dots, m_r\}$ represents r modules. Use the connection of these points $H = \{h_1, h_2, \dots, h_r\} \subseteq M \times M$ represents the communication line between modules. Node weight $\omega: M \rightarrow R$ represents the maximum load that the module can sustain. Assuming the message transfer capabilities between modules are same and have nothing to do with the direction. Without regard to the directionality and weight of edges, the IMA system can be represented by the weighted graph $S = (M, H, \omega)$ (Fig. 2).

The number of partitions in the IMA system is:

$$K = \sum_{i=1}^r k_i \quad (1)$$

The tasks in the IMA system are represented by the weighting graph $G = (V, E, \rho, \sigma)$ (Fig. 3). The node $V = \{v_1, v_2, \dots, v_n\}$ represents the process of tasks, Edge $E = \{e_1, e_2, \dots, e_m\} \subseteq V \times V$ represents that the data communication between tasks. The node weight $\rho: V \rightarrow R$ represents the computational resources required by the task process. The edge weight $\sigma: E \rightarrow R$ represents the traffic between tasks.

When describing a parallel application with the weighted graph, its topology can be represented by the adjacency matrix $A = (a_{ij})_{n \times n}$. Where, $1 \leq i, j \leq n, a_{ij} = 1$, indicates that there is communication between node v_i and node v_j ; $a_{ij} = 0$, indicates that there is no communication between node v_i and node v_j .

The vertex capacity of G represents the computational load of the data processing of the node, expressed by the column vector C:

$$C = \begin{Bmatrix} c_1 \\ \dots \\ c_n \end{Bmatrix} = \begin{Bmatrix} \rho(v_1) \\ \dots \\ \rho(v_n) \end{Bmatrix} \quad (2)$$

On the basis of the above description of the IMA system module and the partition and task process, the load balancing problem can be regarded as a graph embedding problem. The task of load

balancing is to find a mapping $\pi: G \rightarrow S$, which makes some of the evaluation indicators meet optimized. In this paper, we do not need to seek out the optimal mapping, just calculate the load balance of different reconstruction schemes and compare them. After the system reconfiguration, the tasks in modules and partitions in IMA system are fixed. That is, in the analysis of IMA system, graph G is static; and graph S corresponds to the system structure is considered as unchanged.

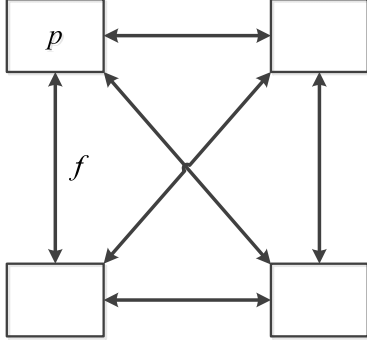


Figure 1

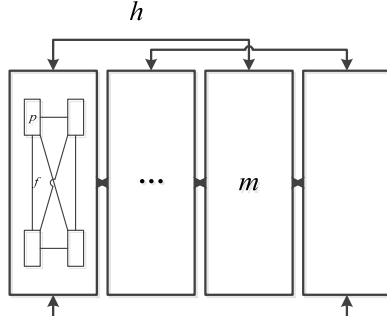


Figure 2

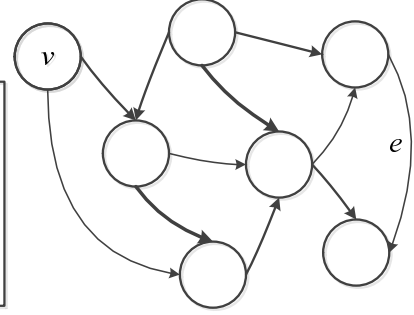


Figure 3

2.2 The load balance evaluation indexes of IMA system.

As mentioned above, the goal of the load-balancing problem is to find an optimal mapping from graph $G = (V, E, \rho, \sigma)$ to the graph $S = ((M_r = (P_r, F_r, \delta)), H, \omega)$. Suppose that mapping is $\pi: V \rightarrow M \rightarrow P_r$, the correspondence between the graph G and the graph M_r must be defined. This relation is defined by the matrix $B = (b_{ijr})_{n \times k}$.

When $b_{ijr} = 1$, The node j in G is assigned to the node i in M_r ; When $b_{ijr} = 0$, The node j in G is not assigned to the node i in M_r .

In order to evaluate the mapping $\pi: V \rightarrow P_r$, three basic indexes of IMA static load balance evaluation are proposed, which are:

Maximum load value on a single partition:

The computational cost of each node in the graph M_r is represented by a column vector:

$$X = \{x_1, x_2, \dots, x_p\}^T = BC \quad (3)$$

The maximum load on a single partition is:

$$load(\pi) = \max_{P_{ri} \in P_r, V_j \in V, \pi(V_j) = P_{ri}} \sum \rho(V_j) = \max_{1 \leq i \leq K} (x_i) \quad (4)$$

The longest communication link required for information transfer between tasks:

Suppose an edge $e = \{v_1, v_2\} \in E$ in the graph G is mapping by π . There will be $\pi(v_1) = p_{i1} \in M_1, \pi(v_2) = p_{ij} \in M_i$.

In the graph S, the links of $\pi(v_1)$ to $\pi(v_2)$ passes $m_1, m_2, \dots, m_{i-1}, m_i$, and

$$h_1 = (m_1, m_2), h_2 = (m_2, m_3), \dots, h_i = (m_{i-1}, m_i) \quad (5)$$

In the graph M_i , the links passes $p_{i1}, p_{i2}, \dots, p_{i(j-1)}, p_{ij}$, and

$$f_{i1} = (p_{i1}, p_{i2}), f_{i2} = (p_{i2}, p_{i3}), \dots, f_{ij} = (p_{i(j-1)}, p_{ij}) \quad (6)$$

Use w_e represents the pass of p_{i1} to p_{ij} , where

$$w_e = (h_1 + h_2 + \dots + h_i) + \sum_{j=1}^i (f_{i1} + f_{i2} + \dots + f_{ij}) \quad (7)$$

Suppose the length of $f_{ij}(i, j=1, 2, \dots)$ is x ; the length of $h_i(i=1, 2, \dots)$ is y ; the number of nodes in the M_i is j_i . The total length is:

$$|W_e| = yi + \sum_{i=1}^i (x j_i) \quad (8)$$

The longest communication link the graph G edge passed is:

$$long(\pi) = \max_{e=\{v_1, v_2\} \in E} |W_e| \quad (9)$$

Maximum communication line traffic:

In order to achieve a better balance of traffic on the communication network, it is necessary to calculate the maximum traffic on the communication line between the partitions

Within the same module, the maximum communication traffic on the communication line f_i is:

$$flow(f_l) = \sum_{e=\{v_1, v_2\} \in E, f_l \in w_e} \sigma(e) \quad (10)$$

The maximum communication traffic within the module is:

$$flow(f) = \max_{f=(p_1, p_2) \in F} (f_l) = \max_{f=(p_1, p_2) \in F} \left(\sum_{e=\{v_1, v_2\} \in E, f_l \in w_e} \sigma(e) \right) \quad (11)$$

Between different modules, the maximum communication traffic on the communication line h_i is:

$$flow(h_i) = \sum_{e=\{v_1, v_2\} \in E, h_i \in w_e} \sigma(e) \quad (12)$$

The maximum traffic between modules is:

$$flow(h) = \max_{h_i=(m_1, m_2) \in H} (h_i) = \max_{h_i=(m_1, m_2) \in H} \left(\sum_{e=\{v_1, v_2\} \in E, h_i \in w_e} \sigma(e) \right) \quad (13)$$

The maximum communication traffic of the system is

$$flow(\pi) = \max(flow(f), flow(h)) \quad (14)$$

For the above three basic evaluation indicators, the lower the value, the better the system load balance distribution. Calculate the weighted sum of these three indicators to obtain the IMA system load balancing strategy comprehensive evaluation index:

$$(\pi) = xload(\pi) + ylong(\pi) + zflow(\pi) \quad (15)$$

Where xyz respectively represents the weight value of the three evaluation indexes.

When $(\pi) = \min(\pi)$, the optimal scheme of load balancing assignment in IMA reconfiguration strategy is obtained.

3. Conclusions

Based on load balancing problem of distributed system, a static load balancing analysis method based on graph theory is proposed in this paper to solve the problem of IMA system reconfiguration strategy optimization. In this method, a graphical model of one-dimensional Task application graph mapping to two-dimensional hardware architecture graph is established. And load balance evaluation indexes for IMA system are proposed. Through the simple screening of possible exhaustive

reconfiguration scheme in IMA system reconfiguration strategy, it avoids the NP-complete problem of load balance solution, and obtains an effective IMA system reconfiguration strategy load balance optimization method.

In the future, aiming at the characteristics of IMA system reconfiguration by changing the system architecture to keep the software task flow unchanged, we will discuss its possible dynamic load balancing analysis method. And then, By adding important attributes of IMA systems such as security and reliability to that analytical method, a rapid and efficient IMA system reconstruction method without screening process can be construct.

Acknowledgements

This work was financially supported by the Science and Technology on Avionics Integration Laboratory and the Aeronautical Science Fund (20155551026).

References

- [1] Hendrickson B, Leland R. An improved spectral graph partition in algorithms for mapping parallel computations [J].SIAM J Scientific Computing, 1995, 16(2)
- [2] Zou C, Lu Y M, Zhang F W, Sun S L. Load-based controlling scheme of virtual machinemigration. Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference on . 2012
- [3] Yong Li, Dan Feng, Zhan Shi, Ying Zheng A probability-based load balancing algorithm for parallel file systems Journal of the Chinese Institute of Engineers, 2015, Vol.38 (6), pp.811-820
- [4] Tarek Helmy, Hamdi Al-Jamimi, Bashar Ahmed, Hamzah Loqman Fuzzy Logic-Based Scheme for Load Balancing in Grid Services Journal of Software Engineering and Applications, 2013, Vol.05 (12), pp.149-156
- [5] David R. Karger, Matthias Ruhl Simple Efficient Load-Balancing Algorithms for Peer-to-Peer Systems Theory of Computing Systems, 2006, Vol.39 (6), pp.787-804