# 基于演化硬件的硬件重构编码方案及演化算法研究

王婷，兰巨龙，邬钧霆
（国家数字交换系统工程技术研究中心，河南 郑州 450002）

**摘 要**：基于柔性网络技术背景，面向 SRAM 结构的 FPGA 平台，提出一种基于 LUT(look up table)结构的二维映射函数增量染色体编码方案（PMFICC, planar mapped function increments chromosome coding），该方案利用二进制配置文件串双平面映射方式进行编码变换，可有效提高硬件的重构效率。同时在该方案的基础上引入局部优化机制，提出一种改进的差分演化（MDE, modified differential evolution）算法，该算法可有效提高收敛速度和全局优化效率。最后对该算法进行了仿真验证，结果表明：MDE 算法改进了差分演化算法容易陷入局部最优的不足，可以更加逼近实际最优解。

**关键词**：演化硬件；硬件重构；染色体编码；元素图；差分演化
**中图分类号**：TP393.08 **文献标识码**：A **文章编号**：1000-436X(2012)08-0035-07

# Hardware reconfigurable coding and evolution algorithm based on evolvable hardware

WANG Ting, LAN Ju-long, WU Jun-ting
(National Digital Switching System Engineering & Technology Research Center, Zhengzhou 450002 China)

**Abstract:** A planar mapping function increments chromosome coding method based on FPGA platform with SRAM-architecture was proposed. The method could improve hardware reconfiguration efficiency by coding mapping realized by double platform mapping of binary configurable file string. Meanwhile, a betterment difference evolution algorithm was proposed based on local optimal mechanism introduced. The algorithm could promote convergence rate and whole efficiency. Finally, the result of the algorithm emulation shows that: MDE improves disadvantage of difference evolution algorithm with local optimal and approaches actual optimization outcome.

**Key words:** evolvable hardware; hardware reconfiguration; chromosome coding; element graph; difference evolution

## 1 引言

目前，以应用为驱动的互联网正朝着高速化和宽带化的方向发展。然而大量差异化业务的大规模部署及新型业务的不断涌现，使得现有网络无法满足其服务需求，底层固有硬件的支撑也不堪重负[1]。面向服务提供的新型网络技术体系的提出为解决上述问题提供了新的思路[2]。在该技术体系之下，对于网络交换节点来说，就是要研究功能可重构、性能可编程的路由交换设备，将控制平面与数据平面相分离，通过重构实现控制平面上对多种路由协议的支持，运行多种路由算法，表象的更新与维护，系统的配置与管理等，通过编程实现数据平面上的线速查表、分类、加密、队列管理等。然而数据平面的处理往往需要很强的实时性，传统模式下依赖的固有硬件系统已经无法适应新功能变更的要求，

**收稿日期**：2011-12-02；**修回日期**：2012-05-05
**基金项目**：国家重点基础发展计划（“973”计划）基金资助项目（2012CB315901, 2012CB315905）
**Foundation Item:** The National Basic Research Program of China (973 Program) (2012CB315901, 2012CB315905)

一旦有新的功能出现，系统或模块必须整体替换或升级，使得网络交换节点的可扩展性及灵活性变得很差。

对于底层硬件设计人员来说如果将开放可编程的逻辑器件作为开发平台，设计实现一系列具有基本功能的可重构硬件构件（硬件逻辑模块），当系统需要对新业务进行服务时，只需找到相关功能的硬件构件加以组合或改进，就可实现对新业务服务要求的支持[2,3]。这种在原有硬件平台上通过重构来支持新业务的方式，必然会在增强路由设备的灵活性及可扩展性、减小重构时隙、节约开发成本上具有很大的优势。

随着半导体工艺水平的迅速提高，单个可编程逻辑器件的片上资源越来越丰富，利用电子设计自动化（EDA）这类软件工具进行硬件下载实现所需硬件电路的方法，与芯片高集成度间的差异越来越大，硬件程序员所设计的硬件功能不能通过 EDA 工具很好地诠释于芯片之上，不仅对硬件资源的利用率不高，很大程度上影响了硬件设计者的初衷。与此同时，当电路功能需要重构来满足新业务的需求时，上述硬件实现方法只能重新编写或修改硬件程序，然后再通过 EDA 工具“翻译”后下载到芯片上，必然会导致重构时隙开销的增加，无法完成可重构路由交换节点的硬件重构要求。因此，利用演化硬件（EHW, evolvable hardware）来解决上述矛盾，成为可重构硬件重构技术研究的一个新热点[4]。演化硬件利用自身的快速并行计算能力及其在解空间的寻优搜索能力，来实现硬件电路的自动设计。

演化硬件可以用下面的公式来定义：EAs+PLD=EHW。即：演化算法+可编程逻辑器件=演化硬件。演化硬件是一种黑盒设计技术，对硬件设计者的专业知识依赖程度不是很高，能够最大程度的实现硬件设计的自动化。它的设计重点是让演化来生成所需电路，硬件设计者只需告诉它做什么而不是怎样做。演化硬件不仅可以对将要投入实现的硬件设计进行修改，而且可以对已经投入使用的电路进行修改，避免了替换已有硬件电路所需的开发成本，而且硬件程序员的设计错误也可以通过人工或进一步演化来修正，减小了设计风险和实现成本[5~9]。基于演化硬件的硬件重构技术可以解决快速发展的网络业务多样性的需求，同时可以节约研发及重复建设网络的硬件成本，符合现今社会绿色网络和低碳经济的需求。将可编程逻辑器件作为开发平台，实现可重构路由交换平台硬件电路重构的原理如图 1 所示。

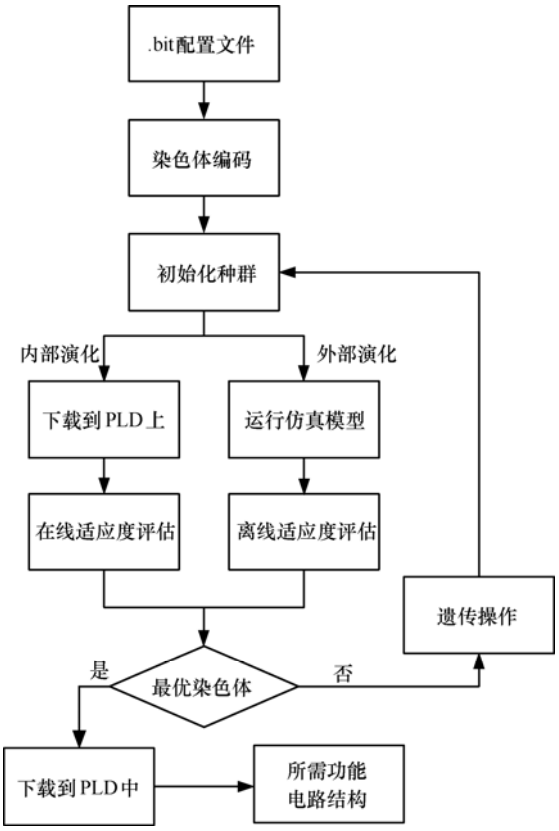


图 1 基于 PLD 的硬件电路重构原理

由图 1 可知，染色体编码方案是实现硬件重构的基础，决定了最终的硬件电路结构的好坏、遗传操作的进化速度以及搜索空间的大小，进而影响整个硬件重构的效率。因此，设计合理的染色体编码方案显得尤为重要。目前，相关学者已经研究出一些成熟的染色体编码方案[10]，大致可分为 2 大类：直接型编码和间接型编码。直接型编码方式就是将下载到PLD中的二进制配置位串直接作为演化算法中的染色体，对其进行遗传操作，以改变门级电路的连接，进而实现硬件电路结构的改变。然而电路规模的大小直接影响着 PLD 配置位串的长度，通常情况下其长度在几万位到几十万位之间[11]，如果直接将整个配置位串作为演化算法中的染色体进行编码，当演化过程对存储和计算资源的要求随电路规模的增长呈指数性增长时，势必会造成片上存储空间的大量开销以及处理器运算量的急剧增大，因此直接型编

码只适用于小规模的硬件电路的重构。间接型编码则是对 PLD 的配置文件进行一种变换，然后再对其进行编码，它将电路更高层次的表达方式作为遗传中染色体进行编码，在适应度评估和生成实际硬件电路配置时，需要经过逆变换后才能生成实际的 PLD 配置位串。间接型编码采用抽象表达的方法，其演化元素不是门电路，而是基本的硬件功能模块。

本文将演化算法与可编程逻辑器件相结合，首先提出一种基于 LUT 结构的二维映射函数增量染色体编码方案（PMFICC, planar mapped function increments chromosome coding），在此方案的基础上又提出了一种改进的差分演化算法，并证明了其有效性。

## 2　基于查找表 LUT 结构的二维映射函数增量染色体编码方案（PMFICC）

基于二级映射的演化硬件体系结构在基因型平面与表现型平面之间采用中间硬件表示层，基因型平面的编码只对中间层进行，而表现型平面的实际 FPGA 器件的配置位串是通过中间层进行转换获得的。中间硬件层的结构相对简单，因而其染色体编码表示比直接对 FPGA 配置编码简单，降低了演化过程中对存储和计算资源的需求。同时，中间硬件层屏蔽了底层 FPGA 的硬件特性，使得演化的设计方法能够适应普遍的条件和环境，能够适用到普遍的电路规模和类型中。Arostegui 等[12]提出了采用规整的功能单元构成二维可重构平面的结构，功能单元的数量可以任意增加，从而使得演化设计具有可扩展性。R. DeMara 等[13]采用二级映射机制实现基因型 / 表现型映射，通过虚拟演化硬件 FPGA(virtual EHW FPGA)实现染色体编码到 FPGA 配置位串的转换，其映射过程如图 2 所示。

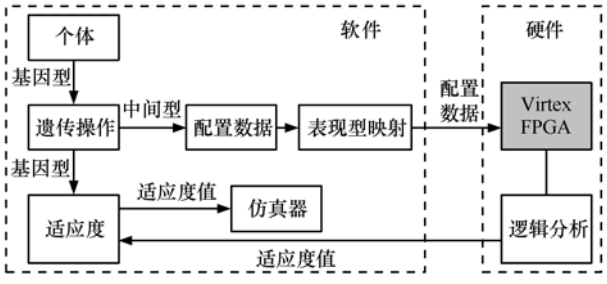


图 2　基于虚拟演化硬件的染色体编码

目前，硬件重构技术大多数是在基于 SRAM 工艺的 FPGA 平台上设计实现的，以 n-LUT 的 FPGA 为例，每一个 LUT 都可以看成是 *n*bit 地址线的 $2^n \times 1$ 的 RAM，当输入 *n*bit 地址信息时，输出就是 RAM 中相应单元中的状态值（$2^n$ 种逻辑功能）。为了研究可重构路由交换平台的硬件重构技术，首先将所需的实际硬件电路按照逻辑功能分成多个逻辑功能块，这些逻辑功能块由若干个 LUT 组成的二维阵列构成，LUT 之间相互连接形成整个硬件电路。本文提出的基于 LUT 结构的 PMFICC 编码方案，是根据实际的底层硬件资源（LUT 的个数）将二进制配置文件串（.bit 文件）即 Bitstream 文件，映射到一维状态平面（SP, status plane，）（也就是 LUT 的内部状态）和二维排序平面（CP, connection plane）（也就是 LUT 之间的连接关系）中，从而将冗长的一维平面上的二进制编码，映射为二维平面上的 PMFICC 编码。以 3 输入的 LUT 为例，片上资源由 16 个 3-LUT 组成，那么基于 LUT 结构的 PMFICC 编码方案示意图如图 3 所示。

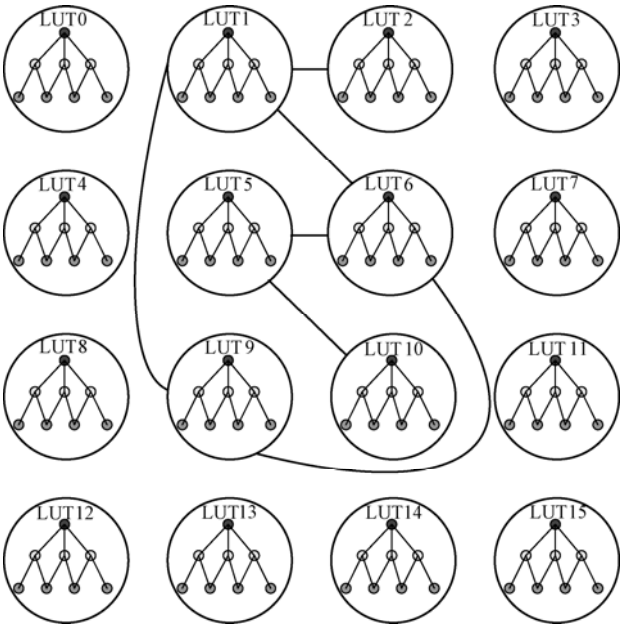


图 3　一种基于 LUT 结构的 PMFICC 编码方案

一维状态平面就可由逻辑功能函数 $f(x)=\{a, b, c, d, e, f, g, h\}$，（其中，$x$ 表示 LUT 的输入地址信息）的编码组合进行表示，如表 1 所示。那么，图 3 所示的电路结构在 PMFICC 编码方案中 SP 平面的编码表示为 $f_1 f_2 f_5 f_6 f_9 f_{10}$。

**表 1　　3-LUT 逻辑功能**

| $x$ | $f(x)$ |
|---|---|
| 000 | $a$ |
| 001 | $b$ |
| 010 | $c$ |
| 011 | $d$ |
| 100 | $e$ |
| 101 | $f$ |
| 110 | $g$ |
| 111 | $h$ |

那么，该方案的编码方法就将 16 个 3LUT 的二进制编码表示简化为 6 个 LUT 的函数表示，其 SP 平面编码就可简化为 6 个 LUT 的连接关系编码。可以用一个无向图 $G$ 来继续简化图 3 的电路结构，如图 4 所示：顶点代表参与电路结构的 6 个 LUT，LUT 间的连接关系则由无向边来表示。PMFICC 编码方案中 CP 平面的编码表示为 1-2 1-6 1-9 5-10 6-5 6-9。那么图 3 所示的电路结构的 PMFICC 编码表示为 $f_1 f_2 f_5 f_6 f_9 f_{10}$ 1-2 1-6 1-9 5-10 6-5 6-9。

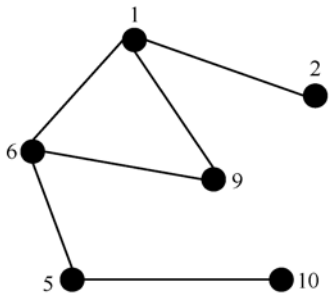


图 4　CP 编码平面的无向图

应用 PMFICC 编码方案重构电路结构时，分别对比重构前后 2 种电路结构 SP 编码和 CP 编码的差异，如果是 LUT 内部有变化，只需对 LUT 的内存状态进行修改得到相应的 $f(x)$ 编码；如果是路由资源发生变化即 LUT 的连接关系发生变化，那么可对前一种电路结构的无向图做逻辑运算（与、或、非、异或）得到所需的边序列编码。

在对 FPGA 进行硬件设计时，80%左右的资源都被路由资源所占，因此基于 PMFICC 编码进行动态重构研究时，首先要考虑 CP 编码的重构，也就是无向图的逻辑运算。为了减小重构时隙实现动态重构技术，首先在 FPGA 内部建立一系列可以完成基本功能的元素图（EG, element graph）库，当电路功能发生改变时，仅需从 EG 库中调取相应个数的 EG 进行逻辑运算，就可得到重构后新的电路结构，这将大大提高动态重构的效率。如图 5 所示：元素图 $G_1$ 和 $G_2$，那么重构后的电路 $G_1 \cup G_2$，$G_1 \cap G_2$ 以及 $G_1 \oplus G_2$ 可由 $G_1$ 和 $G_2$ 的逻辑运算得到。

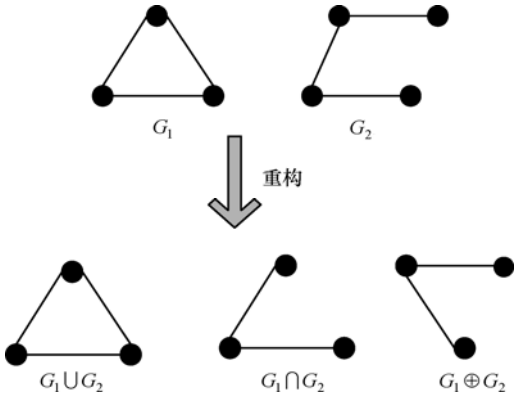


图 5　元素图重构

通过前面的分析，知道 PMFICC 编码方案是以增加染色体基因型到表现型的复杂性为代价，将冗长的一维平面上的二进制编码映射到 SP 平面和 CP 平面。CP 平面的编码只对无向图进行重构，SP 平面则对实际 FPGA 器件的配置位串是进行重构，降低了直接对 FPGA 配置位串进行演化的冗余，减少了演化过程中对存储和计算资源的需求。该编码方法特别适用于基于 Bitstream 的动态重构，重构后新的电路功能与重构前的电路功能差异（Bitstream 差异）可以通过演化算法，将前后 2 代染色体的差异引入到演化过程中，进而得到新的最优染色体。下面将介绍一种改进的差分演化算法。

## 3　一种改进的差分演化算法

如何在上述编码方案的基础上有效、快速地实现目标方案的最优演化是本节研究的重点问题。差分演化（DE, differential evolution）算法是一种非常适用于基于函数增量的染色体编码演化算法，具有操作简单、收敛速度较快等特点，并且其在求解单目标优化问题方面已经取得了成功[14~16]。典型的差分演化算法包括以下几个步骤。

1) 初始化：利用 $NP$ 个维数为 $D$ 的实数向量作为每一代的种群，每个个体表示为 $x_{i,G}$，其中，$i$=1, 2, …, $NP$，$i$ 为个体在种群中的序列，$G$ 为演化代

数，*NP* 为种群规模。初始化种群的一个方法就是在给定边界约束内的值中随机选择，一般假定所有随机初始化种群均符合均匀概率分布。设参数变量的边界条件为 $x_j^{(L)}<x_j<x_j^{(U)}$，则

$$x_{ji,0}=\mathrm{rand}[0,1]\times\left(x_j^{(U)}-x_j^{(L)}\right)+x_j^{(L)}$$

其中，$j$=1, 3,⋯, $D$。

2) 变异：对于每个目标向量 $x_{i,C}$, 定义其变异向量为

$$v_{i,G+1}=x_{r_1,G}+F\times\left(x_{r_2,G}-x_{r_3,G}\right)$$

其中，参数 $r_1$、$r_2$、$r_3$ 互不相同，同时有 $NP\geqslant 4$，变异算子 $F\in[0,2]$。

3) 交叉：定义实验向量

$$u_{i,G+1}=\left(u_{1i,G+1},u_{2i,G+1},\cdots,u_{Di,G+1}\right)$$

$$u_{ji,G+1}=\begin{cases}u_{ji,G+1},\mathrm{if}\left(\mathrm{rand}b(j)\leqslant CR\right)\mathrm{or}\left(j=\mathrm{rnbr}(i)\right)\\ x_{ji,G+1},\mathrm{if}\left(\mathrm{rand}b(j)>CR\right)\mathrm{and}\left(j\neq\mathrm{rnbr}(i)\right)\end{cases}$$

其中，rand$b$($j$)用于在[0, 1]之间随机产生第 $j$ 个估计值，rnbr($i$)∈ 1, 2,⋯, $D$。$CR$ 为交叉算法，取值范围为[0,1]。

4) 选择：决定实验向量 $u_{i,G+1}$ 是否能够成为下一代中的会员。差分演化算法按照贪婪准则将实验向量与当前种群中的目标向量 $x_{i,G}$ 进行比较，并选择更优的向量作为下一代种群中的向量。函数表示为

$$\Delta f=f\left(u_{i,G+1}\right)-f\left(x_{i,G}\right)$$

即如 $\Delta f$>0，则接收新的实验向量，否则不接受。

5) 边界条件的处理。在存在边界约束的环境中，差分演化算法将不符合边界约束的新个体在可行域内随机产生的参数向量代替，即若 $x_{i,G}<x_j^{(L)}$或 $x_{i,G}<x_j^{(U)}$，则

$$u_{ji,G+1}=\mathrm{rand}[0,1]\times\left(x_j^{(U)}-x_j^{(L)}\right)+x_j^{(L)}$$

差分演化算法弥补了遗传算法编码繁琐、实现复杂的缺陷，但其搜索进程缓慢且易于“早熟”。出现上述缺陷的关键在于差分演化算法是在“优胜劣汰”准则下进行选择操作的，这也就决定了算法后期的收敛速度较慢，甚至有可能陷入局部最优。

为了实现硬件重构中的快速演化，本文在差分演化过程中引入局部优化机制，提出一种改进的差分演化（MDE, modified differential evolution）算法。该算法基于 Metropolis 准则进行选择操作，从而能够获得更优的收敛速度和全局优化效率。MDE 算法步骤描述如下。

**步骤 1**　初始化算法参数：确定种群大小 *NP*，变异因子 *F* 及 *CR*，初始化 Metropolis 准则参数 *T* 和降温控制参数 *k*，设定迭代次数 *D*，令 *h*=0。

**步骤 2**　初始化可行解空间，随机产生 *N* 个可行解。

**步骤 3**　执行交叉、变异操作。

**步骤 4**　执行选择操作。执行过程中，随机产生扰动 $\Delta x$，得到新节点 $x^*=x+\Delta x$，如果该节点优于 $x$，则把该节点作为下一次迭代的可行节点；否则计算出该新节点的接受概率 $P=\exp(-\Delta f/T)$，并产生一个[0,1]区间上的伪随机数 $r$，若 $P\geqslant r$，则接受新节点作为下一次迭代的初始点；否则放弃新节点，仍选择原节点作为下一次迭代的初始点。

**步骤 5**　执行降温操作 $T=T\times k$，令 $h=h+1$。

**步骤 6**　如 *h* 等于 *D*，则跳回步骤 3，否则终止算法。

## 4　仿真实验与性能评价

为了进一步验证 MDE 算法的性能，本文选择 2 种典型函数对其进行仿真实验。

**函数 1**　（函数最优值为 $f^*=-78.332\ 3$）

$$\min f(x)=\frac{1}{100}\sum_{i=1}^{100}\left(x_i^4-16x_i^2+5x_i\right),-10\leqslant x_i\leqslant 10$$

**函数 2**　（函数最优值为 $f^*=-0.998$）

$$\min f(x)=0.002-\sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^{2}\left(x_i-a_{ij}\right)^6},$$

$$-65.536\leqslant x_i\leqslant 65.536,i=1,2$$

其中，$a_{1,j}$={−32, −16, 0, 16, 32, −32, −16, 0, 16, 32, −32, −16, 0, 16, 32, −32, −16, 0, 16, 32, −32, −16, 0, 16, 32}，$a_{2,j}$={−32, −32, −32, −32, −32, −16, −16, −16, −16, −16, 0, 0, 0, 0, 0, 16, 16, 16, 16, 16, 32, 32, 32, 32, 32}。

给定初始参数 $T$=5 000，$k$=0.8，$NP$=12，$CR$=0.3，$F$=0.5，迭代次数为 100 次。表 2 分别给出了函数 1 和函数 2 运行 6 次时的结果对比。

**表 2　函数 1 和函数 2 的仿真结果（运行 6 次）**

| 函数 1 | | 函数 2 | |
|---|---|---|---|
| DE | MDE | DE | MDE |
| −78.326 1 | −78.332 1 | −0.996 | −0.998 |
| −78.332 6 | −78.333 8 | −0.998 | −0.998 |
| −78.332 1 | −78.332 3 | −0.998 | −0.997 |
| −32.296 3 | −78.332 8 | −0.325 | −0.994 |
| −78.338 1 | −78.329 9 | −0.991 | −0.998 |
| −78.332 6 | −78.332 2 | −0.995 | −0.996 |

对上述仿真结果采用方差距离比较法进行验证。假设函数的最优值为 $f^*$，迭代次数为 $K$，DE 算法第 $k$ 次迭代结果为 $f_{1k}$，MDE 算法第 $k$ 次迭代结果为 $f_{2k}$，DE 算法最优值的方差距离之和为 $D_1=\sum_{k=1}^{K}d_{1k}=\sum_{k=1}^{K}\left(f_{1k}-f^*\right)^2$，MDE 算法最优值的方差距离之和为 $D_2=\sum_{k=1}^{K}d_{2k}=\sum_{k=1}^{K}\left(f_{2k}-f^*\right)^2$。

计算函数 1 和函数 2 6 次迭代的 $D_1$ 和 $D_2$ 值，其结果都为 $D_1$ 大于 $D_2$。对函数 1 进行 100 次迭代，仿真结果如图 6 所示，DE 的方差距离大于 MDE 的方差距离。

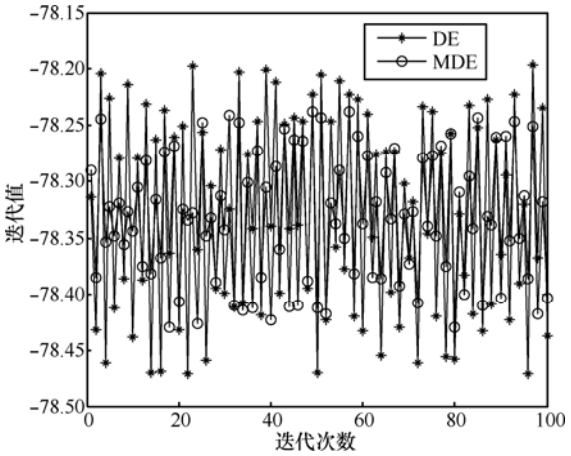


图 6　DE 与 MDE 的方差距离比较

由上述仿真结果可以看出，DE 算法依然存在未收敛到最优解空间的情况，这就是由于 DE 算法存在陷入局部最优解的缺点，而仿真过程中 MDE 算法并未出现上述情况，说明 MDE 算法改进了差分演化算法陷入局部最优解的缺点，更加逼近实际最优解。

## 5　结束语

硬件重构技术为构建柔性网络提供了新的研究方向。当现有网络服务发生变化时，基于硬件重构技术的柔性网络能够采用本文所提出的演化硬件重构编码方案及演化算法，实现服务能力平滑、快速的演变及升级，从而高效地利用网络资源，提高网络的多样服务能力。

演化硬件以可编程逻辑器件(PLD, programmable logic device)为平台，针对其可重复编程重复配置的特点，将下载到 PLD 上的二进制配置位串作为遗传算法的染色体，模拟生物种群的演化过程来搜寻最优的配置，自适应地实现配置 PLD 上可重构逻辑资源并得到所需功能的硬件电路。目前，Xilinx 公司已有多款 FPGA 不仅可以支持全局重构，还能够提供部分重构 (PR, partial reconfiguration )[17]功能。PR 是指在不影响系统总体运行的情况下，对系统的一部分结构进行重新配置以实现新的电路功能，如 Virtex™-II、Virtex™-II Pro/X、Virtex™-4/ FX/LX/SX 系列的 FPGA 都可支持电路的部分重构。这些芯片提供的 PR 功能可以分为 2 种模式：基于模块(module-based)的 PR 和基于差异(difference- based)的 PR[18]。基于模块的 PR 利用总线宏(bus macro)技术保证重构后模块间的正常连接，而基于差异的 PR 通过比较重构前后电路功能的差异进行重构。

本文在基于 SRAM 结构的 FPGA 平台上，提出一种基于 LUT 结构的二维映射函数增量染色体编码方案 PMFICC，该编码方案可利用 Xilinx 公司 FPGA 支持基于差异的 PR 功能，对比重构前后 2 种电路结构 SP 编码和 CP 编码的差异，即重构前后 2 种 Bitstream 文件的差异，通过改进的差分演化算法 MDE 将前后 2 代染色体的差异引入到演化过程中，进而得到新的最优染色体，实现电路的动态重构，快速有效地支持系统的新功能。仿真结果验证了其性能。

## 参考文献：

[1] 姚爱红, 张国印, 关琳. 基于动态可重构 FPGA 的自演化硬件概述[J]. 智能系统学报, 2008, 3(5): 436-440.
YAO A H, ZHANG G Y, GUAN L. A survey of dynamically and partially reconfigurable FPGA-based self-evolvable hardware[J]. CAAI Transactions on Intelligent Systems, 2008, 3(5): 436-440.

[2] 兰巨龙. 可重构信息通信基础网络体系研究.国家重点基础研究发展计划（973 计划）项目计划任务书[R]. 2011.
LAN J L. Reconfigurable Basic Network of Information and

Communication System[R]. National Key Basic Research and Development Program Project planTask(973 Project)[R]. 2011.

[3] 李玉峰, 邱菡, 兰巨龙. 可重构路由器研究的现状与展望[J]. 中国工程科学, 2008, 10(7): 83-87.
LI Y F, QIU H, LAN J L. The study of reconfigurable router status and prospect[J]. China Engineering Science, 2008, 10(7): 83-87.

[4] 姚睿, 于盛林, 王友仁等. 采用主流 FPGA 的数字电路在线生长进化方法[J]. 南京航空航天大学学报, 2007, 39(5): 583-585.
YAO R, YU S L, WANG Y R, *et al*. Online growing evolution and evaluation approach based on mainstream FPGA[J]. Journal of Nanjing University of Aeronautics & Astronautics, 2007, 39(5): 583-585.

[5] KAUFMANN P, PLESSL C, PLATZNER M. Evocaches: application-specific adaptation of cache mappings[A]. NASA/ESA Conference on Adaptive Hardware and Systems[C]. 2009. 13-17.

[6] YANG H Q, CHEN L G, LIU S T, *et al*. A flexible bit-stream level evolvable hardware platform based on FPGA[A]. NASA/ESA Conference on Adaptive Hardware and Systems[C]. 2009. 51-58.

[7] LOHN J, HORNBY C. Evolvable hardware: using evolutionary computation to design and optimize hardware systems[J]. IEEE Computational Intelligence Magazine, 2006, 1(1):19-27.

[8] MAO N, MINORU W. A sixteen-context dynamic optically reconfigurable gate array[A]. NASA/ESA Conference on Adaptive Hardware and Systems[C]. 2009. 121-125.

[9] SHANG L H, ZHOU M, *et al*. A new application-tuned processor architecture for high-performance reconfigurable computing[A]. NASA/ESA Conference on Adaptive Hardware and Systems[C]. 2009. 139-145.

[10] 李敏强, 寇纪淞, 林丹等. 遗传算法的基本理论与应用[M]. 北京: 科学出版社, 2002.
LI M Q, KOU J S, LIN D, *et al*. Genetic Algorithm is the Basic Theory and the Application[M]. Beijing: Science Press, 2002.

[11] XILINX I. Virtex-II platform FPGA user guide[EB/OL]. http://china. Xilinx.com/support/documentation/user_guides/ug002.pdf.

[12] AROSTEGUI M, SANCHEZ E， CABESTANY J. An in-system routing strategy for evolvable hardware programmable platforms[A]. Proc of the 3rd NASA/DoD, Long Beach[C]. 2001. 157-166.

[13] OREIFEJ R, AIHADDAD R, HENG T, *et al*. Layered approach to intrinsic evolvable hardware using direct bit-stream manipulation of virtex-II prodevices[A]. International Conference on Field Programmable Logic and Applications(FPL 2007)[C]. Amsterdam, 2007. 299-304.

[14] MUSRRAT A, PANT M, ABRAHAM A. Simplex differential evolution[J]. Acta Polytechnica Hungarica, 2009, 6(5): 95-102.

[15] MUSRRAT A, PANT M, *et al*. A modified differential evolution algorithm and its application to engineering problems[A]. International Conference of Soft Computing and Pattern Recognition[C]. 2009. 196-200.

[16] YANG Z Y, TANG K, YAO X. Large scale evolutionary optimization using cooperative coevolution[A]. Information Sciences[C]. 2008. 2986-2991.

[17] XILINX C. XILINX development system reference guide 8. li[EB/OL]. http://toolbox. xilinx. com/docsan/ xilinx8/ books/ docs/ dev.pdf, Xilinx data sheet, 2007.

[18] XILINX C. XAPP290: two flows for partial reconfiguration: module based or difference based[EB/OL]. http://www. xilinx. com, 2004.

**作者简介：**

**王婷**（1982-），女，山东兖州人，国家数字交换系统工程技术研究中心博士生、工程师，主要研究方向为可重构硬件重构理论与技术、可重构交换理论与技术。

**兰巨龙**（1962-），男，河北张家口人，博士，国家数字交换系统工程技术研究中心教授、博士生导师，主要研究方向为可重构网络理论与技术。

**邬钧霆**（1979-），男，安徽金寨人，博士，国家数字交换系统工程技术研究中心工程师，主要研究方向为可重构路由器服务质量技术。