

基于矩阵表示的 CMOL 电路容错映射

徐鹏飞, 夏银水*, 查晓婧, 顾贤贵

(宁波大学 信息科学与工程学院, 浙江 宁波 315211)

摘要: 针对存在缺陷的 CMOS/纳米分子混合(CMOS/nanowire/MOLeclular hybrid, CMOL)电路的单元容错映射问题, 提出一种基于矩阵表示的 CMOL 电路容错映射方法. 首先, 将逻辑电路和 CMOL 电路建模为矩阵表示; 然后采用文化基因(memetic)算法进行矩阵间可匹配字符的搜索, 采用小矩阵元值优先匹配的策略完成单元缺陷容忍映射. ISCAS 测试电路的实验结果表明, 与已有方法相比, 本文方法在求解速度上有 36.98%的提升.

关键词: CMOL; 单元容错映射; 矩阵表示; memetic 算法

中图分类号: TP391.41

文献标志码: A

文章编号: 1001-5132 (2021) 02-0001-08

随着晶体管特征尺寸不断缩小至纳米级别, 由于微观世界的寄生效应和量子效应^[1]的影响, 传统集成电路的设计方法不适合当前发展的需求. 人们寄希望于新型的纳米器件和纳米混合电路^[2]能够延续集成电路的发展. CMOS/纳米分子混合(CMOS/nanowire/MOLeclular hybrid, CMOL)电路^[3]结合纳米技术和传统的 CMOS 工艺, 既具有纳米技术的高集成密度, 又具有 CMOS 丰富的逻辑功能, 被认为是后 CMOS 时代最具有前途的替代技术之一.

自从 Likharev 等提出 CMOL 电路结构以来, 国内外研究人员对于 CMOL 电路的研究主要集中在两个方面: 一是不考虑 CMOL 电路中远高于传统 CMOS 器件的缺陷率进行的无缺陷映射, 即 CMOL 单元无缺陷映射^[4-7]; 二是考虑缺陷影响的容错映射, 即 CMOL 单元容忍映射^[8]. 在 CMOL 单元无缺陷映射领域已经有较为成熟的映射算法, 如结合拉格朗日算子的文化基因算法^[4]、模拟进化算法^[5]、禁忌搜索算法^[6]和布谷鸟搜索算法^[7]. 在 CMOL 单元容忍映射领域, Strukov 等^[9]开展了 CMOL 电路缺陷容忍映射研究, 并以 32-bit Kogge-Stone 加法器和 64-bit Full-Crossbar 电路为例, 提出

了可重构缺陷容忍映射思想. Hung 等^[10]针对多缺陷类型, 将 CMOL 电路容忍问题描述为可满足性问题, 将容错方法转化为布尔约束, 但该方法仅能解决小规模电路, 且速度较慢. 陈定亨等^[11]利用最小逻辑单元和纳米二极管协同检测加速提取 CMOL 阵列中无缺陷单元所在区域, 将容忍问题简化为无缺陷映射问题, 但该方法需要舍弃大量无缺陷单元造成可利用资源减少. Sait 等^[12]运用模拟进化算法重构满足连通域约束的解, 虽然能够解决大规模电路映射问题, 但将整个 CMOL 电路作为求解对象, 存在求解速度慢、解质量较差的问题. Chen 等^[13]分析纳米二极管常连缺陷特性, 指出常连缺陷传播的问题, 并使用反相器对的互补信号抑制常连缺陷传播. 查晓婧等^[14]提出以缺陷数递减的方式对 CMOL 电路进行动态分层, 再逐层进行缺陷容忍, 降低电路映射难度, 但是初始解的质量对分层后的局部搜索影响较大.

综上所述, 已有的研究总是将逻辑电路和 CMOL 电路建模为有向图, 将 CMOL 电路的缺陷容忍映射问题转化为有向图的匹配问题. 由于在有向图表示法中的每一节点、每一有向连接边一一校对匹配效率较低, 所以这些方法的缺陷容忍映

收稿日期: 2020-08-10.

宁波大学学报(理工版)网址: <http://journal.g.nbu.edu.cn/>

基金项目: 国家自然科学基金(61571248).

第一作者: 徐鹏飞(1995-), 男, 湖北钟祥人, 在读硕士研究生, 主要研究方向: 逻辑综合与优化. E-mail: 1079290675@qq.com

*通信作者: 夏银水(1963-), 男, 浙江余姚人, 研究员, 主要研究方向: 集成电路逻辑综合与优化. E-mail: xiayinshui@nbu.edu.cn

射速率较低^[15]。鉴于此,本文提出一种基于矩阵表示的CMOL电路容错映射方法。首先将逻辑电路和CMOL电路建模为矩阵表示;然后采用以遗传算法为主,模拟退火算法为辅的文化基因算法^[16]进行矩阵间可匹配字符的搜索,采用小矩阵元值优先匹配的策略完成单元缺陷容忍映射。与现有方法相比,本文方法在求解速度上有较为明显的优势。

1 研究背景

1.1 CMOL 电路结构

CMOL电路是一种结合CMOS工艺和纳米分子器件的新型类FPGA电路结构。图1(a)为其剖面图,CMOL电路结构类似三明治的层叠结构,上面两层是纳米线,其通过高度不同的接口引脚与CMOS反相器相连,其中顶层是输出纳米线层,底层是输入纳米线层,两层间的纳米线相互垂直,同层间的纳米线相互平行。在输入纳米线与输出纳米线的交叉处是纳米二极管,通过控制纳米二极管的导通,可以实现信号传输。图1(b)是其俯视图,为保证每条纳米线只与一个接口引脚相连,纳米线层与CMOS反相器网格单元呈一定的倾角 $\alpha = \arcsin(F_{\text{nano}} / \beta F_{\text{cmos}})$ 排布,同时纳米线周期性断裂,其中, F_{nano} 是相邻两根纳米线间距的1/2, F_{cmos} 是相邻两个CMOS反相器间距的1/2, β 是与CMOS反相器复杂度相关的零量纲因子。由于纳米线周期性断裂使得任意CMOL单元只与其周围 $N_{\text{cell}} = 2R(R-1) - 1$ 个单元相连接,形成了该单元连通域,其中, R 表示连通域半径,传统的连通域半径以曼哈顿距离(若两点坐标分别为 $(x_1, y_1), (x_2, y_2)$,则两点间曼哈顿距离为 $|x_1 - x_2| + |y_1 - y_2|$)来计算。

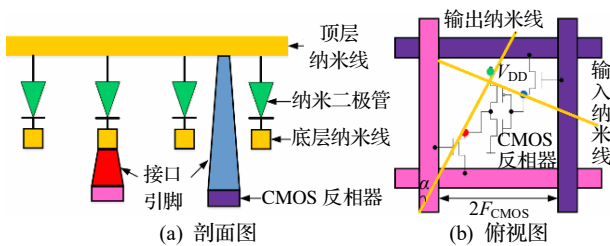


图1 CMOL电路结构^[1]

1.2 CMOL 单元映射及约束

任意逻辑电路可抽象为有向图 $G = (V, E)$,其

中 V 是逻辑门抽象成顶点的集合, $E = V \times V$ 为顶点间连接边的集合。相应地,根据CMOL电路的结构特点,可将任意的CMOL电路也抽象成有向图 $\Phi = (C, \psi)$,其中 C 表示CMOL单元集合, ψ 表示CMOL单元间连接线集合。

CMOL电路单元映射可这样描述:对于一个给定的逻辑电路 G 和一个CMOL电路 Φ ,目标是找到一种映射关系 P 使 G 中所有电路节点都映射到CMOL电路 Φ 上,

$$P: G \rightarrow \Phi,$$

并满足以下约束:

$$\begin{aligned} \forall g_i \in \text{Fin}(g_j): M_{\text{dis}}(c(g_i), c(g_j)) &\leq \\ R \cap c(g_i) &\neq c(g_j), \end{aligned} \quad (1)$$

式中: $\text{Fin}(g_j)$ 表示逻辑电路节点 g_j 的扇入节点; $c(g_i)$ 表示逻辑电路节点 g_i 所映射的CMOL单元; $M_{\text{dis}}(c(g_i), c(g_j))$ 表示逻辑电路节点 g_i, g_j 所映射单元 $c(g_i), c(g_j)$ 之间的曼哈顿距离。式(1)说明,逻辑电路中具有扇入/扇出关系的两节点所映射单元必须满足连通域约束,且一个CMOL单元只能被一个电路节点所映射。

在CMOL电路中,纳米线上实现“或”逻辑,CMOS反相器实现“非”操作,故CMOL电路的基本逻辑功能是“或非/非”。图2所示为一个双输入/单输出的或非门在 3×3 大小的CMOL电路映射示意。将输入端信号 i_1, i_2 分别映射于单元2、6上,节点 g 映射于单元7上,通过激活纳米二极管 n_1, n_2 ,单元7输出实现 $o_1 = i_1 + i_2$ 逻辑功能。

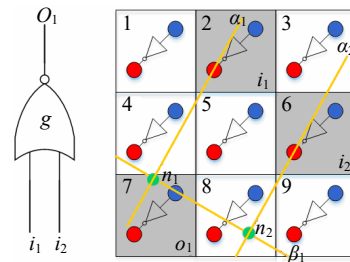


图2 CMOL电路单元映射

1.3 CMOL 单元常开缺陷类型

CMOL电路在制造过程中,由于纳米器件尺寸极小且当前采用自底向上的自组装技术难以精确控制引脚位置,使得CMOL电路不可避免地存在高缺陷率问题。缺陷类型大体上可分为两类,即常连缺陷和常开缺陷。对最常出现的常开缺陷类

型分析如下:

(1)纳米二极管常开. 由于纳米二极管一直处于断开状态, 连接在纳米二极管上的两层纳米线不能实现信号传输, 使得通过该纳米二极管连接的两个 CMOL 单元无法正常通信.

(2)纳米线非周期性断裂. 由于纳米线在非周期性断点处发生不规则断裂, 造成其所连接单元连通域减少.

(3)接口引脚常开. 由于在自底向上的自组装过程中未能实现接口引脚与纳米线精准对齐, 造成接口引脚两端连接的纳米线层与底层 CMOS 单元始终处于断路状态.

图3所示为上述缺陷类型说明示意, 记深灰色单元为 c , 其原始输出连通域为黑色实线框定区域. 由于上述缺陷的存在, 单元 c 的连通域范围和可通信单元数均发生改变. 纳米二极管常开仅导致单元 c 与单元2之间的信号通路被阻断, 但单元 c 和单元2仍然可用. 界面引脚常开致使单元1与其连通域内的所有单元之间的信号通路均被阻断, 因此单元1无法正常使用, 需被舍弃. 纳米线非周期性断裂致使单元 c 输出纳米线断裂成 α_1 和 α_2 两段, 在这种情况下, 单元 c 的输出连通域不再包括单元3、4、5, 换言之, 单元3、4、5在单元 c 的输出连通域外.

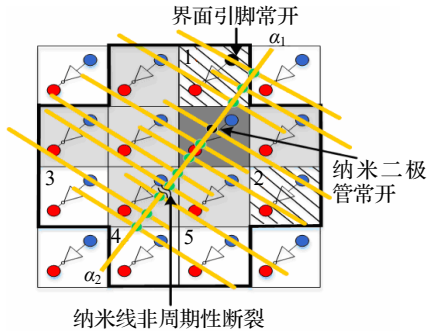


图3 缺陷类型示意

2 CMOL 缺陷容忍映射

CMOL 缺陷容忍映射问题可描述为: 给定一个逻辑电路和存在缺陷的 CMOL 电路, 通过缺陷容忍实现映射后电路的逻辑功能正确. 传统的缺陷容忍方法总是将逻辑电路和 CMOL 电路建模为有向图, 将缺陷容忍映射问题转换为子图同构问题^[17], 但是其匹配效率低下. 为此, 提出将单元配

置的搜索问题转换为矩阵元素匹配问题, 从而实现高效快速的 CMOL 电路缺陷容忍映射.

2.1 逻辑矩阵(LM)的建立

根据逻辑电路中各节点之间的扇入扇出关系, 将逻辑电路构造为矩阵表示的形式. 如图4所示, 含有11个节点的或非门逻辑电路转化为 11×11 规格的逻辑矩阵, 其中矩阵的行号和列号均表示逻辑电路中门节点的编号. LM中 g_1 行和 g_2 列交叉处的矩阵元 $E_L(g_1, g_2)$ 记录节点 g_1 与节点 g_2 的连接关系, 取值情况如下:

$$E_L(g_1, g_2) = \begin{cases} 1, & g_1 \in \text{Fin}(g_2), \\ 0, & g_1 \notin \text{Fin}(g_2), \end{cases} \quad (2)$$

式中, $g_1 \in \text{Fin}(g_2)$ 表示节点 g_1 是节点 g_2 的扇入节点. 以图4(a)中节点7为例, 节点3是节点7的一个扇入节点, 则 $E_L(3, 7) = 1$, 对应图4(b)中第3行与第7列交叉处的值为1; 节点8不是节点7的扇入节点, 则 $E_L(8, 7) = 0$, 即图4(b)中第8行与第7列交叉处的值为0.

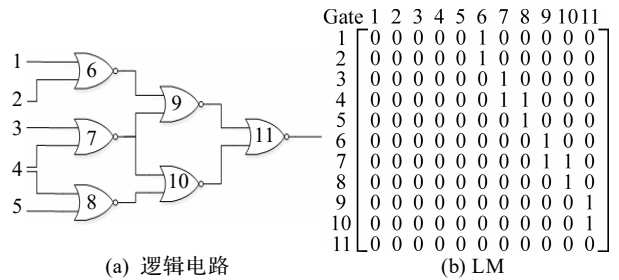


图4 LM转化示意

2.2 CMOL 矩阵(CM)的建立

根据 CMOL 电路连通域约束和电路缺陷分布情况, 将 CMOL 电路构造为矩阵表示. 图5(a)所示为含有25个单元的存在常开缺陷的 CMOL 电路. 其中, 左上角数字表示该单元的编号; 黑色框线内的区域为单元13原始输出连通域, 单元13的输出纳米线非周期性断裂成L1、L2两部分, 故单元11、17、23不在其连通域范围; 单元13与单元9间的纳米二极管①以及单元13与单元7间的纳米二极管②存在常开缺陷; 界面接口引脚常开缺陷致使单元15、24与其连通域范围内的其他单元均无法通信, 因此, 需舍弃该单元. 图5(b)是根据图5(a)构造的CM, 其中矩阵的行列号均表示CMOL单元编号, 矩阵中的每个矩阵元值表示两个单元之间的曼哈顿距离.

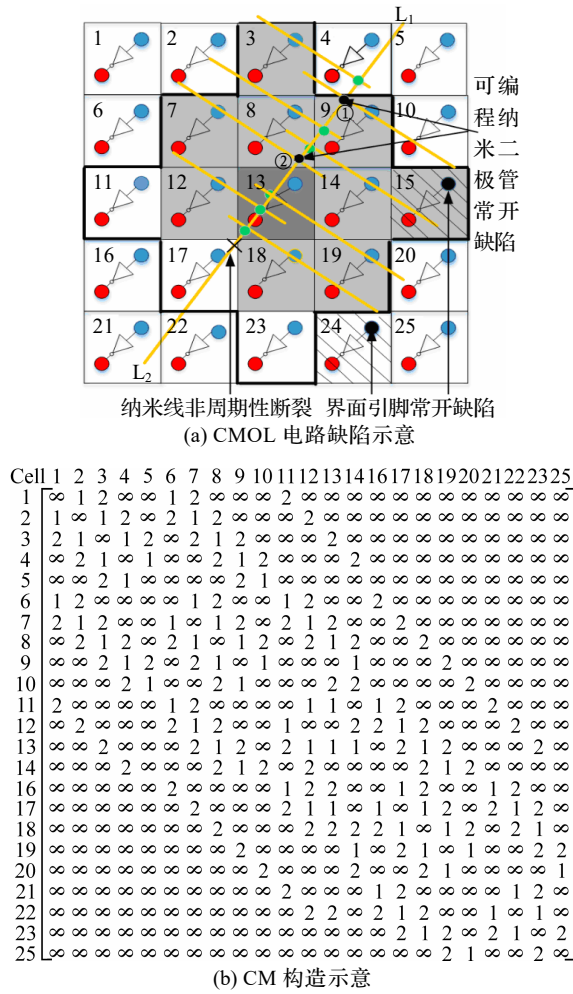


图5 CMOL 缺陷电路及其矩阵表示

记单元 c_1 输出纳米线与单元 c_2 输入纳米线之间的纳米二极管为 $d(c_1, c_2)$; 其中 $d(c_1, c_2) = 0$ 表示该纳米二极管功能正常, $d(c_1, c_2) = 1$ 表示该纳米二极管发生常开缺陷, 则 $d(13, 3) = 0$, $d(13, 7) = 1$, $d(13, 8) = 0$, $d(13, 9) = 1$, $d(13, 12) = 0$, $d(13, 14) = 0$, $d(13, 18) = 0$, $d(13, 19) = 0$. 记单元 c 的输出连通域为 $R(c)$, 以连通域半径为距单元 c 两个曼哈顿距离为例, 则 $R(13) = \{3, 7, 8, 9, 11, 12, 14, 15, 17, 18, 19, 23\}$. 记 CM 中 c_1 行和 c_2 列交叉处的矩阵元为 $E_C(c_1, c_2)$, 取值情况如下:

$$E_C(c_1, c_2) = \begin{cases} M_{dis}, c_2 \in R(c_1) \text{ 与 } d(c_1, c_2) = 0, \\ \infty, c_1 \notin R(c_2) \text{ 或 } d(c_1, c_2) = 1, \end{cases} \quad (3)$$

式中: M_{dis} 为单元 c_1 与单元 c_2 的曼哈顿距离; ∞ 表示信号在单元 c_1 与单元 c_2 间不能传输.

当纳米线非周期性断裂时, 该单元连通域减小, 使得原本在其连通域的单元不在其连通域内, 此种情况可等效为纳米二极管常开缺陷, 即

$E_C(13, 11) = \infty$, $E_C(13, 17) = \infty$, $E_C(13, 23) = \infty$. 本文采用 mR 来取代 ∞ , 其中, m 为 LM 中 $E_L = 1$ 的矩阵元数目之和.

当单元发生界面接口引脚常开缺陷时, 该单元无法与其他所有的正常单元实现通信, 不可将电路节点映射于该缺陷单元上, 故在构造 CM 时, 应该剔除该缺陷单元相关矩阵信息.

2.3 矩阵匹配策略

矩阵匹配的过程实际上是考虑连通域约束的单元容错映射问题. 具体描述为: 若逻辑电路中节点 g_1 是节点 g_2 的扇入, 即 $E_L(g_1, g_2) = 1$, 则节点 g_1 和节点 g_2 所映射的两个 CMOL 单元 i 和 j 需满足连通域约束且单元不存在缺陷, 即 $E_C(i, j) = M_{dis} (M_{dis} < R)$. 反之, 若节点 g_1 不是节点 g_2 的扇入, 即 $E_L(g_1, g_2) = 0$, 则节点 g_1 和节点 g_2 可以映射到任意的两个 CMOL 单元 i 和 j 上.

在矩阵匹配过程中, 具体的表现形式为 LM 中 $E_L = 1$ 的矩阵元对应 CM 中矩阵元值应为非无穷大, $E_L = 0$ 的矩阵元则可以对应 CM 中的任意矩阵元. 令符号 “ $\rightarrow$ ” 表示两矩阵间的矩阵元对应关系, M_{res} 表示 LM 中矩阵元 $E_L = 1$ 的匹配结果, M_{res} 可用下式来定义:

$$M_{res} = M_{dis} / R, \quad E_L = 1 \rightarrow E_C = M_{dis}, \quad (4)$$

$M_{res} < 1$ 表示矩阵元 LE 与矩阵元 CE 匹配成功; 否则, 匹配失败. LM 与 CM 匹配成功的标志为: LM 中所有 $E_L = 1$ 的矩阵元均与 CM 中的矩阵元 E_C 匹配成功.

由于 CMOL 单元数往往多于映射电路的节点数, 即 CM 的行列数大于 LM. 矩阵的匹配过程中, 在 CM 中截取一个与 LM 相同规模的子矩阵, 通过交换子矩阵的行列顺序与 LM 匹配即可, 并且规定截取的子矩阵总是位于 CM 的左上方.

2.4 算法实现

由于矩阵的存储信息量庞大, 单纯地采用遗传算法(GA)来进行矩阵间矩阵元的匹配会存在搜索空间大、收敛速度慢、容易陷入局部最优解等问题. 因此, 采用一种基于种群的全局搜索和基于个体的局部启发式搜索相结合的 memetic 算法(MA). MA 是解决复杂问题的高效启发式算法. 本文将 GA 作为全局搜索算法, 辅以模拟退火算法(SA)进行局部搜索, 可充分发挥 GA 全局搜索的优点以及

SA 局部搜索的能力, 不仅收敛快, 还能在合理的时间得到高质量的解. 其伪代码见算法 1. 首先随机生成规模大小为 P_{size} 的初始种群, 种群中的每一个个体都代表一种可能的解; 然后通过适应值函数计算种群中每个个体的适应值; 根据适应值高低, 选择两个亲代, 其中适应值越低表示适应能力越强, 被选中的机会就越高; 对选中的两个亲代进行交叉和变异操作; 所有的个体(包括亲代和子代)根据适应值升序排序, 然后选择适应值最优的 P_{size} 个个体, 其他的个体被拒绝, 以保持种群规模不变, 进行下一轮迭代. 在每次迭代中运用基于 SA 的局部搜索策略, 使子代能够产生更好的种群. 当 LM 与 CM 匹配成功或者迭代次数达到上限时算法停止.

算法 1 MA 算法

Step1: 生成初始种群 initialize_population();

Step2: 计算个体适应值 calculate_fitness_score();

Step3: 选择亲代 generate_parents();

Step4: 交叉操作 crossover();

Step5: 变异操作 mutation();

Step6: 依据适应值选择存活个体;

Step7: 局部搜索 local_search();

Step8: 若矩阵匹配成功或者迭代次数达到上限, 算法终止; 否则, 重复 Step3~Step7.

2.4.1 适应值函数

使用随机算法生成初始种群, 种群中的每个个体代表着一个 CM, 每个 CM 代表着一个解. 为了减小单元间的线长, 对于 CM 中不同的矩阵元值施加不同程度的惩罚, 矩阵元值越大, 其惩罚系数越大, 因此对 LM 中每一个 $E_L = 1$ 的矩阵元都需要进行评估, 评估函数定义如下:

$$F_{\text{fitness}}(i) = E_C(i) / R, \quad (5)$$

式中: $F_{\text{fitness}}(i)$ 表示 LM 中第 i 个 $E_L = 1$ 的矩阵元评估结果; $E_C(i)$ 表示 CM 中与之进行匹配的矩阵元. F_{fitness} 越小意味着当前 LM 中的矩阵元 E_L 匹配效果越好.

在 MA 中, 适应值函数用来评价种群中每个个体的适应能力, 该函数为下一代选择最优 P_{size} 个个体组成种群提供了依据, 适应值函数反映解的质量. 因此, 适应值函数定义如下:

$$F_{\text{fitness_score}} = \sum_{i=1}^m F_{\text{fitness}}(i), \quad (6)$$

$F_{\text{fitness_score}}$ 值越小, 表示解的质量越好. LM 与 CM 匹配成功的标志是 $F_{\text{fitness_score}} < m$.

2.4.2 交叉、变异

交叉、变异是遗传算法的关键步骤, 由于矩阵的行排列与列排列总是相同的, 只针对行排列来进行遗传操作. 当一轮操作结束后将相应的列进行相同的交换操作, 由于每个 CMOL 单元能且只能被一个电路节点所映射, 故 CMOL 单元编号不可重复, 因此采用部分交叉的策略生成子代, 避免因双切点交叉导致单元重复问题. 具体的交叉过程如图 6 所示, 每一个数字表示 CM 行编号, 亲代各自将内部基因按照箭头所指方向, 即 16-12, 2-11, 10-8, 15-15, 7-1 的交换关系, 进行位置互换得到子代. 变异发生在交叉之后, 变异操作能够重现在选择步骤中丢失的性状, 从而维持种群的多样性并且避免算法陷入局部最优. 变异时, 会随机选择种群中需要变异的个体基因进行位置交换.

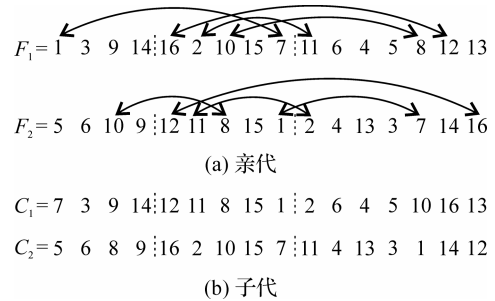


图 6 交叉操作

需要注意的是, 由于 LM 尺寸($N_L \times N_L$)远小于 CM 尺寸($N_C \times N_C$), 在进行 LM 与 CM 匹配时, 所截取子矩阵仅位于 CM 的前 N_L 行, 故亲代的后 $N_C - N_L$ 个基因, 即 CM 的后 $N_C - N_L$ 行的排列不会影响单元映射的结果. 因此在进行交叉操作时, 随机选取两个基因的位置应落在亲代的前 N_L 个基因位上, 即前 N_L 行上. 在进行变异操作时, 随机选择的一对基因总是一个位于前 N_L 个基因位上, 另一个位于后 $N_C - N_L$ 个基因位上.

2.4.3 基于模拟退火算法的局部搜索策略

传统的 GA 收敛速度慢, 但是如果在 GA 的循环迭代中加入基于 SA 的局部搜索策略, 会使种群的个体迅速得到改进, 加快收敛速度并且得到更

高质量的解. 算法随机在精英策略所选择的 P_{size} 个个体中选择 $P_{size} \times R_L$ 个个体进行局部搜索(R_L 为局部搜索概率), 对被选中的个体进行基因交换, 比较交换前后的适应值. 如果适应值有改进, 表示此次交换后所得解的质量更好, 则接受此次交换, 否则以玻尔兹曼概率 ($e^{-\Delta s/T}$) 有选择地接受使适应值上升的交换, 以提升算法对解空间的搜索能力. 局部搜索策略的伪代码见算法 2.

算法 2 基于模拟退火算法的局部搜索

Step1: 读入个体 p 并计算其初始适应值 $init_score$.

Step2: 对个体 p 随机交换基因位置, 计算交换后的适应值 new_score .

Step3: 计算交换前后的适应值差值 $\Delta score = new_score - init_score$.

Step4: 若 $\Delta score < 0$ 则接受交换; 否则产生一个 $[0,1]$ 间的随机数 $random$, 若 $random < e^{-\Delta score/T}$ 则接受交换, 其中 T 为当前温度; 其余情况均拒绝并撤销交换.

Step5: 更新内循环次数.

Step6: 若内部循环次数上限达到 num_inner_loop , 终止内循环. 更新温度, 否则, 重复 Step2~Step5.

Step7: 若温度 T 小于最低设定的温度 T_{old} , 算法终止; 否则, 重复 Step2~Step6.

在 SA 中温度和冷却策略对于算法的成功实施至关重要. 初始温度和模拟退火算法所做的基因交换的接受率有关, 初始温度越高则接受率越高, 反之越低. 由于全局搜索采用 GA 来实现, SA 只是用来进行局部搜索, 因此初始温度的设定不应太高, 根据实验将初始温度设定在 0.2°C 比较合适. 随着算法的执行, 温度会逐渐降低以限制基因交换的接受率. 温度按照下式进行更新:

$$T_{new} = b \times T_{old}, \quad (7)$$

式中, b 为冷却因子, 与基因交换的接受率 a 有关,

$$b = \begin{cases} 0.6, & a > 0.4, \\ 0.9, & 0.2 < a \leq 0.4, \\ 0.6, & a \leq 0.2. \end{cases} \quad (8)$$

3 实验结果与分析

所提出的核心算法采用 C 语言编程, 在

Windows 操作系统及 Inter Core i5-4460, 3.4 GHz CPU, 4.00 GB RAM 的环境下, 应用于 ISCAS 标准测试电路^[18]进行测试. CMOL 电路的常开缺陷率 $P_{open} = 20\%$ 和连通域半径 $R=18$ 与比较文献保持一致, 同时实现了 SimE^[12]方法、DS^[14]方法和 DUM^[11]方法, 并与本文所提出的方法进行比较. 由于 MA 是一种随机搜索方法, 搜索的随机性将会导致解的质量不确定性. 所以所提出的方法在相同缺陷图中对每个标准测试电路进行 20 次试验, 并且记录试验数据的平均值作为最终的测试结果.

根据一些实验和以往的经验, 实验参数设置如下: 种群规模 $P_{size} = 48$, 交叉率 $R_C = 0.33$, 突变率 $R_M = 0.01$, 局部搜索概率 $R_L = 0.2$, 最大迭代次数为 5000. 在基于 SA 的局部搜索中, 初始温度 $T=0.2$, 结束温度 $T_{end} = 0.1$, 内部最大迭代次数为 50. 并且针对常开缺陷的 3 种类型, 将常开缺陷率 P_{open} 细化为 3 个指标: 纳米二极管常开缺陷 P_n 、纳米线非周期性断裂 P_w 、界面接口引脚常开缺陷 P_i , 共同作用于 CMOL 电路. 由于 CMOL 电路中纳米二极管数量远多于 CMOL 单元数目, 因此在相同的 P_w 和 P_i 环境下, 即在 $P_w = P_i = 5\%$, $P_n = 10\%$ 的条件下进行比较.

实验结果见表 1, 其中, “单元”表示标准测试电路节点数, “时间”表示成功完成电路缺陷容忍映射时 CPU 运行时间, “线长”表示映射电路互连线长度总和, “成功率”表示在 20 次试验中成功完成缺陷容忍映射的概率, “平均值”表示对应方法运行的结果平均值, “改进量”表示本文方法相对于其他方法平均值的改进量.

由表 1 可知, 本文方法与 SimE 方法相比, 时间和线长分别有 36.98% 和 26.79% 的改进, 这主要是由于本文方法将 CMOL 电路映射配置单元的搜索问题转化为矩阵间矩阵元的匹配问题, 减少了缺陷容忍映射问题的复杂性, 并且在矩阵匹配过程中对不同的矩阵元值施加不同的惩罚系数, 矩阵元值越大, 其惩罚系数越大. 因此, 最终得到映射结果的紧凑程度会高于 SimE, 而线长是反映电路映射结果紧凑程度的重要指标, 映射结果越紧凑, 线长越小, 反之, 线长越大. 此外, 本文提出的方法当且仅当矩阵中所有矩阵元都匹配成功时才返回解, 并不需要插入反相器对来拓展违反连

表1 不同方法实验结果比较

电路	单元	SimE ^[12]		DS ^[14]		DUM ^[11]			本文方法		
		时间/s	线长	时间/s	线长	时间/s	线长	成功率/%	时间/s	线长	成功率/%
s27	19	0.01	94	0.01	45	<0.01	62	100	<0.01	64	100
s208	136	0.56	1 716	0.12	1 112	0.18	1 438	100	0.07	1 212	100
s298	122	0.43	2 331	0.12	1 429	0.35	1 697	100	0.11	1 351	100
s344	180	1.01	2 867	0.34	1 870	0.48	2 331	100	0.27	1 499	100
s349	184	1.22	2 941	0.36	1 858	0.65	2 247	100	0.26	1 897	100
s382	175	0.86	3 654	0.48	2 161	0.72	2 471	100	0.44	2 079	100
s386	164	0.93	3 058	0.34	2 534	0.79	2 796	100	0.31	2 446	100
s400	188	1.09	3 261	0.48	2 288	0.77	2 824	100	0.29	2 384	100
s420	299	3.74	4 762	1.93	3 428	2.46	4 094	100	1.16	3 922	100
s444	187	1.13	3 342	0.86	2 452	1.87	2 731	100	0.44	2 475	100
s510	304	3.75	6 130	2.06	4 062	5.99	4 649	90	1.24	4 842	100
s526	273	2.65	6 488	1.85	4 081	5.84	4 957	100	1.68	4 305	100
s641	302	3.30	5 108	1.85	3 243	4.16	4 070	100	1.12	3 823	100
s713	321	3.21	6 612	1.90	3 601	4.91	4 478	90	1.71	3 827	100
s820	447	17.79	11 698	14.61	8 842	28.67	9 076	70	12.91	9 690	100
s832	454	15.40	13 468	16.21	8 768	29.46	10 027	50	10.91	9 726	100
s838	606	27.30	14 513	27.17	8 447	35.23	11 062	60	15.92	10 728	100
s1196	675	47.86	18 601	46.20	11 710	58.99	13 077	30	29.91	13 191	100
s1238	724	60.47	18 145	54.57	12 683	89.63	14 269	30	42.79	14 818	100
平均值		10.14	6 778	9.02	4 453	14.27	5 176	85	6.39	4 962	100
改进量/%		36.98	26.79	29.16	-11.43	55.22	4.13	17.29			

通域约束的单元路径, 因此没有产生额外线长的消耗.

与 DS 相比, 时间改进 29.16%, 线长却存在 11.43%的代价. 这是因为 DS 采用动态分层的缺陷容忍映射方法, 分层之后, 由于每层中的映射资源有限, DS 以时间为代价在层内进行就近容忍映射来减小线长, 与本文采用的在全局范围内进行矩阵间矩阵元匹配的缺陷容忍映射方法相比, DS 映射结果的紧凑程度更高, 因此线长更小.

与 DUM 相比, 时间和成功率分别有 55.22%和 17.29%的改进, 线长改进不大. 这主要是由于 DUM 采用缺陷无意识的单元映射方法, 即在 $N_c \times N_c$ 大小的存在缺陷的 CMOL 电路上提取一块 $K \times K$ ($K < N_c$) 大小的无缺陷子电路, 然后在无缺陷子电路中进行单元映射, 映射资源相对集中. 但是随着电路规模的增大, 无缺陷子电路提取的难度增大, 导致大规模电路的缺陷容忍映射成功率降低.

4 结语

针对 CMOL 常开缺陷的容忍映射问题, 本文将逻辑电路和 CMOL 阵列分别建模为矩阵表示, 通过交换矩阵间行列的位置实现矩阵间元素匹配来完成映射, 与已有方法相比, 在求解速度上具有优势. 由于本文不涉及常连缺陷容忍映射, 而常连缺陷存在缺陷传播的问题, 如何针对常连缺陷对 CMOL 阵列进行矩阵的重新建模是未来研究重点.

参考文献:

- [1] Likharev K K, Strukov D B. CMOL: Devices, Circuits and Architectures[M]//Cuniberti G, Richter K, Fagas G. Introducing Molecular Electronics. Heidelberg: Springer, 2005, 680:447-477.
- [2] Lu W, Lieber C M. Nanoelectronics from the bottom up [J]. Nature Materials, 2007, 6(11):841-850.
- [3] Likharev K. Electronics Below 10 nm[M]//Greer J, Korkin A, Labanowski J. Nano and Giga Challenges in

- Microelectronics. Amsterdam: Elsevier, 2003:27-68.
- [4] Xia Y S, Chu Z F, Hung W N N, et al. An integrated optimization approach for nanohybrid circuit cell mapping[J]. IEEE Transactions on Nanotechnology, 2011, 10(6):1275-1284.
- [5] Sait S M, Arafeh A M. Efficient CMOL nanoscale hybrid circuit cell assignment using simulated evolution heuristic [C]//Proceedings of the Great Lakes Symposium on VLSI, New York: ACM Press, 2012:21-26.
- [6] Sait S M, Oughali F C, Arafeh A M. Engineering a memetic algorithm from discrete Cuckoo search and Tabu search for cell assignment of hybrid nanoscale CMOL circuits[J]. Journal of Circuits, Systems and Computers, 2016, 25(4):1650023.
- [7] Sait S M, Arafeh A M. Cell assignment in hybrid CMOS/nanodevices architecture using Tabu search[J]. Applied Intelligence, 2014, 40(1):1-12.
- [8] Farazmand N, Tahoori M B. Multiple fault diagnosis in crossbar nano-architectures[C]//2010 15th IEEE European Test Symposium, IEEE Computer Society, 2010:94-99.
- [9] Strukov D B, Likharev K K. CMOL FPGA: A reconfigurable architecture for hybrid digital circuits with two-terminal nanodevices[J]. Nanotechnology, 2005, 16(6):888-900.
- [10] Hung W N N, Gao C J, Song X Y, et al. Defect-tolerant CMOL cell assignment via satisfiability[J]. IEEE Sensors Journal, 2008, 8(6):823-830.
- [11] 陈定亨, 夏银水, 储著飞. 缺陷无意识的 CMOL 单元容错映射[J]. 计算机辅助设计与图形学学报, 2017, 29(11):2133-2139.
- [12] Sait S M, Arafeh A M. Reconfiguration-based defect-tolerant design automation for hybrid CMOS/nanofabrics circuits using evolutionary and non-deterministic heuristics [J]. Arabian Journal for Science and Engineering, 2015, 40(9):2515-2529.
- [13] Chen D, Xia Y, Wang Z. Stuck-at-close defect propagation and its blocking technique in CMOL cell mapping[J]. Microelectronics Journal, 2018, 72(2):100-108.
- [14] 查晓婧, 夏银水, 储著飞. 动态分层及其在 CMOL 电路缺陷容忍映射优化中的应用[J]. 计算机辅助设计与图形学学报, 2019, 31(8):1457-1466.
- [15] 陈定亨. CMOL 电路的容错映射研究[D]. 宁波: 宁波大学, 2018.
- [16] Chu Z F, Xia Y S, Hung W N N, et al. A memetic approach for nanoscale hybrid circuit cell mapping[C]//2010 13th Euromicro Conference on Digital System Design, IEEE Computer Society, 2010.
- [17] Gören S, Ugurdag H F, Palaz O. Defect-aware nanocrossbar logic mapping through matrix canonization using two-dimensional radix sort[J]. ACM Journal on Emerging Technologies in Computing Systems, 2011, 7(3):1-16.
- [18] Brglez F, Bryan D, Kozminski K. Combinational profiles of sequential benchmark circuits[C]//Proceedings of IEEE International Symposium on Circuits and Systems, IEEE Computer Society, 1989:1929-1934.

Defect-tolerant mapping of CMOL circuit expressed by matrix

XU Pengfei, XIA Yinshui*, ZHA Xiaojing, GU Xiangui

(Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China)

Abstract: To address the problem of cell defect-tolerant mapping in flawed CMOS/nanowire/molecular hybrid (CMOL) circuits, a matrix-based defect-tolerant mapping method for CMOL circuits is proposed. The logic circuit and CMOL circuit are first expressed by a matrix respectively, followed by using the memetic algorithm to search for matching characters between the matrices. The matching strategy with priority assigned to small matrix element value is used to complete the cell defect-tolerance mapping. The experimental results of the ISCAS benchmarks show that, compared with the existing approaches, the CPU runtime of the proposed method is increased by 36.98%.

Key words: CMOL; defect-tolerant cell mapping; matrix representation; memetic algorithm

(责任编辑 韩 超)