

考虑任务不确定性的片上网络鲁棒性应用映射问题研究

王新玉^{*①} 李治莹^① 邵 帅^① 虞志刚^②

^①(东北财经大学管理科学与工程学院 大连 116025)

^②(中国电子科学研究院 北京 100041)

摘 要: 标准应用映射问题中, 每个任务的通信量是确定值, 而实际应用中任务通信具有突发性和时变特征, 因此将任务通信量建模为不确定值具有现实意义。该文利用区间流法对任务不确定性进行描述, 基于保守因子对鲁棒性应用映射问题建模, 提出了求解问题的改进禁忌搜索算法(Tabu-RAM), 通过5个Benchmark案例对本文模型和算法进行了验证。实验结果表明Tabu-RAM能够求解传统应用映射问题, 且优于现有文献中给出的算法。此外, 与传统禁忌搜索算法相比, Tabu-RAM算法在求解鲁棒性应用映射问题时具有更好的性能和稳定性。

关键词: 片上网络; 应用映射; 不确定性; 区间流; 禁忌搜索

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2019)05-1152-08

DOI: 10.11999/JEIT180600

Robust Application Mapping for Networks-on-chip Considering Uncertainty of Tasks

WANG Xinyu^① LI Zhiying^① SHAO Shuai^① YU Zhigang^②

^①(School of Management Science and Engineering, Dongbei University of Finance
and Economics, Dalian 116025, China)

^②(China Academy of Electronics and Information Technology, Beijing 100041, China)

Abstract: In the standard application mapping problem, it is assumed that the communicating traffic of a task is a fixed value. In the real applications, the communication traffic is uncertain due to the time-varying and bursty characters. Therefore, it has the practical significance modeling the task with communicating traffic uncertainty. Given the interval flow and a conservation factor, the robust application mapping problem is formulated by a min-max model, and then solved by a revised Tabu-based algorithm (Tabu-RAM) in this paper. The algorithm is verified under five benchmark instances. As the experimental results show, under the standard application scenarios, the Tabu-RAM performs better than other methods proposed in the literature. In addition, under the application scenarios with uncertain tasks, experimental results show that the Tabu-RAM performs better and more stable than the traditional tabu algorithm.

Key words: Network-on-chip; Application mapping; Uncertainty; Interval flow; Tabu search

1 引言

在“中国制造2025”和“绿色制造”战略背景下, 集成电路和芯片制造产业在性能、功耗、体积、可靠性等方面面临更大的挑战。片上网络具有良好的并行通信能力, 成为新的芯片互连设计趋势^[1,2]。

给定应用场景(核通信图)和拓扑结构, 如何将应用场景中的核资源在芯片上合理布局是芯片综合设计的关键步骤^[3,4]。标准应用映射问题定义为: 给定核通信图 $G(V, E)$ 和网络拓扑结构图 $G(T, L)$, 确定 V 中节点到 T 中节点的映射关系, 使得整个芯片的功耗(或通信)开销最小。其中 V 代表核集, E 代表通信任务集, T 代表NoC路由器集, L 代表NoC物理链路集, 一般限定 $|V| \leq |T|$ 。

针对2维Mesh网络拓扑结构, Hu等人^[5]对片上网络动态通信功耗进行分析, 建立了基于通信量和通信距离的功耗模型, 将应用映射问题建模为整数规划问题, 基于分支定界思想给出了求解方法。为了进一步提高搜索效率, 算法中引入核排序、优先

收稿日期: 2018-06-20; 改回日期: 2019-01-09; 网络出版: 2019-01-25

*通信作者: 王新玉 Distribute_2008@163.com

基金项目: 教育部人文社会科学研究一般项目(18YJC630185), 国家自然科学基金(61402086, 71501032, 71602021)

Foundation Items: The General Project of Humanities and Social Sciences Research of the Ministry of Education (18YJC630185), The National Natural Science Foundation of China (61402086, 71501032, 71602021)

级队列、对称性分析等策略。在MMS案例中，与Ad hoc方法比，该方法节省近60%的功耗。

Chou等人^[6]分析了通信资源竞争对网络性能的影响，以最小化加权通信距离和资源竞争为目标，建立双目标应用映射优化问题模型。分支定界和整数规划同属于精确求解方法的范畴，求解耗时较长。

近年来国内外学者基于进化的启发式方法求解应用映射问题。Zhang等人^[7]针对3维Mesh网络设计了两阶段映射算法，第1阶段利用聚类策略将核分配到网络的某一层上，第2阶段利用遗传算法对每一层进行映射优化，这种方法有效降低了网络的峰值功耗。Cheng等人^[8]基于贪心策略对通信核和通信任务进行排序，提出了适用于Mesh网络的构造式启发式应用映射优化算法。张大坤等人^[9]对3维NoC中任务映射进行研究，提出新的基于遗传算法的低功耗映射算法，该算法不仅降低功耗，而且大幅度减少算法的运行时间等。Wang等人^[10]分析了Mesh的对称特性，以最小化功耗为目标，提出了改进的文化基因算法。Fang等人^[11]针对树型结构的特点，利用Kernighan-Lin划分算法生成初始解，利用多种群策略避免遗传算法早熟现象。

Radu等人^[12]针对不同产品需求，分别建立单一目标(最小化功耗)和多目标(最小化功耗、热分布均衡)优化模型，提出了一组遗传算法对两类问题进行求解，引入领域知识机制提高算法的求解效率和速度。Tosun等人^[13,14]建立了片上网络功耗评估模型，以最小化网络功耗为目标建立问题模型，并对5种不同的求解机制进行了比较和分析，包括随机映射、基于模拟退火的映射、基于遗传算法的映射、基于ILP的映射以及CastNet映射方法，通过应用实例验证了CastNet的求解速度，也验证了遗传算法的有效性。Sahu等人^[15]以最小化网络功耗为目标，通过重新定义粒子的位置、速度、加法运算等，提出了改进的离散粒子群优化算法进行求解，通过引入初始解构造策略和多种群策略提高算法的性能。

针对多应用场景映射问题，Sepulveda等人^[16]设计了人工免疫算法，并引入遗传算法对所求的解方案进一步优化。Zhu等人^[17]分析了消息平均延迟的影响因素，建立了消息传输延迟模型，以最小化最差场景下的消息传送延迟为目标，提出了一种Sort-Select-Swap启发式求解方法。

在标准应用映射问题中，每个任务的通信量是确定值。通信存在突发性和时变特征，实际的通信场景是不确定的。区间流(interval flow)法对变量的不确定性建模，应用于许多经典组合优化鲁棒性

问题的变种^[18]。在给定任意的保守因子 θ 下，与传统的鲁棒性优化问题类似^[19]，本文研究最小化最差场景下通信开销的应用映射优化问题，即基于min-max的鲁棒性应用映射问题。

2 问题描述及数学模型

2.1 标准应用映射问题

给定包含 m 个核的应用，可以用一个 $m \times m$ 的矩阵 $\mathbf{F} = [f_{ij}]^{m \times m}$ 表示。假设核 i 到 j 的任务表示为 t_{ij} ， f_{ij} 代表任务 t_{ij} 的通信量。 $\mathbf{F}$ 称为一个场景，在标准应用映射问题中，场景是唯一的。

给定包含 n 个路由器的NoC平台，两点间最短通信距离可以用 $n \times n$ 矩阵 $\mathbf{D} = [d_{ij}]^{n \times n}$ 表示。前人文献^[5,8,10,14]指出，通信距离与NoC通信延迟和通信功耗成正比，低功耗低延迟的应用映射优化问题可以转化为以最小化总的开销(通信量 $\times$ 距离)为目标。

一个布局方案表示一个可行解，布局方案可用 $m \times n$ 的0-1矩阵 $\mathbf{X} = [x_{ij}]^{m \times n}$ 表示， $x_{ij} \in \{0, 1\}$ ，为1表示核 i 连到路由器 j 上，为0表示二者无连接关系。令 $\mathbf{P} = \{0, 1\}^{m \times n}$ ，标准应用映射问题(记为：AM($\mathbf{F}, \mathbf{D}$))的数学模型如式(1)–式(5)

$$\min \{ \langle \mathbf{F}, \mathbf{D}, \mathbf{X} \rangle \mid \mathbf{X} \in \mathbf{P} \} \quad (1)$$

$$\text{s.t. } \langle \mathbf{F}, \mathbf{D}, \mathbf{X} \rangle = \sum_{i=1}^m \sum_{j=1}^m \sum_{k=1}^n \sum_{l=1}^n f_{ij} d_{kl} x_{ik} x_{jl} \quad (2)$$

$$\sum_{k=1}^n x_{ik} = 1, \forall i \in \{1, 2, \dots, m\} \quad (3)$$

$$\sum_{i=1}^m x_{ik} \leq 1, \forall k \in \{1, 2, \dots, n\} \quad (4)$$

$$x_{ij} \in \{0, 1\} \quad (5)$$

其中，式(1)表示目标函数，式(2)用于计算给定解通信开销，式(3)表示一个核只能与一个路由器连接，式(4)表示一个路由器最多与一个核连接。

2.2 考虑任务不确定性的鲁棒性应用映射问题

2.2.1 鲁棒性应用映射优化的min-max模型

用 $\tilde{t}_{ij}$ 表示不确定任务，其区间流为 $[f_{ij}^-, f_{ij}^+]$ ，下界代表额定通信量，上界代表峰值通信量， f_{ij} 可能取区间内的任意值。下面对变量 f_{ij} 进行标准化。令 $\delta_{ij} = f_{ij}^+ - f_{ij}^-$ ，则 $f_{ij} = f_{ij}^- + \varepsilon_{ij} \delta_{ij}$ ， ε_{ij} 取值于 $[0, 1]$ ，称为波动因子，可以直接确定变量的取值。所有任务流区间的笛卡尔积构成该应用的完全场景集合 S 。假设不确定任务集合为 E ，令 $e = |E|$ ，给定保守因子 θ ($0 \leq \theta \leq 1$)，给定一个场景 ε ，如果其所有不确定任务波动因子之和不大于 $\theta \cdot e$ ，则该场景

属于保守因子 θ 下的可能场景。则保守因子 θ 下的场景集 U_θ 可以表示如式(6)

$$U_\theta = \left\{ \varepsilon \in S \mid \sum_{t_{ij} \in E} \varepsilon_{ij} \leq \theta \cdot e, \varepsilon_{ij} \in [0, 1] \right\} \quad (6)$$

下面给出考虑任务不确定性的鲁棒性映射问题(记为RAM($\tilde{F}, D, \theta$))的整数规划数学模型如式(7)

$$\min_{X \in P} \max_{\varepsilon \in U_\theta} \langle F(\varepsilon), D, X \rangle \quad (7)$$

鲁棒性应用映射问题的约束条件与标准应用映射问题的约束条件式(3)~式(5)一致。给定解 X , 称其在最差场景下的开销为 X 的鲁棒开销RC(X), 在额定场景下的开销为 X 的额定开销NC(X)。给定一个解 X , 确定它的最差场景是求解RAM的关键

$$RC(X) = \max_{\varepsilon \in U_\theta} \langle F(\varepsilon), D, X \rangle \quad (8)$$

$$NC(X) = \langle F(0), D, X \rangle \quad (9)$$

记 $d_{ij}(X) = \sum_{k=1}^n \sum_{l=1}^n (d_{kl} \cdot x_{ik} \cdot x_{jl})$, 将式(2)代入式(8), 计算RC(X)如式(10)

$$\begin{aligned} RC(X) &= \max_{\varepsilon \in U_\theta} \sum_{i=1}^m \sum_{j=1}^m \{f_{ij}(\varepsilon) \cdot d_{ij}(X)\} \\ &= \max_{\varepsilon \in U_\theta} \sum_{i=1}^m \sum_{j=1}^m \{(f_{ij}^- + \delta_{ij} \cdot \varepsilon_{ij}) \cdot d_{ij}(X)\} \\ &= \sum_{i=1}^m \sum_{j=1}^m \{f_{ij}^- \cdot d_{ij}(X)\} \\ &\quad + \max_{\varepsilon \in U_\theta} \sum_{i=1}^m \sum_{j=1}^m \{\delta_{ij} \cdot \varepsilon_{ij} \cdot d_{ij}(X)\} \\ &= NC(X) + DC(X) \end{aligned} \quad (10)$$

根据式(10)可知, 给定解 X , δ_{ij} 和 $d_{ij}(X)$ 是确定已知的, 确定最差场景就是在满足 $\sum_{t_{ij} \in E} \varepsilon_{ij} \leq \theta \cdot e$ 条件下确定每个任务的波动因子。

2.2.2 可行解 X 的最差场景确定

给定不确定任务 $\tilde{t}_{ij}$, 其 δ_{ij} 是确定已知的, 给定解 X , 任务 $\tilde{t}_{ij}$ 的通信距离 $d_{ij}(X)$ 也是确定已知的, 可以利用贪心策略确定解 X 在保守因子 θ 下的最差场景, 贪心依据为 $g_{ij}(X) = \delta_{ij} \cdot d_{ij}(X)$ 。

解 X 在保守因子 θ 下最差场景的确定方法如下: 将所有不确定任务按照 $g_{ij}(X)$ 降序排序, 前 $\lfloor \theta \cdot e \rfloor$ 个任务的波动因子取值为1, 第 $\lfloor \theta \cdot e \rfloor + 1$ 个任务的波动因子取值为 $\theta \cdot e - \lfloor \theta \cdot e \rfloor$, 其余任务波动因子取值为0。

确定最差场景的问题可以看作如下背包问题: 背包容量为 $\theta \cdot e$, 有 e 个待选择物品(任务), 每个物品(任务)的重量均为1, 物品(任务 $\tilde{t}_{ij}$)的价值为 $g_{ij}(X)$ 。确定最差场景的时间复杂度为 $O(e \lg e)$ 。

3 求解鲁棒性应用映射问题的启发式算法

3.1 解的表示形式

假设将 m 个核映射到 n 个路由器上, 本文提出的启发式算法中用1到 n 的排列表示解, 如表1所示。

表1 核与路由器的映射对应关系

核编号 <i>i</i>	1	2	3	4	5	6	7	8	9	10
路由器编号	3	1	5	8	7	4	10	6	9	2

给定排列 Y , 第 i 个元素为 Y_i , i 代表核编号, Y_i 代表路由器编号。如果 $i \leq m$, 表示核 i 映射到路由器 Y_i 上; 否则(即 $i > m$), 核 i 是虚拟核。

3.2 基于2-opt算子的邻域解开销计算

令 $N = \{1, 2, 3, \dots, n\}$, 利用交换操作 $\text{swap}(Y, a, b)$ 定义解 Y 的邻域空间为 $\{Z \mid Z = \text{swap}(Y, a, b), a, b \in N, a \neq b, \min(a, b) \leq m\}$ 。给定一个解 Y , 交换核 a, b ($a \neq b$)的路由器, 得到一个新的解 Z 。则 Z 可以表示如式(11)

$$Z_i = \begin{cases} Y_i, & i \neq a, i \neq b \\ Y_b, & i = a \\ Y_a, & i = b \end{cases} \quad (11)$$

给定解 Y , 用 $d_{ij}(Y)$ 代表 Y_i 和 Y_j 之间的距离, 根据式(10)可计算RC(Y)。设执行 $\text{swap}(Y, a, b)$ 得到解 Z , $\Delta_{ab} = RC(Z) - RC(Y)$ 计算如式(12)

$$\left. \begin{aligned} \Delta_{ab} &= \Delta_{ab}^1 + \Delta_{ab}^2 \\ \Delta_{ab}^1 &= NC(Z) - NC(Y) \\ \Delta_{ab}^2 &= DC(Z) - DC(Y) \end{aligned} \right\} \quad (12)$$

Δ_{ab}^1 与最差场景无关, 计算 Δ_{ab}^1 如式(13)

$$\begin{aligned} \Delta_{ab}^1 &= \langle F(0), D, Z \rangle - \langle F(0), D, Y \rangle \\ &= \sum_{i=1}^m \sum_{j=1}^m \{f_{ij}^- \cdot d_{ij}(Z)\} - \sum_{i=1}^m \sum_{j=1}^m \{f_{ij}^- \cdot d_{ij}(Y)\} \\ &= \{f_{ab}^- \cdot d_{ba}(Y) - f_{ab}^- \cdot d_{ab}(Y)\} \\ &\quad + \{f_{ba}^- \cdot d_{ab}(Y) - f_{ba}^- \cdot d_{ba}(Y)\} \\ &\quad + \sum_{i=1, i \neq a, b}^m \{f_{ai}^- \cdot d_{bi}(Y) - f_{ai}^- \cdot d_{ai}(Y)\} \\ &\quad + \sum_{i=1, i \neq a, b}^m \{f_{ia}^- \cdot d_{ib}(Y) - f_{ia}^- \cdot d_{ia}(Y)\} \\ &\quad + \sum_{i=1, i \neq a, b}^m \{f_{ib}^- \cdot d_{ia}(Y) - f_{ib}^- \cdot d_{ib}(Y)\} \\ &\quad + \sum_{i=1, i \neq a, b}^m \{f_{bi}^- \cdot d_{ai}(Y) - f_{bi}^- \cdot d_{bi}(Y)\} \end{aligned} \quad (13)$$

Δ_{ab}^2 与最差场景有关, 计算如式(14)

$$\begin{aligned}
\Delta_{ab}^2 &= DC(\mathbf{Z}) - DC(\mathbf{Y}) \\
&= \max_{\varepsilon \in U_\theta} \sum_{i=1}^m \sum_{j=1}^m \{g_{ij}(\mathbf{Z}) \cdot \varepsilon_{ij}\} \\
&\quad - \max_{\varepsilon \in U_\theta} \sum_{i=1}^m \sum_{j=1}^m \{g_{ij}(\mathbf{Y}) \cdot \varepsilon_{ij}\} \quad (14)
\end{aligned}$$

式(14)中, $g_{ij}(\mathbf{Z})$ 计算如式(15)

$$g_{ij}(\mathbf{Z}) = \begin{cases} g_{ij}(\mathbf{Y}), & i \neq a, b, j \neq a, b \\ \delta_{ab} \cdot d_{ba}(\mathbf{Y}), & i = a, j = b \\ \delta_{ba} \cdot d_{ab}(\mathbf{Y}), & i = b, j = a \\ \delta_{aj} \cdot d_{bj}(\mathbf{Y}), & i = a, j \neq a, b \\ \delta_{bj} \cdot d_{aj}(\mathbf{Y}), & i = b, j \neq a, b \\ \delta_{ia} \cdot d_{ib}(\mathbf{Y}), & i \neq a, b, j = a \\ \delta_{ib} \cdot d_{ia}(\mathbf{Y}), & i \neq a, b, j = b \end{cases} \quad (15)$$

对于 $g_{ij}(\mathbf{Z})$, 有 $4m - 4$ 项需要计算, 根据2.2.2节的贪心方法计算出最差场景, 可计算出 Δ_{ab}^2 。 $\Delta_{ab} = \Delta_{ab}^1 + \Delta_{ab}^2$, 若 $\Delta_{ab} < 0$, 解 $\mathbf{Z}$ 优于解 $\mathbf{Y}$ 。若由式(10)计算解 $\mathbf{Z}$ 的开销, $NC(\mathbf{Z})$ 的计算需要执行 m^2 次乘法和加法, 而利用式(13)计算 $NC(\mathbf{Z})$ 只需要执行 $8m - 12$ 次乘法和加法; 类似地, 利用式(10)计算 $DC(\mathbf{Z})$, 最差场景的确定计算 m^2 次乘法得到所有的 $g_{ij}(\mathbf{Z})$, 对 m^2 个元素排序。利用式(15)确定最差场景只需要执行 $4m - 4$ 次乘法得到所有的 $g_{ij}(\mathbf{Z})$, 对 $4m - 4$ 个元素排序后再与 $m^2 - 4m + 4$ 个有序元素合并排序。当问题规模较大时, 本节中计算 $RC(\mathbf{Z})$ 的方法可以减少时间开销。

3.3 求解鲁棒性应用映射问题的Tabu-RAM算法

本节给出求解鲁棒性应用映射问题的改进禁忌搜索(Tabu for Robust Application Mapping, Tabu-RAM)算法。与传统的Tabu算法^[20]相比, Tabu-RAM给出了一种新的邻域解开销计算方法, 提出了基于贪心策略的初始解设计方法和基于优先级的藐视规则, 有效提高了算法搜索效率。

3.3.1 禁忌表设计

本文采用如下方式设计禁忌表, 表由 n 行 n 列矩阵组成, 元素 $\text{tabulist}[i][j]$ 代表是否允许将核 i 交换到路由器 j 上, 为了避免迂回搜索, 给定解 $\mathbf{Y}$, 对于 $\text{swap}(\mathbf{Y}, a, b)$, 为简便起见, 假设 $a \leq b$, 利用表2所示过程判定 $\text{swap}(\mathbf{Y}, a, b)$ 是否被禁止。

禁忌表初始化为false, 一旦将核 i 交换到路由器 j 上, 禁忌表元素 $\text{tabulist}[i][j]$ 将会在接下来的 λ 代内为true, 之后解除禁忌。 λ 为禁忌长度, 根据文献^[21,22]可知, λ 从 $[0.9n, 1.1n]$ 中取随机值时效果最好。

3.3.2 藐视准则

禁忌搜索中通常采用藐视准则赦免被禁忌的优

表2 swap($\mathbf{Y}, a, b$) 计算过程

步骤1	令 $\text{aFlag} = \text{true}$, $\text{bFlag} = \text{true}$;
步骤2	易知 $a \leq m$, 若 $\text{tabulist}[a][\mathbf{Y}_b]$ 为真, 代表禁止将核 a 放置到路由器 $\mathbf{Y}_b$ 上, $\text{aFlag} = \text{false}$;
步骤3	当 $b \leq m$ 时, 若 $\text{tabulist}[b][\mathbf{Y}_a]$ 为真, 代表禁止将核 b 放置到路由器 $\mathbf{Y}_a$ 上, $\text{bFlag} = \text{false}$; 当 $b > m$ 时, b 是虚拟核, 若存在某个 $\text{tabulist}[c][\mathbf{Y}_a]$ ($c \geq b$) 为真, 禁止将路由器 $\mathbf{Y}_a$ 置为空, $\text{bFlag} = \text{false}$;
步骤4	若 aFlag 和 bFlag 均为false, $\text{swap}(\mathbf{Y}, a, b)$ 交换被禁止; 否则, 交换不被禁止, 对应的解作为候选解。

良状态, 实现全局优化。给定解 $\mathbf{Y}$, 对于 $\text{swap}(\mathbf{Y}, a, b)$, 若交换后的解 $\mathbf{Z}$ 优于目前搜索到的全局最优解 $\mathbf{G}$, 则 $\text{swap}(\mathbf{Y}, a, b)$ 通过藐视准则。

将邻域空间中通过3.3.1中禁忌检验的解构成的集合记为 T , 将通过藐视准则的解集合记为 A , 则 $T \cup A$ 称为候选解集。给定候选解集, A 集合中的解优先级高, 而 T 集合中的解优先级低。原因如下: 根据藐视准则, A 集合中包含非常优秀的解。在同一优先级的解, 选择最好的作为下次迭代的初始解。

3.3.3 初始解构造

给定保守因子 θ , 确定临时最差场景用于初始解的构造。将不确定任务按照其 f^+ 从大到小排序, 将排在前 $\lfloor \theta \cdot e \rfloor$ 个任务的波动因子取值为1, 将第 $\lfloor \theta \cdot e \rfloor + 1$ 个任务波动因子取值为 $\theta \cdot e - \lfloor \theta \cdot e \rfloor$, 剩下任务波动因子取值为0, 该场景记为临时最差场景 ω 。

已映射核集合 M 初始化为空, 未映射核集合 U 初始化为所有待映射的核 V 。首先从 ω 中选择通信量最大的任务, 设为 (c_1, c_2) , 为核 c_1 在拓扑结构中选择一路由器随机位置, 并将核 c_1 加入到 M 中。

(1) 核选择: 令 $U = V - M$, 选择跨越 U 和 M 集合的通信量最大的任务, 记为 (c_3, c_4) , 必定存在 $c_3 \in M$ 且 $c_4 \in U$ (或 $c_4 \in M$ 且 $c_3 \in U$)。

(2) 路由器选择: 将来自 U 的核加入到 M , 将其映射到空闲路由器上使得 M 中增加的通信开销最小。当有多个可选位置时, 随机选择一个。

重复上述核选择和路由器选择过程, 直到所有核都确定其连接的路由器位置为止。

3.3.4 Tabu-RAM算法流程

为了保证搜索的多样性, 采用变起点的禁忌搜索框架, 将一次Tabu搜索的迭代次数设置为 n (n 为问题规模), 重复调用禁忌搜索过程, 若连续5次调用全局最优解未更新, 利用初始解构造策略更换搜索起点。本文中将最大搜索时间作为迭代终止条件, 具体算法流程如表3所示。

表3 Tabu-RAM算法流程

步骤1	根据3.3.3节生成初始解 Y ，全局最优解 $G = Y$ ，连续未更新次数 $NIN=0$ ；
步骤2	对 Y 进行Tabu搜索，迭代次数为 n ，根据需要更新 Y 和 G ，若达到最大搜索时间，转步骤5；
步骤3	若 G 未更新， $NIN++$ ；否则 $NIN=0$ ；
步骤4	若 $NIN \geq 5$ ，利用3.3.3中方法构造解赋值给 Y ，转步骤2；否则，直接转步骤2；
步骤5	迭代终止，返回 G 。

4 实验结果

采用Java语言实现本文算法，在Intel E5-1620 v3 3.5 GHz工作站上运行。由于启发式方法存在随机性，给出10次运行中的最优解的开销和平均开销来分析算法的求解质量和稳定性。基于5个常用Benchmark案例^[5,10,11,15]进行验证，将文献中的确定性场景作为标准化场景。本文以标准应用任务的通信量 f 为均值，为每个任务生成随机区间 δ ，将 $[f - \delta/2, f + \delta/2]$ 作为其区间流。

4.1 确定应用场景下的结果比较

确定应用场景可以看作是不确定应用场景的特例。将每个任务的 δ 区间设为0，对Tabu-RAM进行验证。如表4所示，与CastNet, GA和PSO算法相比，本文算法得到的解最好。

4.2 不确定场景下的结果比较

针对不确定场景，表5中给出了10次运行结果的最优值和平均值。可以看出，MC方法的效果最差，当问题规模较小(≤ 16)时，传统的Tabu方法与本文Tabu-RAM算法相当，本文算法略优于传统的Tabu算法；当问题规模增加时，二者差距增加，以DVOPD映射到 4×8 的Mesh为例，当 θ 取值为0.6时，Tabu-RAM算法将鲁棒开销降低了5.74%。

4.3 算法稳定性比较

稳定性是评价启发式算法优劣的另一个重要指标。表5给出了Tabu-RAM和Tabu方法的最优值和平均值。将10次运行中的最优解和平均解之间的开销差距定义为：开销差距=(平均值-最优值)/最优值，并将其作为评价算法稳定性的指标，如表5所示。不同场景下Tabu-RAM的开销差距低于Tabu的开销差距，Tabu-RAM的稳定性较好。图1和图2分别给出测试案例4和5下当 θ 取值为0.6和0.8时10次运行的鲁棒性开销的结果，可以看出Tabu-RAM算法的稳定性较好。

5 结论

传统应用映射优化过程中未考虑任务的不确定性。本文将任务的不确定性考虑在内，利用区间流法对不确定任务进行描述，当给定保守因子的情况

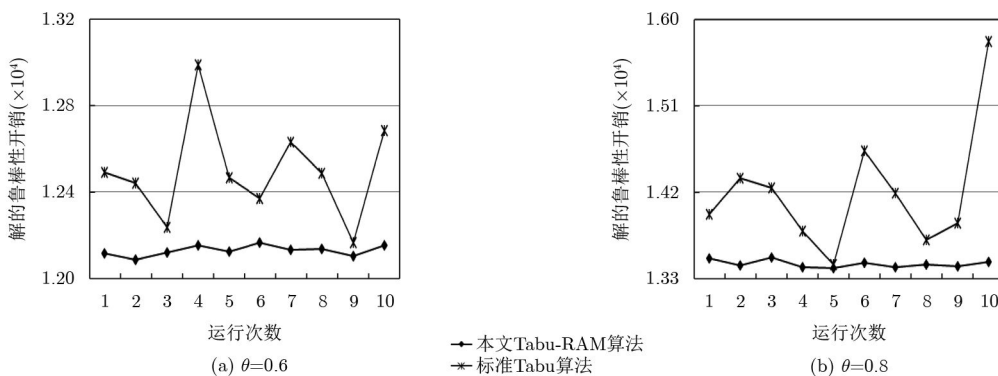
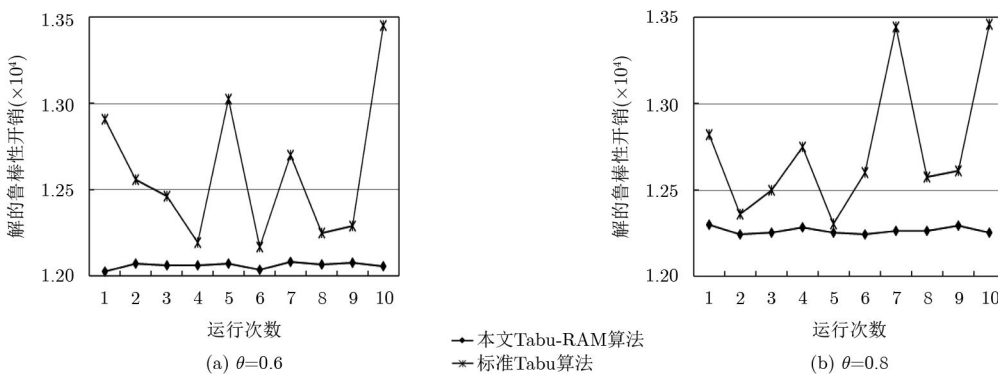
图1 测试算例4映射到 4×8 Mesh网络中10次结果比较图2 测试算例5映射到 6×6 Mesh网络中10次结果比较

表4 确定应用场景下本文算法与已有文献中的算法

编号	测试用例	核数	映射Mesh结构	CastNet ^[13]	GA ^[14]	PSO ^[15]	本文算法
1	MPEG-4	12	4×4	3852	3567*	3567*	3567*
2	VOPD	16	4×4	4135	4290	4119*	4119*
3	MMS	25	5×5	689503	689713	-	688297*
4	DVOPD	32	4×8	-	-	9602	9570*
5	DVOPD	32	6×6	9618	10006	-	9522*

表5 不确定应用场景下不同算法比较

编号	测试用例		本文Tabu-RAM算法			标准Tabu算法			文献 ^[17] 中MC算法
	名称	θ 取值	最优值	平均值	开销差距(%)	最优值	平均值	开销差距(%)	最优值
1	MPEG-4 4×4 Mesh	0	42328.00	42328.00	0	42328.00	43270.20	2.23	49962.00
		0.2	77628.00	77628.00	0	77628.00	79150.96	1.96	86986.80
		0.4	92888.00	92888.00	0	92888.00	96519.68	3.91	99176.40
		0.6	98359.40	98359.40	0	98359.40	101965.82	3.67	118958.80
		0.8	99924.60	99924.60	0	99924.60	106761.86	6.84	116436.80
		1.0	99993.00	99993.00	0	99993.00	103121.00	3.13	113627.00
		平均值			0			3.62	
2	VOPD 4×4 Mesh	0	2147.00	2147.00	0	2147.00	2148.60	0.07	2444.00
		0.2	4566.60	4567.00	0.01	4570.60	4573.00	0.05	5761.40
		0.4	5530.60	5530.60	0	5530.60	5534.36	0.07	7168.00
		0.6	5818.20	5818.20	0	5818.20	5820.40	0.04	7294.00
		0.8	6004.40	6004.40	0	6004.40	6018.66	0.24	7765.80
		1.0	6070.00	6070.00	0	6070.00	6092.90	0.38	7858.00
		平均值			0			0.14	
3	MMS 5×5 Mesh	0	411649.00	411750.50	0.02	412039.00	416316.10	1.04	622005.00
		0.2	786490.40	786640.70	0.02	787536.40	820504.72	4.10	1254921.60
		0.4	917007.00	917152.10	0.02	917396.00	940330.06	2.50	1421245.80
		0.6	952629.00	952869.40	0.03	953018.00	982380.46	3.08	1379116.40
		0.8	959803.40	960015.00	0.02	960168.80	980107.78	2.08	1436050.60
		1.0	960575.00	960846.40	0.03	961210.00	995897.40	3.61	1558999.00
		平均值			0.02			2.75	
4	DVOPD 4×8 Mesh	0	5593.00	5606.80	0.25	5726.00	5871.70	2.54	11706.00
		0.2	10277.00	10317.82	0.40	10315.00	11101.32	7.62	22954.40
		0.4	12083.80	12126.82	0.36	12161.80	12493.80	2.73	25964.00
		0.6	12974.40	13011.08	0.28	13145.40	13803.52	5.01	29696.00
		0.8	13413.40	13452.02	0.29	13447.40	14197.74	5.58	32243.40
		1.0	13527.00	13591.80	0.48	13845.00	14281.30	3.15	29641.00
		平均值			0.34			4.44	
5	DVOPD 6×6 Mesh	0	5565.00	5573.70	0.16	5710.00	5853.60	2.51	12535.00
		0.2	10236.00	10273.40	0.37	10276.00	10985.50	6.90	24139.00
		0.4	12024.40	12060.12	0.30	12167.40	12601.10	3.56	27665.80
		0.6	12885.40	12905.66	0.16	12956.40	13489.16	4.11	29484.00
		0.8	13292.00	13319.40	0.21	13366.60	13892.86	3.94	28463.20
		1.0	13439.00	13493.30	0.40	13716.00	14250.20	3.89	31575.00
		平均值			0.26			4.15	

下, 利用min-max模型对鲁棒性应用映射问题进行建模, 并提出求解鲁棒性应用映射问题的改进禁忌搜索算法Tabu-RAM。实验结果表明, 与现有文献中求解方法相比, 改进算法Tabu-RAM在求解标准应用映射问题时效果最好; 在求解鲁棒性应用映射问题时本文Tabu-RAM算法的求解质量和稳定性均优于传统的Tabu算法。

参 考 文 献

- [1] 李刚, 汪鹏君, 张跃军, 等. 基于65 nm工艺的多端口可配置PUF电路设计[J]. 电子与信息学报, 2016, 38(6): 1541–1546. doi: [10.11999/JEIT150968](https://doi.org/10.11999/JEIT150968).
- LI Gang, WANG Pengjun, ZHANG Yuejun, *et al.* Design of multi-port configurable PUF circuit based on 65 nm technology[J]. *Journal of Electronics & Information Technology*, 2016, 38(6): 1541–1546. doi: [10.11999/JEIT150968](https://doi.org/10.11999/JEIT150968).
- [2] SU Nan, GU Huaxi, WANG Kun, *et al.* A highly efficient dynamic router for application-oriented network on chip[J]. *The Journal of Supercomputing*, 2018, 74(7): 2905–2915. doi: [10.1007/s11227-018-2334-5](https://doi.org/10.1007/s11227-018-2334-5).
- [3] ABDELFATTAH M S, BITAR A, and BETZ V. Design and applications for embedded networks-on-chip on FPGAs[J]. *IEEE Transactions on Computers*, 2017, 66(6): 1008–1021. doi: [10.1109/TC.2016.2621045](https://doi.org/10.1109/TC.2016.2621045).
- [4] POPOOLA O and PRANGGONO B. On energy consumption of switch-centric data center networks[J]. *The Journal of Supercomputing*, 2018, 74(1): 334–369. doi: [10.1007/s11227-017-2132-5](https://doi.org/10.1007/s11227-017-2132-5).
- [5] HU Jingcao and MARCULESCU R. Energy- and performance-aware mapping for regular NoC architectures[J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2005, 24(4): 551–562. doi: [10.1109/TCAD.2005.844106](https://doi.org/10.1109/TCAD.2005.844106).
- [6] CHOU C L and MARCULESCU R. Contention-aware application mapping for network-on-chip communication architectures[C]. Proceedings of 2008 IEEE International Conference on Computer Design, Lake Tahoe, USA, 2008: 164–169. doi: [10.1109/ICCD.2008.4751856](https://doi.org/10.1109/ICCD.2008.4751856).
- [7] ZHANG Bixia, GU Huaxi, YANG Yintang, *et al.* Thermal and competition aware mapping for 3D network-on-chip[J]. *IEICE Electronics Express*, 2012, 9(19): 1510–1515. doi: [10.1587/elex.9.1510](https://doi.org/10.1587/elex.9.1510).
- [8] CHENG C and CHEN Weimei. Application mapping onto mesh-based network-on-chip using constructive heuristic algorithms[J]. *The Journal of Supercomputing*, 2016, 72(11): 4365–4378. doi: [10.1007/s11227-016-1746-3](https://doi.org/10.1007/s11227-016-1746-3).
- [9] 张大坤, 宋国治, 林华洲, 等. 二次改进遗传算法与3D NoC低功耗映射[J]. 计算机研究与发展, 2016, 53(4): 921–931. doi: [10.7544/jssn1000-1239.2016.20150682](https://doi.org/10.7544/jssn1000-1239.2016.20150682).
- ZHANG Dakun, SONG Guozhi, LIN Huazhou, *et al.* Double improved genetic algorithm and low power task mapping in 3D networks-on-chip[J]. *Journal of Computer Research and Development*, 2016, 53(4): 921–931. doi: [10.7544/jssn1000-1239.2016.20150682](https://doi.org/10.7544/jssn1000-1239.2016.20150682).
- [10] WANG Xinyu, LIU Haikuo, and YU Zhigang. A novel heuristic algorithm for IP block mapping onto mesh-based networks-on-chip[J]. *The Journal of Supercomputing*, 2016, 72(5): 2035–2058. doi: [10.1007/s11227-016-1719-6](https://doi.org/10.1007/s11227-016-1719-6).
- [11] FANG Juan, YU Lu, LIU Sitong, *et al.* KL_GA: An application mapping algorithm for Mesh-of-Tree (MoT) architecture in network-on-chip design[J]. *The Journal of Supercomputing*, 2015, 71(11): 4056–4071. doi: [10.1007/s11227-015-1504-y](https://doi.org/10.1007/s11227-015-1504-y).
- [12] RADU C, MAHBUB M S, and VINTAN L. Developing domain-knowledge evolutionary algorithms for network-on-chip application mapping[J]. *Microprocessors and Microsystems*, 2013, 37(1): 65–78. doi: [10.1016/j.micpro.2012.11.003](https://doi.org/10.1016/j.micpro.2012.11.003).
- [13] TOSUN S. New heuristic algorithms for energy aware application mapping and routing on mesh-based NoCs[J]. *Journal of Systems Architecture*, 2011, 57(1): 69–78. doi: [10.1016/j.sysarc.2010.10.001](https://doi.org/10.1016/j.sysarc.2010.10.001).
- [14] TOSUN S, OZTURK O, OZKAN E, *et al.* Application mapping algorithms for mesh-based network-on-chip architectures[J]. *The Journal of Supercomputing*, 2015, 71(3): 995–1017. doi: [10.1007/s11227-014-1348-x](https://doi.org/10.1007/s11227-014-1348-x).
- [15] SAHU P K, SHAH T, MANNA K, *et al.* Application mapping onto mesh-based network-on-chip using discrete particle swarm optimization[J]. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2014, 22(2): 300–312. doi: [10.1109/TVLSI.2013.2240708](https://doi.org/10.1109/TVLSI.2013.2240708).
- [16] SEPÚLVEDA M J, CHAU W J, GOGNIAT G, *et al.* A multi-objective adaptive immune algorithm for multi-application NoC mapping[J]. *Analog Integrated Circuits and Signal Processing*, 2012, 73(3): 851–860. doi: [10.1007/s10470-012-9869-9](https://doi.org/10.1007/s10470-012-9869-9).
- [17] ZHU Di, CHEN Lizhong, YUE Siyu, *et al.* Balancing on-chip network latency in multi-application mapping for chip-multiprocessors[C]. Proceedings of the 2014 IEEE 28th International Parallel and Distributed Processing Symposium, Phoenix, USA, 2014: 872–881. doi: [10.1109/IPDPS.2014.94](https://doi.org/10.1109/IPDPS.2014.94).
- [18] ASSUNÇÃO L, NORONHA T F, SANTOS A C, *et al.* A linear programming based heuristic framework for min-max regret combinatorial optimization problems with interval

- costs[J]. *Computers & Operations Research*, 2017, 81: 51–66. doi: [10.1016/j.cor.2016.12.010](https://doi.org/10.1016/j.cor.2016.12.010).
- [19] LÓPEZ J, POZO D, CONTRERAS J, *et al.* A multiobjective minimax regret robust VAr planning model[J]. *IEEE Transactions on Power Systems*, 2017, 32(3): 1761–1771. doi: [10.1109/TPWRS.2016.2613544](https://doi.org/10.1109/TPWRS.2016.2613544).
- [20] FEIZOLLAHI M J and FEYZOLLAHI H. Robust quadratic assignment problem with budgeted uncertain flows[J]. *Operations Research Perspectives*, 2015, 2: 114–123. doi: [10.1016/j.orp.2015.06.001](https://doi.org/10.1016/j.orp.2015.06.001).
- [21] BENLIC U and HAO Jinkao. Memetic search for the quadratic assignment problem[J]. *Expert Systems with Applications*, 2015, 42(1): 584–595. doi: [10.1016/j.eswa.2014.08.011](https://doi.org/10.1016/j.eswa.2014.08.011).
- [22] DOKEROGLU T and MENGUSOGLU E. A self-adaptive and stagnation-aware breakout local search algorithm on the grid for the Steiner tree problem with revenue, budget and hop constraints[J]. *Soft Computing*, 2018, 22(12): 4133–4151. doi: [10.1007/s00500-017-2630-7](https://doi.org/10.1007/s00500-017-2630-7).
- 王新玉：女，1985年生，博士，副教授，研究方向为并行分布式计算、智能优化等。
- 李治莹：女，1997年生，本科生，研究方向为智能优化算法等。
- 邵 帅：男，1994年生，硕士生，研究方向为应用映射算法等。
- 虞志刚：男，1988年生，博士，研究方向为片上网络路由器设计等。