以边为中心的密码逻辑阵列高能效映射算法

徐金甫 章宇雷* 李 伟 陈 韬

(战略支援部队信息工程大学 郑州 450001)

摘 要：为解决密码算法在粗粒度可重构密码逻辑阵列(CRCLA)上映射性能不高及编译时间长的问题，该文提出一种密码算法和硬件资源的描述形式，在映射过程中能够更加直观地显示各个资源的占用情况；并通过分析密码算法运算特征与粗粒度可重构密码逻辑阵列硬件结构的内在关联，以减少关键路径延时为目标，提出了一种以边为中心的密码逻辑阵列高能效映射算法(ECLMap)。通过边映射来指导节点映射，结合相关映射策略，引入回溯机制来提高映射成功率。在仿真平台下对多种密码算法进行实验，相比于其他通用的映射算法，结果表明该文提出的算法映射性能最佳，在算法能效上平均提升了约20%，同时在编译时间上平均提升了约25%。实现了算法的高能效映射。

关键词：密码算法；阵列；映射；以边为中心；能效

中图分类号：TN918.2 **文献标识码**：A **文章编号**：1009-5896(2021)06-1587-09

DOI: 10.11999/JEIT210008

A Edge-Centered High Energy-Efficient Mapping Algorithm for Cipher Logic Array

XU Jinfu ZHANG Yulei LI Wei CHEN Tao

(*Strtegic Support Force Information Engineering University*, *Zhengzhou* 450001, *China*)

Abstract: In order to solve the problem of low mapping performance and long compilation time of cipher algorithms on Coarse-grained Reconfigurable Cipher Logic Arrays (CRCLA), a description of cryptographic algorithms and hardware resources is proposed, which can display the occupancy of each resource more intuitively during the mapping process. Then by analyzing the inherent relationship between the cryptographic algorithm operation characteristics and the coarse-grained reconfigurable cipher logic array hardware structure, with the goal of reducing the critical path delay, an Edge - centric Cipher Logical array Mapping (ECLMap) algorithm is proposed. Using edge mappings to guide the mapping of nodes, combined with relevant mapping strategy, the backtracking mechanism is introduced to improve the success rate of mapping. Compared with other common mapping algorithms, the results show that the algorithm proposed in this paper has the best mapping performance, with an average increase of about 20% in the algorithm energy efficiency and about 25% in the compilation time. The high-efficiency mapping of the algorithm is realized.

Key words: Cryptographic algorithm; Array; Mapping; Edge centered; Energy-efficient

1 引言

粗粒度可重构密码逻辑阵列(Coarse -grained Reconfigurable Cipher Logical Array, CRCLA)是一种基于数据流的高效运算结构，它结合了专用集成电路高性能与通用处理器高灵活性的特点[1]，具有丰富的计算资源和互连资源，能够实现密码算法的高能效实现及灵活性切换。但由于其复杂的硬件结构，在映射密码算法时，也带来了一定的难度。不同的映射方式将直接影响算法的实现性能及功耗，因此合理的映射方式是充分发挥粗粒度可重构阵列结构优势的关键[2]。

将密码算法映射到CRCLA上，从数据流图的角度出发，映射问题实际上就是布局和布线两个子问题的组合。目前，国内外针对布局布线算法的研究主要是基于现场可编程逻辑门阵列(Field Programmable Gate Array, FPGA)进行的，而该项研究上均是将布局布线过程拆分成两个阶段进行的，按照先布局后布线的顺序[3,4]，采用了模拟退火算法及路径搜索算法。与传统的FPGA的布局布线算法不同的是，在CRCLA的映射问题上，布局和布线两个子问题存在着很大的关联性，若将其独立分

收稿日期：2021-01-04；改回日期：2021-03-05；网络出版：2021-03-27
*通信作者： 章宇雷 462960074@qq.com

开，将不能达到较好的映射效果，且映射失败往往发生在布线的过程中，因此需要将两者结合。文献[5]提出了一种基于图映射的算法SPKM启发式映射算法，通过“分裂-推送”的方式将循环部分映射至粗粒度可重构阵列上，减少了映射的阵列行数，但其流水性能较差；文献[6]提出了一种存储感知映射方法RAMP，在调度步骤之前，智能地探索各种路由资源，提高了映射的效率和质量；文献[7]提出了一种基于传输感知的有效循环映射方法TAEM，该方法可以有效利用CGRA上的异构资源，显著提高了编译速度；文献[8]对数据进行存储划分后利用模调度的方式进行映射，并采用了路径重用进一步优化路由资源，但其依赖可重构单元进行路由，并未讨论存在互连资源的可重构阵列结构；文献[9]提出了一种以边为中心的模调度(Edge-centric Modulo Scheduling, EMS)映射算法，在时间维度上的模调度采用了边的映射为中心，节点为边放置过程中的产物的方式，缩短了算法的搜索空间，减少了编译时间的同时，也能够达到较好的映射性能，但其搜索空间可能存在局部最优解；文献[10]针对流水气泡问题进行了优化，通过直通算子共享的方式减少了算法的启动间隔，提升了算法映射的性能等。

上述算法主要基于通用型可重构阵列的结构下进行讨论的，而密码算法的运算具有特殊性，且CRCLA内部由专为密码处理而设计的运算单元排列而成，通过配置信息来形成特定的数据通路[11]，因此上述映射算法均不适用基于CRCLA的密码算法映射问题。本文通过结合密码算法的运算特征及CRCLA的硬件结构，提出了一种映射过程中的建模方式，对映射过程具有更加清晰地描述；并提出了一种以边来指导节点的空间映射算法，通过确定根节点的映射后，按照关键路径优先的原则构造搜索步，并制定了路径搜索的方案使得映射路径最短，为避免映射失败，引入了回溯机制，提升了映射的能效及效率。

2 粗粒度可重构密码逻辑阵列映射模型

2.1 CRCLA基本结构

本文基于课题组前期设计的一款粗粒度可重构密码逻辑阵列开展映射算法验证，该阵列结构已在55 nm工艺下流片并验证。图1所示为课题组设计的4×4规模阵列，该阵列结构中，4×4个同构的运算单元(Processing Element, PE)构成了PE的计算资源，运算位宽为32 bit，其内部包含了多个运算功能单元(Functional Unit, FU)，如比特置换单元、逻辑运算单元、非线性运算单元等，能够满足密码算法运算的基本需求，并且各个FU单元之间存在互连网络，能够实现数据的互通。为消除密码运算过程中的流水气泡以及满足部分算法数据的寄存需求，在各个FU单元后添加了寄存器传输网络，可以通过配置来实现数据的寄存。同时，每一行的PE之间存在级联网络，能够满足部分密码算法大位宽数据的处理需求。

连接盒(Connect Box, CB)和开关盒(Switch Box, SB)构成了CRCLA的互连资源，CB分布在各个PE的4个方向上，能够实现PE与PE之间的互连以及PE与SB之间的互连；SB位于4个CB的中心，能够实现各个CB之间的互连。各个单元之间采用双向数据传输的方式，同一个方向上不能同时存在两个输出或输入，否则即为错误映射方式。

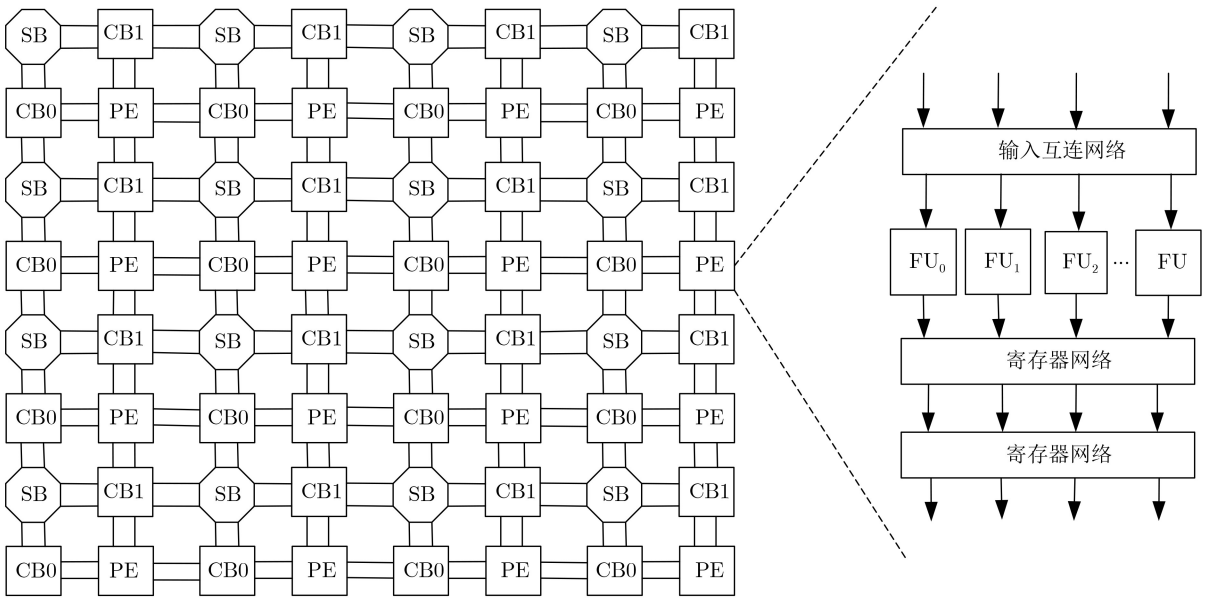


图 1 CRCLA整体硬件结构

2.2 CRCLA映射描述形式

基于以上分析，密码算法的映射问题实际上就是将密码算法转换成CRCLA的配置数据的过程。考虑到密码算法表示的多样性，结合粗粒度可重构密码逻辑阵列的结构，采用密码算法的数据流图作为密码算法映射问题的输入。

在进行数据流图的映射之前，由于CRCLA的一个PE中含有多种运算功能单元，能够同时实现多个节点的运算，为充分利用PE内部资源以及降低映射难度，需要对初始的密码算法数据流图进行任务划分，任务划分过程参照文献[12]，本文不再赘述。

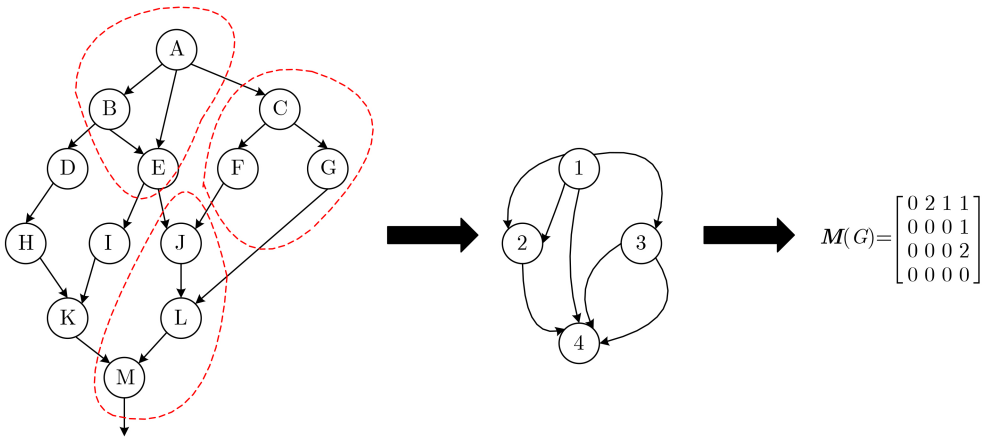
在经过任务划分后，同样可以得到一个新的数据流图，但该数据流图中的节点不再表示单一的运算节点，而是由多个运算节点形成的节点簇，在后文中，如无特殊说明，均称为节点，且本文中的密码算法数据流图均为经过了任务划分后的数据流图。因此，各节点之间可能存在多条互连关系。

定义1[13]　密码算法数据流图可以表示为一个二元数组$G=\{V,E\}$。节点集$V=\{v_i|v_i,1\le i\le n\}$，$|V|=n$为$G$中节点的个数。有向边集$E=\{e_{ij}|e_{i,j}=<v_i,v_j>,1\le i,j\le n\}$，其中，$e_{ij}$表示从$v_i$到$v_j$的有向边，$v_i$先于$v_j$执行，$|E|=m$为边的数量。

为便于后续算法的设计及分析，将密码算法经过划分后的数据流图中的各节点及互连关系通过矩阵$\boldsymbol{M}(\boldsymbol{G})$进行表示。

定义2　数据流图矩阵$\boldsymbol{M}(\boldsymbol{G})=\{b_{ij}|1\le i,j\le n\}$。若$v_i$与$v_j$之间存在$m$条有向边，且$v_i$指向$v_j$，$b_{ij}$=$m$，否则$b_{ij}$=0。矩阵中的第$i$行(列)反映出节点$v_i$与其他节点的互连关系。

下面将举例说明此种表示方式。如图2(a)所示为一个初始的数据流图，此时，一个节点表示一个运算操作；在经过任务划分后，得到如图2(b)所示的数据流图，此时一个节点中包含多个运算操作；图2(c)即为该数据流图的矩阵表示方法。



(a) 初始数据流图　　(b) 划分后　　(c) 矩阵表示

图 2 数据流图的矩阵表示过程

同样地，对粗粒度可重构密码逻辑阵列的计算及互连资源参数化模型进行表示。图3所示为对PE的资源占用情况采用链表式描述。

Tag_row(PE)表示当前PE所在行数，Tag_line(PE)表示当前PE所在列数，Occupied(PE)表示当前PE是否被占用，其值可取0或1，值为1表示被占用。当阵列规模较小时，可能无法容纳一个数据流图，需要多个配置页面来进行配置，因此添加一个Page(PE)来表示当前PE的配置页面。

对于互连资源，由2.1节可知，每一个PE周围均有8个互连开关盒，为便于描述，将一个PE左边及上边的两个CB、左上角的一个SB作为一个集合，如图4(a)所示。

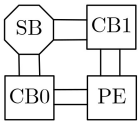
Tag_row(PE)	Tag_line(PE)	Occupied (PE)	Page(PE)

图 3 PE资源占用情况表示

SB及CB的位置用该PE所在的行列进行表示，同时需要有对应的标量对其进行区分，其表示方法如图4(b)所示。

Tag(Con)表示当前链表指向的互连单元，其值可取0, 1, 2，分别代表SB, CB0, CB1。E_in, E_out, N_in, N_out, S_in, S_out, W_in, W_out分别表示东南西北4个方向的输入和输出占用情况，其值可取0或1，值为1表示已被占用，即表示不能从此方向输入(输出)。

综上，本小节对密码算法的数据流图和CRCLA的计算资源及互连资源进行了描述，使得在映射过



(a) 一个包含互连及运算单元的集合

Tag_row(PE)	Tag_line(PE)	Tag(Con)					
E_in	E_out	S_in	S_out	W_in	W_out	N_in	N_out

(b) 表示方式

图 4 一个包含互连及运算单元的集合及其表示方式

程中能够更清晰地表示资源占用情况及映射的输入输出，为后续的算法映射奠定了一定的基础。

3 密码算法映射分析及问题定义

3.1 映射分析

由2.2节分析可知，密码算法映射问题面向的是一个经过划分后的能够满足CRCLA的各种约束的数据流图。由于划分后的数据流图中的每个节点在映射时直接对应于一个PE，且PE内部各个FU均可通过全互连网络实现数据之间的交互，因此无需考虑节点内部算子的连接关系。受限于面积限制，CRCLA各个PE之间无法采用全互连网络，因此，其互连资源较为匮乏。通常，传统的映射方法都是以节点为中心的，其重点在于如何将节点分配给各个PE，然而当密码算法映射时，随着映射的推进，即使还存在有大量的空闲状态下的PE，也不可避免地由于互连资源限制导致映射失败。

密码算法的映射性能取决于映射到阵列上的关键路径的延迟，而关键路径上的延迟同时取决于计算单元PE的关键路径延迟以及互连资源的关键路径延迟。假设一个密码算法映射到阵列上，则其关键路径的延迟可以表示为

$$T_{\text{critical}} = \sum_{i=1}^{n} t_{\text{PE}_i} + \sum t_{\text{CB}} + \sum t_{\text{SB}} \tag{1}$$

其中，t_{PE_i}表示关键路径中各个节点对应的各PE的关键路径延迟，t_{CB}表示CB的关键路径延迟，t_{SB}表示SB的关键路径延迟。则一个密码算法的吞吐率可以表示为

$$\begin{aligned}\text{Throught} &= \frac{L}{T_{\text{critical}}} \\ &= \frac{L}{\sum t_{\text{PE}_i} + \sum t_{\text{CB}} + \sum t_{\text{SB}}}\end{aligned} \tag{2}$$

其中，L表示密码算法的数据长度。在任务划分阶段各个节点内的算子对应的PE内部的计算功能单元已经确定，且一个节点对应一个PE的映射，故PE内部的关键路径延迟几乎无法优化，假设关键路径上经过的CB, SB个数分别为x, y，则$\sum t_{\text{CB}}$和$\sum t_{\text{SB}}$的值可以表示为

$$\sum t_{\text{CB}} + \sum t_{\text{SB}} = x t_{\text{CB}} + y t_{\text{SB}} \tag{3}$$

由式(3)可知，在布局布线可以通过尽量减少x, y的值来降低关键路径上的延迟，即减少CB, SB的个数。因此各个节点映射的PE应尽可能的紧凑，且各PE之间的连线应尽可能短。

结合以上所有分析，互连资源的映射直接决定了密码算法的映射性能及映射的成功与否。另外，对于密码算法的映射问题，经过大量研究可知，其布局及布线有着十分紧凑的联系，不能将两者单独分开进行讨论。

3.2 问题描述

将阵列中计算单元及互连单元均看作一些分布的节点，为了进一步对密码算法的映射问题进行数学化描述，进行如下定义：

定义3 一个$N \times N$的CRCLA的资源集合可以表示为$C = \{P, L\}$, $P = \{p_{ij} | 1 \le i, j \le N\}$为阵列中的计算单元PE，$L$为阵列中的互连单元集合。对于其中任意两个元素，表明p和q之间存在着互连且q可以使用p的计算结果，但p不能使用q的计算结果。

定义4 一个完整的应用映射操作可以描述为一个映射函数$f: G \to C$，同时包含了两个函数$f_V: V \to P$, $f_E: E \to L$。f_V为一个单射函数，它将应用内核的节点$v \in V$映射到可重构计算单元$p \in P$；f_E为一个多值函数，它将应用内核的数据依赖边$e = < u, v > \in E$映射到互连资源$l \in L$。

定义5 路径存在约束：$\forall e_1 = < u_1, v_1 >, e_2 = < u_2, v_2 > \in E, e_1 \ne e_2$，若$f_E(e_1 = < u_1, v_1 >) = l_1 \in L, f_E(e_2 = < u_2, v_2 >) = l_2 \in L$，则 $l_1 \cap l_2 = \emptyset$。

定义6 方向一致性约束：$\forall u, v \in V, e = < u, v > \in E$，若$f_E(e = < u, v >) = l \in L$，$f_V(u) = p \in P$, $f_V(v) = q \in P$则$\forall l' \in l$, l'的方向均与l一致，即$p \to q$。

定义7　密码算法映射问题：给定一个经过划分后的密码算法数据流图$G=\{V,E\}$和阵列$C=\{P,L\}$，在满足路径存在约束、方向一致性约束及资源约束的前提下，找到一个最少使用互连资源的映射$f(G)$。

4　算法设计

4.1　EMS算法概述

EMS(Edge-centric Modulo Scheduling)算法，是一种以边为中心的模调度算法，它摒弃了传统的先布局后布线以节点为中心的模拟退火映射算法，采用显示的方法来取代模拟退火的隐式管理，传统的模拟退火算法在映射时编译时间过长，而EMS算法则采用了以数据流图中的边为中心的映射方式，首先考虑的是布线的资源利用率问题，在路由边的过程中寻找可以映射节点的单元。因此，只要边映射成功了，节点的映射也随之成功，这一方法将布局布线两个过程结合起来，搜索空间大大缩小，提高了编译的效率。在映射过程中，遵循以下几个准则：(1)最小化布线资源；(2)主动避免布线失败；(3)对边的映射进行优先级度量。

EMS算法是一种时域映射模调度算法，而本文所研究的密码算法具有强烈的数据相关性，在映射上通常为空间映射，即不存在子图之间的时序问题，且面向的对象为互连及计算资源较为丰富的粗粒度密码逻辑阵列，模调度将不适用于密码算法的映射。因此，该算法难以适用于本文的映射问题，本文在此基础上，针对密码算法在CRCLA上的映射问题，提出一种以边为中心的空间映射算法(Edge-centric Cipher Logical array Mapping, ECLMap)。

4.2　ECLMap算法策略

在总体映射策略上，假设阵列资源能够满足密码算法的映射需求，采用空间映射的手段，结合上述分析，密码算法的映射问题更贴切于EMS算法中的以边为中心指导节点进行映射的思想。因此在映射过程中，同样遵循4.1节提到的EMS算法的3个准则。

在映射一个密码算法数据流图时，按照优先级顺序将其拆分成逐级递进的递增子图，以边为引导，按照递增子图的顺序依次进行映射。图5所示为对一个数据流图拆分的过程。

如图5所示，在映射由A, B, C, D4个节点组成的数据流图时，按照递增子图的顺序进行映射，对图1而言，假设首先映射节点A至某个PE后，从该PE出发沿着阵列对边1进行搜索，当寻找到未被占用的PE时，停止本次搜索，此条路径即为边1对应的映射对象，而该PE为节点B的映射对象，即同时完成了该子图的布局布线。依次按照递增子图的顺序进行搜索。此图相应的搜索路径顺序为A1B→B2C→C4D→D3A。结合密码算法的映射特征及CRCLA的硬件结构，采用以下策略来完成整个映射过程：

策略1 起始节点的映射。就绪队列中包含节点及节点之间的边，在进行边的映射之前，首先需要确定一个起始位PE，然后从这个PE出发，对边进

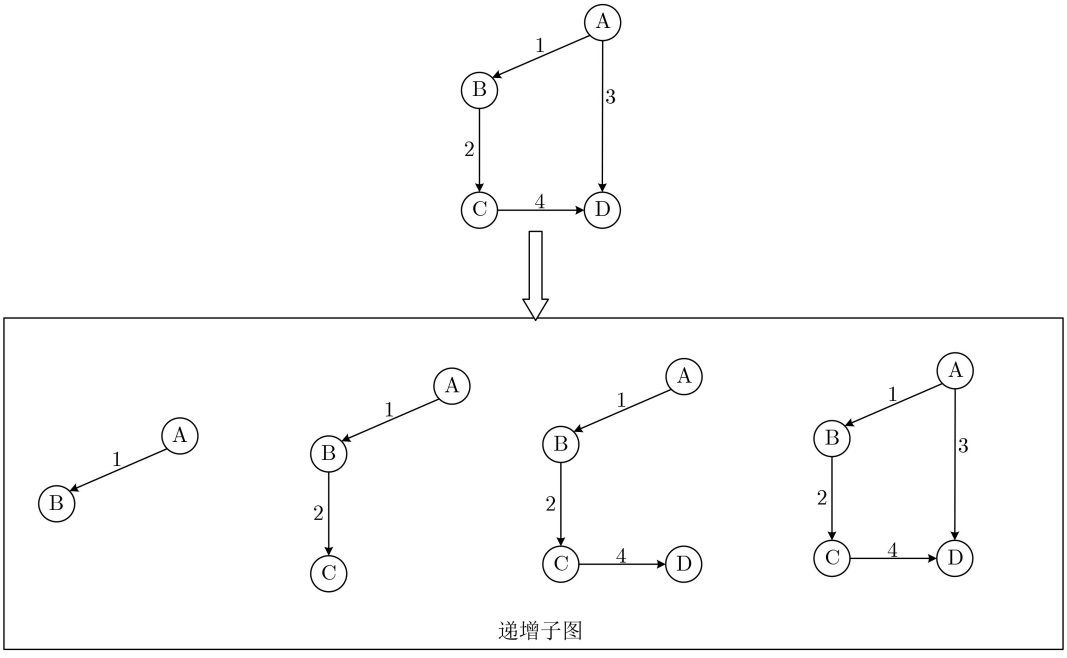


图 5 数据流图拆分映射过程

行映射。由于目标阵列的第1行PE为输入行PE，外部数据只能通过第1行PE输入而进行运算，因此，第1行PE作为起始节点的映射对象。由于本文的研究内容为空间映射，无需考虑各个节点的时间上的先后顺序，因此，从所有入度为0的起始节点中，按照扇出边的值从大到小的顺序，选择扇出边的值最大的起始节点进行映射。这是因为扇出边的值较大的节点对互连资源的需求较大，随着不断进行映射，可用的互连资源和PE个数不断减少，易受其他已经映射的节点占用的资源影响，可能难以找到有效的映射。

策略2 搜索步的构造。在确认了起始节点后，需要明确搜索路径的顺序，即构造一个搜索步。为贴合密码算法的运算特性，一个已映射的节点可能存在多个依赖关系，也就存在多条有向边，从该节点出发，优先选择关键路径上的边进行映射。而如果优先映射非关键路径的边，可能会造成在映射关键路径的边时，由于部分互连资源被占用，需要绕线从而使用了更多的互连资源，不仅影响其他节点及边的映射，还可能使得关键路径的延迟变大从而导致最终的映射性能不佳，甚至会导致之后的边映射失败。

因此，本文在深度优先搜索的基础上，按照关键路径优先级最大的方式构造搜索步。该过程可以描述为如下步骤：

步骤 1 输入为算法的数据流图G，输出为搜索路径$\mathrm{SE}=\{e_1,e_2,\cdots,e_n\}$。从根节点$v$开始，将该节点标记为已访问；

步骤 2 将关键路径优先纳入集合SE中，并将这条关键路径上的节点均标记为已访问；

步骤 3 搜索与根节点v相连的所有的边$e=<v,u>$，若节点u未被访问，则将边e纳入SE中，然后以节点u为起点，循环步骤3至与v相连的所有的边均已被访问。

步骤 4 若图中仍然存在未被访问的节点，则以该节点为起始节点，跳至步骤3，直到所有节点均被标记为已访问。

步骤 5 得到一条搜索路径$\mathrm{SE}=\{e_1,e_2,\cdots,e_n\}$。之后的映射则按照该搜索路径进行。

策略3 路径选择。一个PE周围存在着东南西北4个方向的互连资源，即连接盒CB，当确定一条就绪队列中待映射的边e后，通过查找链表来确定哪个方向的CB没有被占用；在映射前期，当存在多个方向的CB未被占用时，有多个CB可以进行选择。此时，从已映射的节点对应的PE进行多条路径搜索。对于单条路径搜索，当首次搜索到某个CB周围存在未映射的PE时，作为本次路径的终点，停止本次路径搜索，并记录该条路径所经过的CB, SB以及未映射的PE的编号。定义一条路径搜索(边$e=<u,v>$的映射)为

$$\begin{aligned}\mathrm{Road}(u,v)=&\mathrm{Path}[\mathrm{PE},\mathrm{CB},\mathrm{SB}]=\mathrm{PE}(x_1,y_1),\\&\mathrm{CB}(x_2,y_2),\mathrm{SB}(x_3,y_3),\cdots,\mathrm{PE}(x_n,y_n)\end{aligned}$$

在进行路径搜索时，为避免大量无意义的解而造成编译时间的浪费，算法并不穷举有向边的所有可能的映射子图。当搜索到某个未映射的PE时，对路径的终点PE进行分析，定义一个PE的亲和度为通过与该PE可进行数据交互的其他PE的个数和数据流图中与待映射节点u存在直接连接关系的节点个数之间的关系。通过计算该PE与待映射节点的亲和度来进行取舍，其计算公式为

$$\mathrm{Affinity}(u,\mathrm{PE})=\begin{cases}0, & \mathrm{Con}_u>\mathrm{Con}_{\mathrm{PE}}\\ \dfrac{\mathrm{Con}_u}{\mathrm{Con}_{\mathrm{PE}}}, & \mathrm{Con}_u\le\mathrm{Con}_{\mathrm{PE}}\end{cases}\tag{4}$$

式(4)为亲和度计算公式，其中，Con_u为除去已映射的节点，与节点u有着直接连接关系的节点的个数；$\mathrm{Con}_{\mathrm{PE}}$表示与路径终点PE能够进行数据交互的未映射的PE的个数。由公式可知，若$\mathrm{Con}_u>\mathrm{Con}_{\mathrm{PE}}$，则在后续的映射过程中会导致与节点$u$有直接连接关系的部分节点映射失败，因此其亲和度值为0；若$\mathrm{Con}_u\le\mathrm{Con}_{\mathrm{PE}}$，则该条路径的后续映射能够满足其映射需求，此时，若Con_u越接近于$\mathrm{Con}_{\mathrm{PE}}$，则该PE越符合该节点$u$的映射需求，则其亲和度的值越高，其最大值为1。这样既能够保证节点u的直接相邻节点被成功映射，又不会造成资源的浪费，保留了其他具有更多可进行数据交互的PE个数的PE，可供后续节点的映射。因此，若该PE的亲和度值为0，则删除该条搜索路径。

为使其互连资源尽可能少，在经过算法搜索后得到的所有路径中，筛选出最短的路径，即所经过的CB, SB个数最少，从最短的路径中，选出亲和度值最高的路径，作为最佳路径，为本次映射的最终方案。若存在多个最佳路径，则进行记录，并随机挑选一个路径作为本次映射的最终方案。

策略4 回溯机制。该算法采用的搜索方式是启发式的，在后续的节点的映射过程中，随着映射的推进，由于可用的资源越来越紧张，可能会出现映射失败的情况。因此，采用一种反向回溯机制，该机制基于一种称为回溯表(Failure Table，FT)的数据结构[14]。在前向映射某个节点v_i时，每次挑选出最佳路径时，对剩下的路径依照亲和度值及路径大小按照从小到大进行排序，因此，回溯表可以表示为$\mathrm{FT}_i=(v_i,\{\mathrm{Road}_1,\mathrm{Road}_2,\cdots,\mathrm{Road}_n\})$，其中，

Road_i则表示从v_i的前向节点对应映射的PE出发，进行搜索得到的路径，包含经过的CB, SB的位置信息及终点PE的位置信息。

若在映射过程中，从某个节点v_i映射对应的PE出发的路径在搜索的过程中，当出现以下情况时：(1)无法搜索到未映射的PE；(2)所有路径中的亲和度的值均为0；(3)没有可以进行映射的路径；则进行反向回溯，回到在映射节点v_j时的路径搜索过程中，在回溯表中删除所有到达该PE的路径。之后查找回溯表中的信息，选择亲和度值最大的最短路径Road_j进行映射，并将Road_j从回溯表中删除，然后进行后续的边及节点的映射。当出现v_j的回溯表为空时，说明v_j及前向路径仅存在当前的一种映射方案，则往前回溯到v_j的前向节点v_i的回溯表中继续寻找其他映射方案，并依此类推，直到找到回溯的节点的回溯表不为空。

4.3 ECLMap完整算法描述

如表1所示，为ECLMap的完整过程的伪代码。算法的输入为经过任务划分后的密码算法数据流图及CRCLA的资源集合，输出为最终的映射结果。

首先，对算法映射结果以及回溯表进行初始化(行(1) —行(2))；接着依据策略1对起始节点进行选取，并依据策略2构造相应的搜索步(行(3)—行(5))。然后对搜索步中的起始节点进行映射，再从已映射的节点对应的PE出发，按照搜索步的顺序依次对每一条边的映射路径依据策略3进行搜索，并通过亲和度值来筛选可选路径，在所有可选路径中确定最短路径即为该边的最终映射方案，该路径的终点为节点对应的映射PE，并在回溯表中记录其他候选方案(行(6)—行(15))。若映射过程中某一条边没有可选路径，映射失败，则进行反向回溯，从而进行新的映射(行(16) —行(26))。直到所有的节点及边映射完毕后，输出最终映射结果。

表 1　ECLMap算法描述

输入： $G=\{V,E\}, C=\{P,L\}$
输出： Map

(1) FT←Initialize_FT(); //初始化回溯表
(2) Map←Initialize_MP(); // 初始化映射
(3) **repeat**
(4) V_{root}←Choose_Root_Node(V); //起始节点的选取
(5) SE←Sort_Edge(V_{root});　//构建搜索步
(6) PE_{root}←Map_Root(V_{root});　//起始节点的映射
(7) **for** each edge e in SE **do**
(8)　R_M←Search_Road(e); //路径搜索
(9)　M←Choose_Road_by_Affinity($R`_M$,PE);　//亲和度函数进一步路径选择
(10) $R`_M$←Choose_Min_Road(R_M,PE);　//选择最小的路径
(11)　**if** $M\neq\emptyset$ **then**
(12)　　Road_M←$M(0)$;
(13)　　V_M←Get_Destination(Road_M);　//路径终点即为节点的映射
(14)　　Upadate_mapping(Road_M, V_M, Map);
(15)　　FT←Updata_FT(V_M,FT,M,$M(0)$); //回溯表的确定
(16)　**else**
(17)　　R_{BK}←Edge_To_Backtrack(e,FT); //M为空则进行回溯
(18)　**end if**
(19)　**if** R_{BK}==NULL **then**
(20)　　**break;**
(21)　　**else**
(22)　　Modify_FT(R_{BK}, FT); //修改回溯表
(23)　　Delete_MP(R_{BK},Map);　//删除映射失败的路径
(24)　**end if**
(25) **end for**
(26) Until all nodes and edge in SE are mapped;
(27) **return** Map

5　实验验证与分析

为验证本文提出的ECLMap算法相较于其他算法在密码算法映射问题上的有效性及优势，本节在实验室前期研制的粗粒度可重构密码逻辑阵列上进行测试，该阵列结构已完成相关功能验证，在仿真平台下使用Verilog HDL进行了相应的模块化描述，本文选取了典型的密码算法AES, DES, A5-1, ZUC和SM3等密码算法进行映射实验，并利用了综合仿真验证工具进行了功能验证及相关性能测试。

采用Intel Core i5- 6300HQ@ 2. 30 GHz CPU及16 GB内存环境进行算法映射及仿真测试。首先，对算法自身的回溯机制进行了验证，主要体现在对有无回溯时算法映射的成功率进行了统计与分析，其结果如图6所示。从图6可以看出，由于引入了回溯机制，使得算法在某一步映射失败时，可以通过查询回溯表返回至上一步，选择其他候选方案进行重新映射，一定程度上避免了映射失败，提高了映射成功率。

为验证ECLMap算法的高效性，对算法的编译时间进行了测试，并将其与SA[4], SPKM[5], EPIMap[15]和2-StepACO[16]等几种通用的映射算法进行了对比，结果如图7所示。从图中可以看出，本文提出的ECLMap算法对密码算法的数据流图与CRCLA的硬件结构之间紧密的联系进行了充分的考虑，并对初始的密码算法的数据流图进行了任务

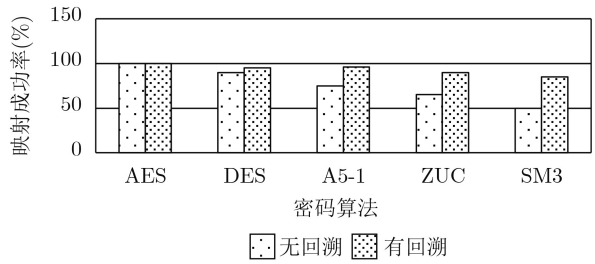


图 6 ECLMap算法映射成功率对比

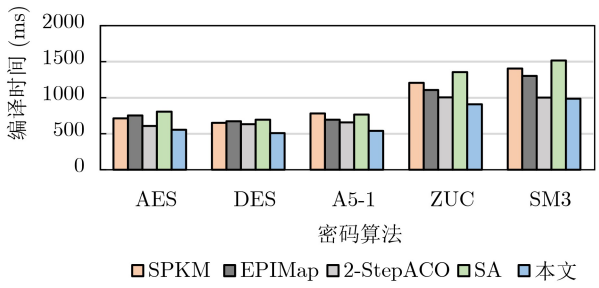


图 7 CRCLA密码算法映射编译时间对比结果

划分的优化，因此在编译时间上相较于其他算法有着一定的优势，平均减少了约25%。

对于密码算法，其更为重要的是映射在CRCLA硬件结构上实现的性能及功耗。因此，经过不同算法运行后得到的不同配置方案下的配置信息，在CRCLA仿真平台下进行映射实验。表2和表3分别展示了几种通用的映射算法和本文的ECLMap算法下的映射的性能及1.2 V工作电压下的功耗结果对比。图8进一步直观地显示了几种密码算法在不同的映射算法下实现的能效。从图8可以看出，在编译时间优于其他算法的同时，ECLMap算法在映射能效上同样比其他算法要高，平均提升

表 2 不同映射算法下的密码算法映射性能(Mbps)

密码算法	SPKM	EPIMap	2-StepACO	SA	本文
AES	410	420	446	440	450
DES	247	225	219	256	269
A5-1	115	110	112	120	128
ZUC	160	156	150	168	175
SM3	228	217	219	235	256

表 3 不同映射算法下的密码算法映射功耗(mW)

密码算法	电压(V)	SPKM	EPIMap	2-StepACO	SA	本文
AES	1.2	273	280	265	260	250
DES	1.2	256	267	252	248	241
A5-1	1.2	278	286	272	266	245
ZUC	1.2	285	291	280	273	258
SM3	1.2	293	296	283	275	264

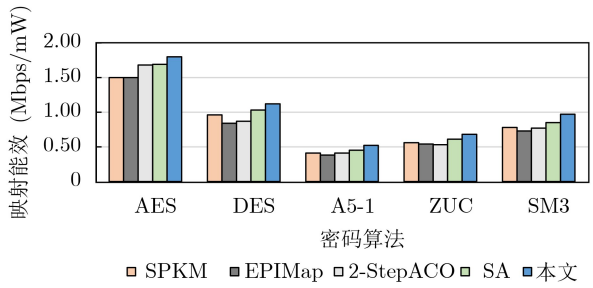


图 8 CRCLA密码算法映射能效对比结果

了约20%，这是因为ECLMap算法的设计对于密码算法更具针对性，从而能够满足密码算法的高能效映射。

6 结束语

为实现密码算法在粗粒度密码逻辑阵列上的高能效映射，对密码算法的映射问题进行了分析。本文对密码算法的数据流图及粗粒度可重构密码逻辑阵列的资源进行了相应的描述，分析了影响密码算法映射性能的关键因素及可进行提高的方面，并提出了一种以边为中心的密码逻辑阵列映射算法ECLMap，算法引入了回溯机制来进一步确保映射的成功率。实验表明，在4×4阵列规模下，该算法无论是编译时间还是映射性能上，相较于其他算法，均具有一定的优势。下一步将结合可变的阵列规模对算法进行改进。

参 考 文 献

[1] 杜怡然, 李伟, 戴紫彬. PVHArray: 一种流水可伸缩的层次化可重构密码逻辑阵列结构[J]. 电子学报, 2020, 48(4): 781–789. doi: 10.3969/j.issn.0372-2112.2020.04.020.
DU Yiran, LI Wei, and DAI Zibin. PVHArray: A pipeline variable hierarchical reconfigurable cryptographic logic array structure[J]. *Acta Electronica Sinica*, 2020, 48(4): 781–789. doi: 10.3969/j.issn.0372-2112.2020.04.020.

[2] 杜怡然, 南龙梅, 戴紫彬, 等. 可重构分组密码逻辑阵列加权度量模型及高能效映射算法[J]. 电子学报, 2019, 47(1): 82–91. doi: 10.3969/j.issn.0372-2112.2019.01.011.
DU Yiran, NAN Longmei, DAI Zibin, *et al*. Reconfigurable block cryptographic logic array weighted metric model and high energy-efficient mapping algorithm[J]. *Acta Electronica Sinica*, 2019, 47(1): 82–91. doi: 10.3969/j.issn.0372-2112.2019.01.011.

[3] 韩国栋, 肖庆辉, 张帆. 可重构系统中硬件任务布局布线算法研究[J]. 计算机科学, 2011, 38(11): 291–295. doi: 10.3969/j.issn.1002-137X.2011.11.068.
HAN Guodong, XIAO Qinghui, and ZHANG Fan. Algorithms of placing and routing hardware task in reconfigurable system[J]. *Computer Science*, 2011, 38(11): 291–295. doi: 10.3969/j.issn.1002-137X.2011.11.068.

[4] 行华彧, 景乃锋. 一种基于多阶段模拟退火的异构可重构阵列布局算法[J]. 微电子学与计算机, 2020, 37(6): 1–5. doi: 10.19304/j.cnki.issn1000-7180.2020.06.001.
XING Huayu and JING Naifeng. A placement algorithm for HGRA based on multi-stage simulated anneal[J]. *Microelectronics & Computer*, 2020, 37(6): 1–5. doi: 10.19304/j.cnki.issn1000-7180.2020.06.001.

[5] YOON J W, SHRIVASTAVA A, PARK S, *et al*. SPKM: A novel graph drawing based algorithm for application mapping onto coarse-grained reconfigurable architectures[C]. The 2008 Asia and South Pacific Design Automation Conference, Seoul, Korea, 2008: 776-782. doi: 10.1109/ASPDAC.2008.4484056.

[6] DAVE S, BALASUBRAMANIAN M, and SHRIVASTAVA A. RAMP: Resource-aware mapping for CGRAs[C]. The 55th Annual Design Automation Conference, San Francisco, USA, 2018: 1–6. doi: 10.1145/3195970.3196101.

[7] KOU Mingyang, GU Jiangyuan, WEI Shaojun, *et al*. TAEM: Fast transfer-aware effective loop mapping for heterogeneous resources on CGRA[C]. The 57th ACM/EDAC/IEEE Design Automation Conference, San Francisco, USA, 2020: 1–6. doi: 10.1109/DAC18072.2020.9218668.

[8] 张兴明, 袁开坚, 高彦钊. 基于存储划分和路径重用的粗粒度可重构结构循环映射算法[J]. 电子与信息学报, 2018, 40(6): 1520–1524. doi: 10.11999/JEIT170748.
ZHANG Xingming, YUAN Kaijian, and GAO Yanzhao. Coarse grained reconfigurable architecture loop mapping algorithm based on memory partitioning and path reuse[J]. *Journal of Electronics & Information Technology*, 2018, 40(6): 1520–1524. doi: 10.11999/JEIT170748.

[9] PARK H, FAN K, MAHLKE S, *et al*. Edge-centric modulo scheduling for coarse-grained reconfigurable architectures[C]. The 17th International Conference on Parallel Architectures and Compilation Techniques, Toronto, Canada, 2008: 166–176. doi: 10.1145/1454115.1454140.

[10] 明畅. 面向密码算法的可重构自动映射方法的设计与实现[D]. [硕士论文], 东南大学, 2018.
MING Chang. Design and implementation of a reconfigurable automatic mapping method for cipher algorithms[D]. [Master dissertation], Dongnan University, 2018.

[11] 李伟, 高嘉浩, 杜怡然, 等. 一种密码专用可编程逻辑阵列的分组密码能效模型及其映射算法[J]. 电子与信息学报, 2021, 43(5): 1372–1380. doi: 10.11999/JEIT200079.
LI Wei, GAO Jiahao, DU Yiran, *et al*. Energy efficiency model and mapping algorithm of block cipher for cipher specific programmable logic array[J]. *Journal of Electronics & Information Technology*, 2021, 43(5): 1372–1380. doi: 10.11999/JEIT200079.

[12] LIU Min, YAN Yinjian, LI Wei, *et al*. A dependence-first clustering based partitioning algorithm for coarse-grained reconfigurable cipher logic array[C]. 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference, Chongqing, China, 2018: 88–92. doi: 10.1109/IAEAC.2018.8577573.

[13] 陈乃金, 江建慧, 陈昕, 等. 一种考虑执行延迟最小化和资源约束的改进层划分算法[J]. 电子学报, 2012, 40(5): 1055–1066. doi: 10.3969/j.issn.0372-2112.2012.05.032.
CHEN Naijin, JIANG Jianhui, CHEN Xin, *et al*. An improved level partitioning algorithm considering minimum execution delay and resource restraints[J]. *Acta Electronica Sinica*, 2012, 40(5): 1055–1066. doi: 10.3969/j.issn.0372-2112.2012.05.032.

[14] 尹文志, 赵仲元, 毛志刚, 等. 一种快速高效的粗粒度可重构架构编译框架[J]. 微电子学与计算机, 2019, 36(8): 45–48, 53. doi: 10.19304/j.cnki.issn1000-7180.2019.08.010.
YIN Wenzhi, ZHAO Zhongyuan, MAO Zhigang, *et al*. A fast and efficient compiler framework for CGRAs[J]. *Microelectronics & Computer*, 2019, 36(8): 45–48, 53. doi: 10.19304/j.cnki.issn1000-7180.2019.08.010.

[15] HAMZEH M, SHRIVASTAVA A, and VRUDHULA S. EPIMap: Using Epimorphism to map applications on CGRAs[C]. The DAC Design Automation Conference 2012, San Francisco, USA, 2012: 1280–1287. doi: 10.1145/2228360.2228600.

[16] ZHOU Li, ZHANG Jianfeng, and LIU Hengzhu. Ant colony algorithm for Steiner tree problem in CGRA mapping[C]. The 4th International Conference on Information Science and Control Engineering, Changsha, China, 2017: 198-202. doi: 10.1109/ICISCE.2017.51.

徐金甫：男，1965年生，教授，博士生导师，研究方向为专用集成电路设计技术.

章宇雷：男，1996年生，硕士生，研究方向为可重构逻辑电路设计.

李　伟：男，1983年生，副教授，博士生导师，研究方向为密码处理器设计、ASIC专用芯片设计.

陈　韬：男，1979年生，副教授，硕士生导师，研究方向为安全专用芯片设计.

责任编辑：马秀强