

5

Síntese Automática de Circuitos de QCA por Algoritmos Genéticos

5.1.

Descrição do Problema

Hardware Evolucionário (EHW) compreende a síntese automática de sistemas eletrônicos por meio de algoritmos evolutivos, tal como os Algoritmos Genéticos (AG). Devido à manipulação genética da representação de um circuito, EHW encontra, muitas vezes, soluções otimizadas, cujas topologias são consideradas inimagináveis quando comparadas às topologias tradicionalmente utilizadas pelos especialistas humanos.

Conforme já foi apresentado anteriormente, o cromossomo é o componente básico de um AG e representa um indivíduo (solução) de um espaço de busca de um determinado problema. Com o intuito de sintetizar circuitos de QCA otimizados, com a lógica desejada, o AG deve ser capaz de evoluir a topologia do circuito, posicionando as células corretas em suas respectivas zonas de *clock*. Logo, o sistema evolucionário usado neste trabalho tem, na realidade, três cromossomos, baseados em um modelo co-evolucionário cooperativo [81], conforme detalhado a seguir.

Neste modelo, duas ou mais espécies formam um ecossistema. Como na natureza, as espécies são geneticamente isoladas, isto é, cada indivíduo pode somente se reproduzir juntamente com outro indivíduo da mesma espécie. Isto é obtido simplesmente isolando cada espécie em uma população separada. As diferentes espécies somente interagem umas com as outras através de um domínio compartilhado e possuem uma relação de cooperação. O modelo co-evolucionário cooperativo genérico é ilustrado na figura 27.

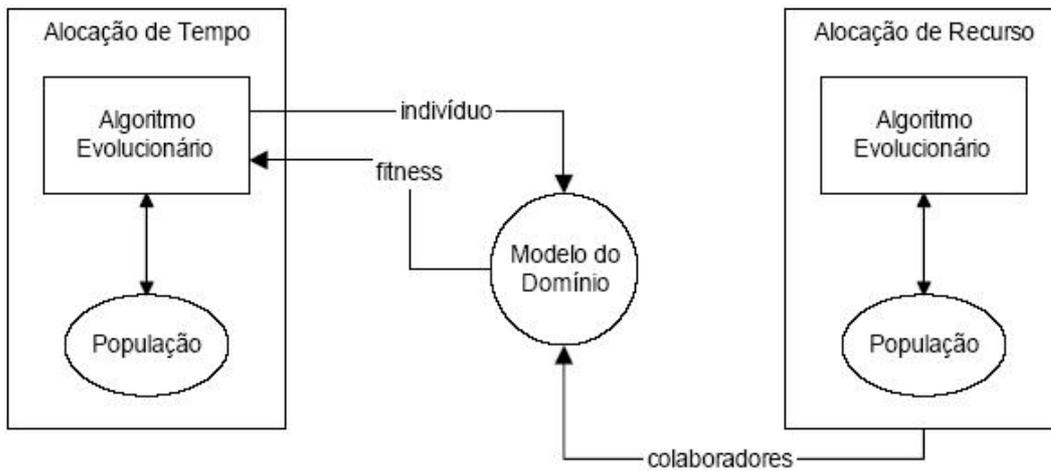


Figura 27– **Modelo co-evolucionário cooperativo genérico**

Apesar da figura 27 mostrar apenas duas espécies, o modelo pode ser generalizado para qualquer número de espécies diferentes. Cada espécie evolui em sua própria população (já que são geneticamente isoladas) e se adapta ao ambiente através de diversas aplicações do algoritmo evolucionário. Com o objetivo de calcular o *fitness* do indivíduo de uma certa espécie, cada indivíduo deve ser submetido ao modelo de domínio (que contém a função de avaliação) juntamente com um ou mais colaboradores de cada uma das espécies remanescentes (formando uma solução completa para o problema a ser resolvido).

A solução final encontrada ao juntar-se indivíduos de cada uma das espécies de modo a formar um elemento que represente a solução completa é chamada de colaboração. Após ter-se montado a colaboração, a mesma pode ser submetida à função de avaliação no modelo do domínio. Se apenas um indivíduo de cada espécie for selecionado para construir a colaboração, como no caso deste trabalho, ter-se-á apenas um valor de avaliação que pode ser usado como *fitness*.

Com relação à de seleção de colaboradores, existem diversos métodos diferentes que variam o grau de pressão exercido nesta seleção [82]. Neste trabalho utilizou-se um método extremamente ganancioso, que seleciona sempre o melhor indivíduo da geração anterior. Esta escolha foi feita a partir de resultados de experimentos preliminares.

5.2. Representação dos Indivíduos

O simulador desenvolvido, descrito no capítulo 4, recebe como entrada uma matriz que representa a topologia do circuito a ser simulado. Esta matriz pode ser vista como uma grade, como a ilustrada na figura 28 que possui 7x7 elementos. Cada ponto no interior da grade representa o centro de uma célula, logo cada célula de QCA é definida por quatro quadrados da grade, conforme já foi explicado anteriormente. O exemplo da figura 28 possui duas células de entrada (cinza) e uma célula de saída (preta). Todas as outras n posições (pontos internos) da grade, que estão disponíveis para a colocação de uma célula de QCA, são numeradas de 1 a n . Observe que todos os pontos nas bordas das células de entrada e nas bordas da célula de saída não são numerados. Isto ocorre porque novas células de QCA não podem ser posicionadas nestas posições, já que elas ocupariam a mesma posição de parte de uma célula já existente na grade.

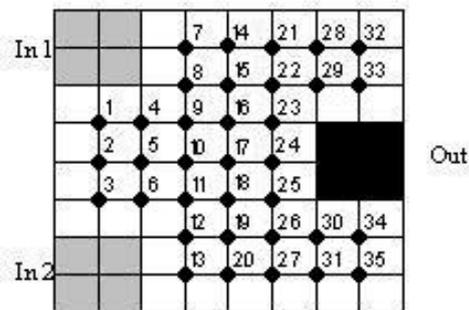


Figura 28– **Grade de um circuito com duas entradas (cinza) e uma saída (preto). Os números indicam as posições (o ponto à esquerda e abaixo do número) onde novas células de QCA podem ser posicionadas.**

A representação usada para criar uma topologia de circuito idealmente deve garantir que soluções ilegais não serão geradas durante o processo de evolução. Soluções legais implicam que as células de saída serão atingidas a partir das células de entrada. Além disso, a representação deve garantir que o caminho entre uma célula de entrada e uma célula de saída deve ser único, isto é, este caminho não deve passar por nenhuma outra célula de entrada ou célula de saída. Esta última restrição aumenta a chance de que todas as células de entrada tenham influência em todas as células de saída. Um critério que engloba ambas as

restrições é verificado e aceito se, partindo de uma saída é possível chegar a todas as entradas através de caminhos distintos.

Para garantir estas restrições, este trabalho propõe um modelo inspirado em um modelo anterior aplicado ao problema do caixeiro viajante com problemas de restrições de precedência [83]. Logo, na topologia do circuito, uma nova célula de QCA pode ser posicionada na grade se, e somente se, existir uma célula vizinha a esta posição, que não seja uma célula de saída. Assim, o cromossomo que define a topologia do circuito é um vetor com n elementos. Cada elemento guarda um único valor entre 1 e n , que determina a ordem de prioridade de se colocar uma célula de QCA na posição correspondente a este número. Sempre que uma célula é posicionada no circuito, volta-se ao início do cromossomo para verificar qual a próxima célula que deverá ser posicionada. Abaixo é ilustrado um exemplo de cromossomo.

Cromossomo da topologia =

(23 32 12 7 30 8 14 20 13 26 35 10 6 24 17 31 13 16 21 28 5 18 34 22 11 4 15 33 29 19 25 9 2 27 1).

O segundo cromossomo, que representa o tipo de célula a ser colocado no circuito, também é um vetor com a mesma dimensão (n elementos), mas neste caso o cromossomo é binário. Logo, se uma célula é colocada na posição k da grade, o tipo da célula é determinada pelo valor binário do elemento k deste cromossomo. O binário zero indica que a célula será uma célula de QCA convencional, enquanto o binário um indica que será uma célula de QCA rotacionada.

Cromossomo do Tipo =

(0 1 0 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 0 1 1 0 1 0 1).

De maneira similar, o terceiro cromossomo, que representa a zona de *clock* a que cada célula de QCA pertence, também usa valores binários. Neste caso, o valor binário zero significa que a célula está na mesma zona de *clock* que seus vizinhos, já o valor binário um significa que a célula está uma zona de *clock* à frente de seus vizinhos.

Cromossomo do Clock =

(0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0 1 0 1).

Assim, usando os três cromossomos e o algoritmo apresentado na figura 29 para construir uma solução, é possível gerar um circuito válido.

procedure: gera circuito válido

input: grade com as células de entrada e saída previamente definidas

while (condição de parada é falsa)

- Varrer o vetor de topologia até encontrar uma posição na grade onde pode ser posicionada uma nova célula de QCA;

- Checar o tipo de célula de QCA que será colocada na posição encontrada;

- Verificar se muda o clock;

if (todas as células de saída foram atingidas com um caminho único partindo de cada entrada)

- Condição de parada é verdadeira.

end if

end while

end procedure

Figura 29– **Algoritmo de construção do circuito.**

Os passos seguintes são utilizados como exemplo para o desenvolvimento de um circuito, utilizando os três cromossomos apresentados anteriormente. A primeira posição onde uma nova célula pode ser colocada, considerando as prioridades definidas pelo cromossomo de topologia e a restrição de que uma célula só pode ser posicionada se tiver pelo menos uma célula vizinha (que não seja a saída), é a posição 12. Os outros dois cromossomos indicam que esta célula é convencional e está na mesma zona de *clock* que a célula de entrada.

Após se posicionar esta primeira célula, verifica-se se a condição de parada foi alcançada. Caso ainda não tenha chegado ao fim, volta-se ao início do cromossomo de topologia e verifica-se qual é a próxima célula a ser posicionada na grade.

A seqüência da execução indica que uma célula rotacionada e na mesma zona de *clock* que a célula de entrada é colocada na posição 7. A seguir, uma célula rotacionada e uma zona de *clock* à frente da zona de *clock* da célula na posição 7 é colocada na posição 23.

Observe que, neste momento, a saída é atingida. Porém, partindo-se da saída só é possível atingir a célula de entrada no canto superior esquerdo, mas não a entrada no canto inferior esquerdo. Logo, a condição de parada ainda não foi satisfeita. Em seguida, uma célula convencional e na mesma zona de *clock* que a célula na posição 23 é posicionada na posição 32. Finalmente, uma célula de QCA convencional é colocada na posição 26, uma zona de *clock* à frente da zona de *clock* da célula 12. Agora a condição de parada é satisfeita, já que, partindo-se da saída, ambas as entradas são alcançadas, através de caminhos distintos. A figura 30 mostra a construção do circuito passo a passo.

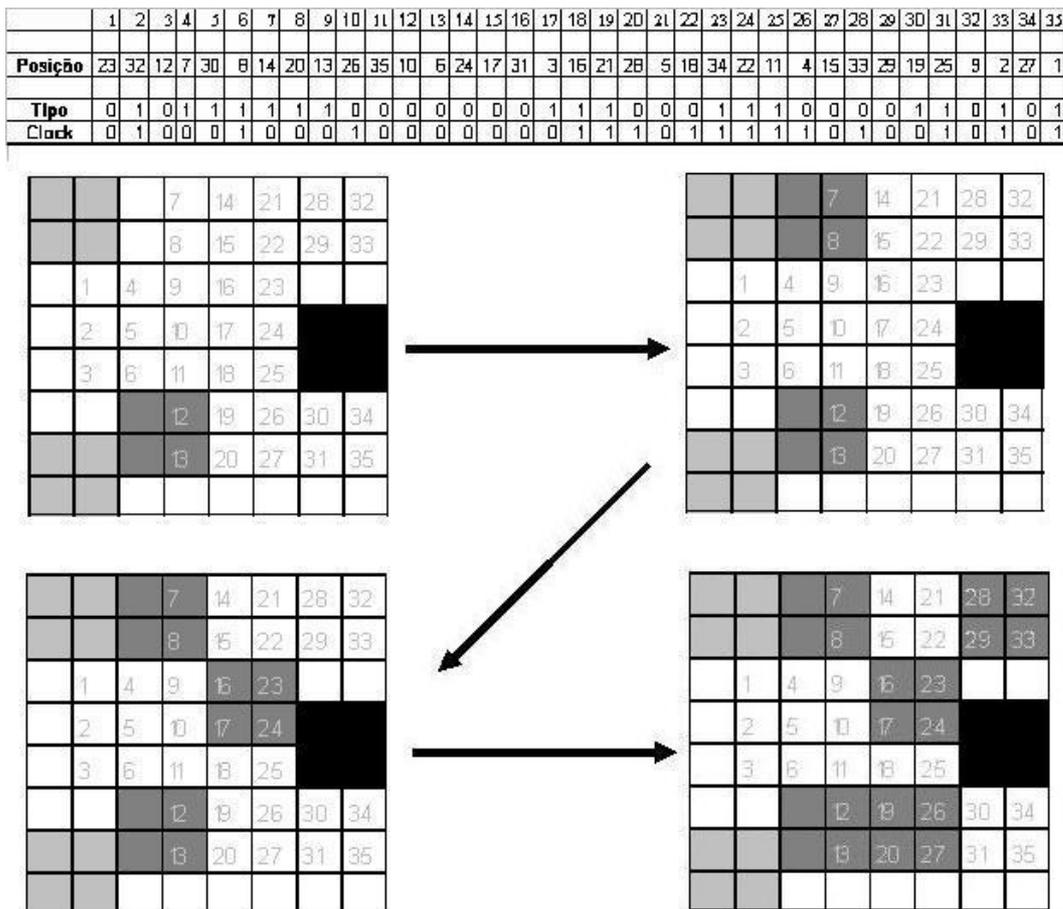


Figura 30– Criação passo a passo da topologia do circuito de acordo com os cromossomos utilizados.

5.3. Avaliação dos Indivíduos

O número de saídas corretas (acertos), em comparação com uma tabela verdade pré-definida, é usado pela função de avaliação para medir o desempenho do circuito. No caso do circuito possuir n entradas, o objetivo é encontrar (evoluir) um circuito que retorne 2^n saídas corretas. Logo, a função de avaliação é dada pela seguinte equação:

$$\text{Aptidão} = \text{Acertos}, \quad 0 < \text{Acertos} < 2^n \quad (8)$$

Sempre que um circuito digital está sendo evoluído, o objetivo é encontrar um circuito que é totalmente correto, segundo suas especificações. Logo, ao contrário da evolução de circuitos analógicos, a evolução de circuitos que apresentam 99% de respostas corretas não é interessante. Porém, Algoritmos Genéticos são facilmente atraídos para estes mínimos locais.

Com o objetivo de resolver este problema, é necessário incluir outros termos na função de avaliação, conforme Zebulum et al [5]. Tais modificações na função de avaliação são, de alguma forma, específicas para cada tipo de problema, logo eles serão apresentados separadamente para cada estudo de caso apresentado na próxima seção. Zebulum et al apresenta um exemplo, descrito a seguir, que ilustra bem este caso.

A figura 31 abaixo mostra a tabela verdade de uma função lógica hipotética de 3 entradas e 1 saída, com 2^3 possíveis valores de saída. Este exemplo consiste de uma soma de produtos, onde a função objetivo é dada pela soma dos termos $m1$ e $m2$. A quantidade de acertos do termo $m1$ é igual a 7 e do termo $m2$ é 6 acertos. Da mesma forma, o termo $m3$ também possui um número de acertos igual a 7 em um total de 8. Portanto, $m3$ tem uma maior probabilidade de ser encontrado no GA do que o $m2$. Porém, como é usado o paradigma da soma de produtos, $m3$ não faz parte da solução, já que ele retorna o resultado 1 no caso da entrada ser 100, quando o resultado correto seria 0. Logo, este termo será penalizado pela nova função de avaliação, pois caso contrário poderia levar o AG a um mínimo local, dificultando a busca da solução correta.

a b c	ab	$a\bar{b}c$	a	$ab + a\bar{b}c$ (objetivo)
0 0 0	0	0	0	0
0 0 1	0	0	0	0
0 1 0	0	0	0	0
0 1 1	0	0	0	0
1 0 0	0	0	1	0
1 0 1	0	1	1	1
1 1 0	1	0	1	1
1 1 1	1	0	1	1

$m1 = ab$
$m2 = a\bar{b}c$
$m3 = a$

Figura 31– Exemplo do problema de ótimo local quando a função de avaliação só leva em consideração o número de acertos.

Outro objetivo importante é reduzir o número de células de QCA presentes no circuito. Circuitos com menos células são mais rápidos e têm menos tendências a falhas. Para alcançar este objetivo, um novo termo na função de avaliação acrescenta um bônus para os circuitos que já tenham acertado todas as saídas, ou seja, já possuem a lógica correta. O bônus é inversamente proporcional ao número de células no circuito, logo, quanto menos células o circuito possuir, maior sua avaliação.

A idéia de somente dar bônus a circuitos com a lógica totalmente correta evita que circuitos menores e com um número de acertos insuficientes tenham uma boa avaliação, o que pode dificultar a evolução de circuitos corretos. O valor do bônus também é específico para cada experimento.

O capítulo 6 a seguir discute os estudos de casos, apresentando os circuitos de QCA sintetizados pela técnica proposta nesta dissertação.