A Survey on Placement and Routing for Field-Coupled Nanocomputing

Ruan Evangelista Formigoni[1], Ricardo dos Santos Ferreira[2], José Augusto M. Nacif[1]

[1] Science and Technology Institute, Universidade Federal de Viçosa, Florestal, Brazil
[2] Informatics Department, Universidade Federal de Viçosa, Viçosa, Brazil
e-mail: {ruan.formigoni,ricardo,jnacif}@ufv.br

Abstract— **CMOS technology is reaching power, thermal, and physical limits at an alarming pace. As a response, post-silicon research investigates alternative technologies to perform computation. Field-Coupled Nanocomputing (FCN) presents low power dissipation, high frequencies, and room temperature operation. Nevertheless, FCN imposes several challenges in the development of efficient and scalable CAD tools. The placement and routing step is especially tricky in FCN compared to CMOS because of synchronization issues inherent to these technologies, such as path balancing and reconvergent paths. In this work, we survey the state-of-art of placement and routing algorithms for FCN. We describe the most recent FCN placement and routing algorithms, highlighting their limitations and, finally, presenting future work directions.**

I Introduction

The Complementary Metal Oxide Semiconductor (CMOS) transistor came forth as the current standard for the fabrication of integrated circuits. For continuous performance improvements, the primary goals are to decrease the feature size and power consumption [1]. In 1965, Gordon E. Moore stated that the number of components in an integrated circuit would increase in a factor of 2 every two years [2]. Industry advances sustained this miniaturization pace until the early 2000s [3] and are still undergoing, although the size of the transistors is reaching near-atomic limitations. Research efforts are in progress to avoid issues such as power dissipation and static power consumption due to leakage current [4].

For further advancements in power and consumption efficiency, several research groups are working on Field-Coupled Computing (FCN) technologies [5]. FCN includes nanomagnetic logic (NML), an approach to perform computation with stray-field interactions between single-domain nanomagnets, at room temperature [6–8]. Moreover, FCN is considerably efficient in terms of power dissipation [9]. In addition to NML, an important FCN paradigm is the Quantum-dot Cellular Automata (QCA) [10]. QCA utilizes cells composed of four quantum-dots with two mobile electrons. Electron tunneling represents binary information with two antipodal arrangements, known as polarization of the cell. The mutual repulsion between the electrons allows for the propagation of logic signals between wire arrangements. Also, QCA allows for ultra-low power dissipation through adiabatic clocking [11, 12].

In FCN, the problem of Placement and Routing (P&R) is different from well-established CMOS, presenting several new challenges, such as path balancing [13–16]. FCN P&R is also NP-Complete [17], which emphasizes the investigation of alternative approaches for feasible solutions in terms of computational complexity, area overhead, and throughput. Similarly to other P&R problems [18], based on greedy approaches [19,20], a crucial requirement for FCN P&R is path balancing, which increases complexity. Furthermore, each technology has its intrinsic characteristics, e.g., the number of clocking phases and patterns to keep circuit synchronization (clocking schemes) the number of cells in each clock phase.

In this paper, we review the current state-of-the-art of FCN P&R. We discuss meta-heuristics algorithms [21], graph node duplication [22], graph partitioning [23], trade-offs between wire crossing minimization and area overhead [24], orthogonalization P&R techniques [25], clocking constraint bypassing [26], and divide-and-conquer [27] approaches.

We organize this paper as follows: In section II, we review the basic concepts of the QCA and NML technologies; In section III, we introduce the basic concepts of P&R for FCN technologies. In section IV, we present traditional approaches for graph partitioning and refinement, together with the first approaches for P&R. In section V, we explore further proposals that take advantage of novel structures with feedback support for sequential circuits. Lastly, in section VI, we express our closing thoughts and conclude the paper.

II Field-Coupled Nanotechnology

In this section, we explain the basic concepts and of the QCA and NML technologies.

II.A QCA Basics

We refer to the QCA technology fundamental logic element as a cell. This element contains, generally, four quantum-dots. Two mobile electrons form two possible antipodal configurations (polarization) through Coulomb repulsion. Figure 1a represents the binary logic *1*, as opposed to Figure 1b, which indicates logic state *0*.

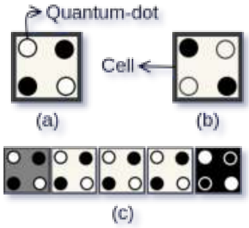

Fig. 1 (a) QCA Binary 1. (b) QCA Binary 0 (c) QCA Wire.

Digital Object Identifier 10.29292/jics.v16i1.480

It is possible to perform logic operations through signal propagation by pairing two or more cells, as shown in Figure 1c. This configuration represents a wire, where we highlight the input in gray and the output in black. A signal transmitted through a QCA wire degenerates when the number of consecutive cells exceeds five. Therefore, to create more complex designs, we need to use a clocking system, thus splitting the signal transmission into four phases: *switch*, *hold*, *release*, and *relax* [28]. Figure 2a shows the transitions of a single cell from a phase to another. Next, in Figure 2b, an example of the signal propagation demonstrates how the clocking system systematically transmits the input signal (black), to the output (gray), with no disruption.

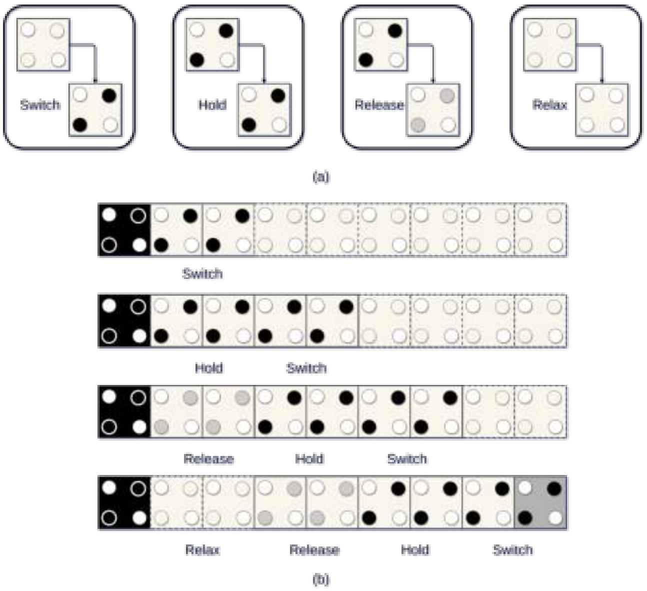

Fig. 2 (a) QCA Clocking System. (b) Signal Propagation with over 5 cells.

Figure 3 presents the QCA *majority gate* (MG) [29], which acts as the technology basic logic building block. In Figure 3a *A*, *B*, and *C* are the inputs of the structure, replicating the predominant signal to *M* and thereafter to the output *O*. It is possible to configure the majority gate to realize only *AND* or *OR* functions by fixing one the inputs. Figure 3b shows an *OR* gate, and Figure 3 an *AND* gate. In addition, new advances in logic synthesis research are based on majority logic [30, 31]. As opposite to CMOS, an inverter in QCA presents a considerable area overhead. We show a QCA inverter in Figure 4.

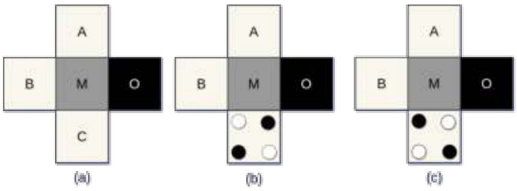

Fig. 3: (a) The basic majority gate. (b) Majority gate with *OR* function. (c) Majority gate with *AND* function.

II.B Nanomagnetic Logic Basics

The Nanomagnetic Logic (NML) is a promising FCN paradigm to perform room-temperature computation [7] with low power dissipation [9]. NML utilizes single-domain nanomagnets as the technology's basic building block. Single-domain nanomagnets have an elongated shape and well-defined bistable behavior [6]. In these nanomagnets, the magnetization of the structure represents binary values. The upwards arrow denotes binary logic value 1 (Figure 5a) while the downwards arrow relates to the binary value 0 (Figure 5b).

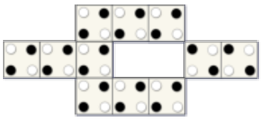

Fig. 4 An inverter implemented in QCA.

In contrast to the QCA technology, NML reverses the signal when propagating logic to a neighbor nanomagnet in a horizontal arrangement, known as antiferromagnetic coupling. Figure 5c depicts this behavior, where an odd number of nanomagnets allow correct signal propagation from input to output. Thus, it is possible to create an inverter with an even number of nanomagnets. Figure 5d shows an additional vertical arrangement, which is known as ferromagnetic coupling, and preserves the cascading signal as previously explained for the QCA technology.

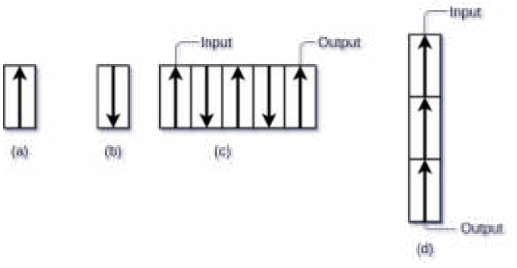

Fig. 5: (a) Binary 1. (b) Binary 0. (c) Antiferromagnetic coupling. (d) Ferromagnetic coupling.

When we increase the number of nanomagnets, NML wires suffer from precipitate switching induced by thermal noise [32], thus propagating incorrect logic values. The limit of cascading signal propagation is around five magnets to avoid this issue. In NML, several works [33, 34] use a 3-phase clocking system. The *reset* state puts neighboring nanomagnets in an unstable state with a magnetic field. Subsequently, the *switch* state slowly removes the magnetic field and the input signal can influence the closest neighbor and propagate the signal. Finally, the *hold* phase holds the signals in a fixed magnetization to propagate to nanomagnets in the switch state (Figure 6).

NML utilizes the majority gate [35] for logic operations, Figure 7 shows the possible configurations for the logic element. Figure 7a presents the majority voter characteristic of the center nanomagnet concerning the three inputs. Figures 7b and 7c illustrate the usage of a majority gate to perform fundamental logic functions by fixing one of the inputs downwards (*AND*) or upwards (*OR*), respectively.

II.C Clocking Schemes

Clocking systems coordinate the synchronization of the propagated signals, having a variable number of technology-dependent phases. A *zone partition* [36] organizes several

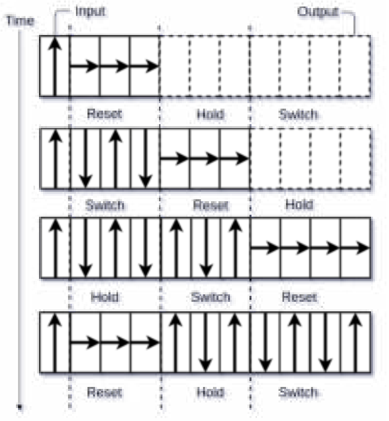

Fig. 6 Signal propagation with a 3-phase clocking system in NML.

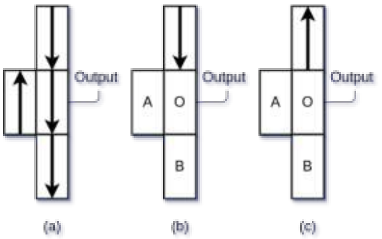

Fig. 7: Majority gate. (a) Inputs lead to *0* state. (b) *And* logic gate. (c) *Or* logic gate.

logic elements under the same clocking phase. In contrast, logic elements cannot correctly propagate an input signal indefinitely, because thermal fluctuations lead to signal degeneration. In literature, a well-accepted number of consecutive elements is five and constitutes a constraint for partition creation. More cells in the same *clocking zone* are desirable; since more logic operations can occur in the same clocking phase.

Earlier clocking scheme models define partitions with sequential columns that repeat the clocking phase, ordering consistent signal propagation. Figure 8a depicts a 4-phase cutout and Figure 8b depicts a 3-phase cutout, both designed to work with combinational logic [37]. Figure 8b highlights the components of a basic cutout. The smallest element is a cell that holds a logic element, e.g., a nanomagnet or a quantum-dot. Each column is a clocking zone, and every element in it is in the same clocking phase. The whole composes the cutout, which, when replicated, creates an arbitrarily large *clocking scheme*, as demonstrated in Figure 8c.

III Fundamentals of Field-Coupled Nanotechnologies

III.A Tiles

The clocking zones discussed in section IIII.C are unidimensional, i.e., the columns transitively change clocking phases from left to right. Subsequent proposals show models that have distinct clocking zones in two dimensions (rows and columns), as the Two-Dimensional Wave clocking scheme (2DDWave) [28, 38], shown in Figure 9a-b, where the signals flow from the left to right or from top to bottom. Figure 9c-d shows the Universal Scalable and Efficient (USE) clocking scheme [39], and the Robust Efficient and Scalable (RES) clocking scheme [40], the bidirectional signal flow is indicated by the arrows on the edge of the cutouts. These designs introduce a new term called *tile*, indicating a single position in the mxn matrix representation of the proposed cutouts. Between the advantages of this approach, the partitioned columns provide possible area reduction, furthermore the bidirectional signal flow allows for the implementation of sequential circuits with feedback paths.

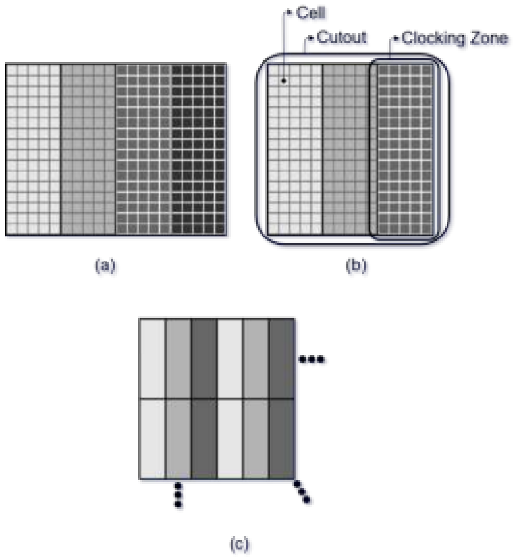

Fig. 8: Clocking Schemes (a) 4-Phases. (b) 3-Phases. (c) Cutout expansion to create a clocking scheme.

A recent proposal [41] introduces further improvements to area overhead through a diamond-shaped *cutout*. In the Convenient Flexible and Efficient (CFE) clocking scheme, shown in Figure 9e, the cutout shares a QCA clocking zone with all other possible three, which provides flexible layout arrangements for QCA designs, as diagonally coupled cells, which result in area and delay reduction. For the NML technology, the Bidirectional Alternating Clocking Scheme (BANCS) [42], shown in Figure 9f, provides a tiled design with the bidirectional signal flow, for a 3-phase clocking system. A remarkable distinction of BANCS is the upwards signal flow that replicates the same tile twice, allowing for a tile transition, without an increase in delay. Figure 9g depicts another recent QCA approach [43] based on the USE clocking scheme. The arrangement allows for shorter feedback paths and solves the problem of metal wire crossing. As mentioned in sections IIII.A and IIII.B, signal disruption occurs in both QCA and NML, and, therefore, the target technology bounds tile size. Most approaches use a *5x5* tile for QCA and a tile of *3x3* for BANCS in NML.

III.B Reconvergent Paths

One of the initial challenges is to balance all internal paths to guarantee the correct circuit synchronization. Consider the graph *G* from Figure 10a, the three paths from the node *I* to the output *O* are reconvergent. However, the signals from the paths *I-A* and *I-C* reach *O* before the path *I-B-D* due to the path delay mismatching. The paths could be balanced by inserting *buffer nodes* as shown in Figure 10b where the nodes W_1 and W_2 equalize the three paths.

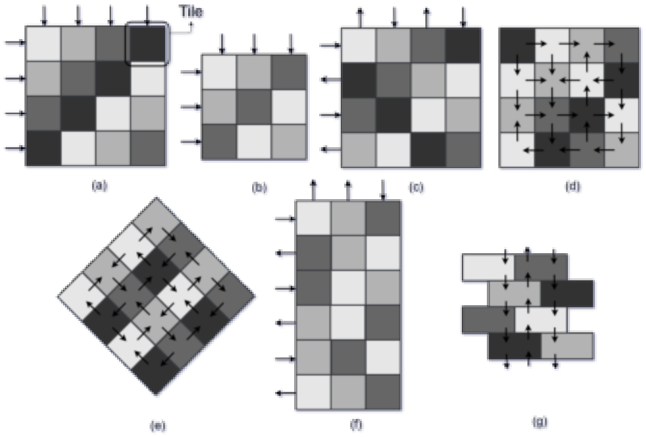

Fig. 9: (a) 4-Phase 2DDWave. (b) 3-Phase 2DDWave. (c) USE. (d) RES. (e) CFE. (f) BANCS. (g) USE-Based.

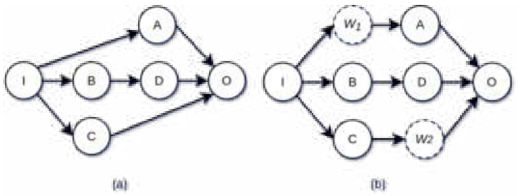

Fig. 10 Graph reconvergence. (a) Unbalanced paths. (b) Balanced paths.

III.C Area Overhead

We define the area overhead as the area occupied by the empty regions. We consider a minimum rectangular area to place the entire circuit graph.

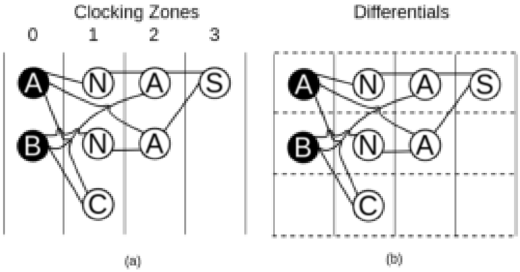

Fig. 11: A Half Adder circuit. (a) Clocking Zone. (b) Overhead introduced in empty regions.

Figure 11 shows the realization of a half adder. Figure 11a depicts the timing constraints in clock zones, and Figure 11b illustrates the overhead (empty regions) derived from the process of cell placement, where there are four empty regions. At the graph level, we do not consider the cross-wiring.

III.D Throughput

In section III.B reconvergent paths achieved perfect synchronization of the output, but, unfortunately, as discussed in section III.C, the result increased the area overhead, the maximum column has a maximum height of 2, in Figure 10a, and 3, in Figure 10b. Reconvergent path balancing is a design choice, rather than a hard requirement, as the circuit in Figure 10a generates a valid result, but not every clock cycle, since the input signal reaches the output through the two shortest paths, I, A, O, and I, C, O, before path I, B, D, O. Thus, we need to consider a trade-off between area and delay. The metric indicating the rate at which a circuit generates a valid result is the *throughput* [44], which starts at 1 and declines as the circuit becomes unsynchronized.

IV Graph-level Placement and Routing

An initial proposal of how to handle the placement and routing (P&R) problem is presented in [36], the process is divided into *zone partitioning* and *zone placement*. In addition, the proposal defines four constraints: (1) the *clocking constraint* requires equal length reconvergent paths as previously discussed in section III.B; (2) the *acyclic constraint* requires that the graph has no cycles; (3) *the logic constraint* defined as the number of logic elements in a single *clocking zone*; (4) *the wire capacity* defined as the number of wires in a single *clocking zone*.

The *zone partitioning* phase starts with a topological sort algorithm and splits the graph in a topological view, assigning an identifier to each level. After that, the authors use the Fiduccia and Mattheyses algorithm (FM) [45] to optimize adjacent partitions minimizing the *cutsize* and *variance*. The *cutsize* is the total number of edges that spawn to external partitions, and the *variance* is the total number of edges that spawn to internal partitions. Figure 12 exemplifies the edge B that belongs to the *cutsize*, and the edge A that belongs to the *variance*.

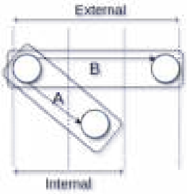

Fig. 12 Internal partition labeled *A* and external partition labeled *B*.

Figure 13 shows the *zone partitioning* phase steps for [36]. Figure 13a, presents the graph of a half-adder, followed by the execution of the topological sort algorithm, which yields the levelized graph of Figure 13b. The next step inserts the buffer nodes to equalize the path delays, depicted in Figure 13c; Figure 13d, shows wire sharing, which minimizes the number of wire nodes. Not all P&R strategies support majority-inverter graphs (MIGs), but it is possible to convert them to And-Or-Inverter graphs (AOIGs). We depict the result in Figure 13e, which adds a novel layer of overhead wire nodes as a trade-off.

The approach [46] is similar to [36], where for *zone partitioning*, they use the algorithm presented in [47] to solve the minimum buffer insertion problem for a pipelined circuit. The solution deems acceptable if the column height variation is low; otherwise, the graph is perturbed. A perturbation inserts a constraint where a node must change its level to make the column heights more uniform, i.e., minimize the area overhead, as a trade-off in latency. Figure 14a shows an example graph, while Figure 14b demonstrates a topological rearrangement, where the global column height size reduces from 3 to 2, by moving node A to the next level.

A third approach at the graph optimization is presented in [22], where the main difference is the usage of node duplication. For an initial solution, it uses the Barycenter heuristic

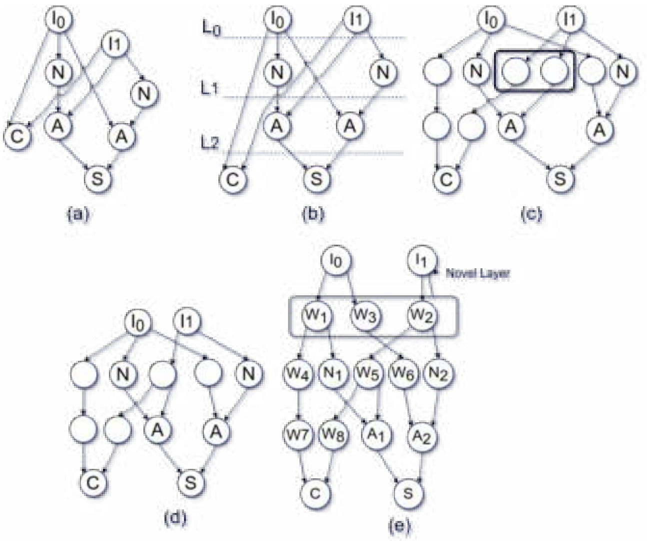

Fig. 13: Half adder circuit. (a) Original graph. (b) Topologically sorted graph. (c) Insertion of buffer nodes. (d) Sharing (merge) of buffer nodes. (e) Novel overhead layer due to the AIOG conversion.

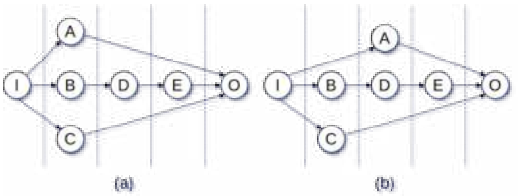

Fig. 14 (a) Original level assignment. (b) Rearrangement.

with simulated annealing, shown in [36]. The output positions are perturbed to add new solutions at each iteration. Moreover, the algorithm aims to reduce the number of wire crossings by duplicating nodes. Therefore, there is a trade-off between wire crossings and area overhead. Figure 15a shows a sample graph with one wire crossing caused by a fan-out of node 1. In Figure 15b, the node is duplicated, and the wire crossing is solved. It is possible to merge the duplicated nodes to minimize the extra area overhead. A similar approach to wire crossing reduction through duplication is demonstrated in [24] for molecular QCA.

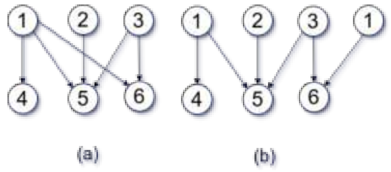

Fig. 15: Half adder graph. (a) Original graph. (b) Wire crossing reduction by replication.

The *zone placement* phase uses the output from the previous phase as input, a K-Layered Bipartite Graph (KLBG). The objective is to embed the nodes onto an mxn grid performing placement on a level-by-level fashion. Unfortunately, the optimal position for the nodes on each level is discovered through exhaustive permutation of every node position. A partition positioning is optimal if it has minimized wire lengths and wire crossings. [36] uses the barycenter method [48]. This heuristic approach sorts the nodes on each level and assigns a position based on inter-partition edges, a random level is selected in the graph, with the simulated annealing algorithm [49]. The solution is then perturbed, i.e., a new partition is used for that level, if it improves the solution's cost.

The fourth proposal at graph level is presented in [50], which introduces a novel *Row Folding* algorithm. A more significant number of nodes in a graph level increases the minimum rectangular area, as it increases the area width of the final layout. The row folding splits a level in a n number of levels to minimize area overhead, at the cost of increased delay. Finally, the graph level does not consider design rules such as the number of wires per cell, cross-wiring, etc. In the next section, we present the P&R approaches at the tile level, which is close to the final layout.

V Tile Based Approaches

A P&R that considers the tiled clocking schemes is NP-Complete [17]. Moreover, [51] presents an exact solution to the problem. The tile approaches allow feedback paths to implement sequential circuits. In this section, we explore tiled P&R approaches, with the simulated annealing and A* algorithms, divide-and-conquer strategies, orthogonalization, and a feasible proposal for sequential circuit realization.

Trindade et al. [52] introduce a novel algorithm for the Universal Scalable and Efficient (USE) clocking scheme [39]. It is a greedy approach by performing graph traversal in breadth-first order, from its outputs to inputs. We execute this process for the half adder example from Figure 15c in Figure 16a. The algorithm progressively increases the distance needed between the nodes of each level, if it finds no feasible solution, starting with an initial value of 1. The distance between the nodes of the graph, in tiles, is shown in Figure 16b.

As a greedy approach, the P&R algorithm [52] can not guarantee a feasible solution. A direct extension to a simple brute-force approach, where it evaluates all possible solutions, quickly becomes unfeasible even for small circuits.

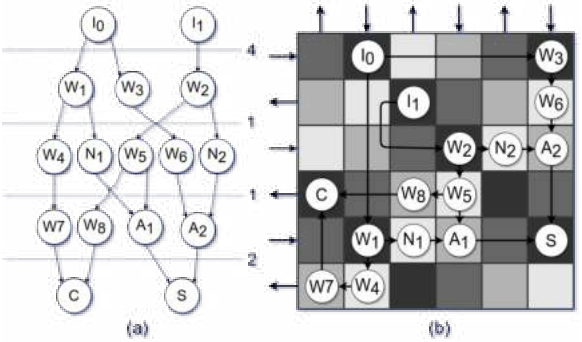

Fig. 16: P&R for the half adder example. (a) Runtime computed level weights. (b) Final layout.

Formigoni et al. [21] use the simulated annealing algorithm to scale further. The improvements are weighted accordingly to the proximity of the fan-ins. The scalability is improved but still unsatisfactory due to the sparse nature of the process in the placement phase. Scalability is feasible with the application of a divide-and-conquer strategy [27] to avoid the rapid growth in complexity. In this methodology, for each graph level, the initial weight of *1* is assigned for the distance in clocking zones of each internal partition. Subsequently, the algorithm divides the graph into three partitions. The first partition encompasses common nodes with the other partitions. The second partition has at least, one common node and edges, and the third is composed of disjoint nodes, which are more challenging and lead to a more significant area and wire lengths overhead. Several subgraph partitions are explored to find improved solutions. For this example, the algorithm places and routes each partition separately, as shown by Figure 17a. Next, the merge process overlaps both solutions to generate the final layout, as shown in Figure 17b. However, the largest evaluated circuit in [27] has 102 gates. A more efficient partition strategy is required to handle larger circuits inside a tiled clocking scheme.

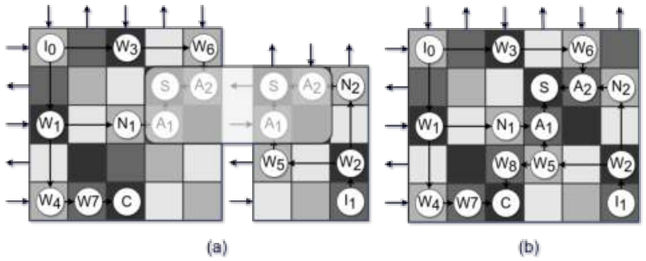

Fig. 17: (a) Individual P&R is performed. (b) Merge of solutions onto intersecting nodes.

The work in [25] presents an approach based on the orthogonal graph drawing (OGD) problem [53]. OGD allows for the embedding of a graph G in a 2-dimensional space, which draws all the vertices in an orthogonal arrangement. The are several algorithms for application in the OGD problem, and the authors chose the approach presented in [54]. However, all nodes should have, at most, a degree of 3. Therefore, the majority gate, which has a degree equals 4 (3 inputs, 1 output), cannot be used, which is a drawback since this gate optimizes the circuit representation considerably [30, 55–57]. In this approach, the process of P&R is done with the enforcement of a particular direction (South and East). Figure 18 presents an example to determine the nodes' positions, where the algorithm labels the incoming edges with the targeted direction, either south or east. Figure 18a shows the result of this preprocessing. The placement phase performs the assignment of a 2D coordinate for each node. Then, the algorithm uses the manhattan distance to perform routing. The routing result is shown in Figure 18b. This approach minimizes wire crossings when compared to the previous approach. The placement and routing perform independently and place the inputs/outputs at the layout borders.

Targeted for the NML technology, an A* algorithm for the routing phase is proposed in [58]. This work is also limited to a maximum node degree of 3. First, a three-step preprocessing phase is done: fan-in management, fan-out management, and graph balancing. The fan-in and fan-out management steps limit the in-degrees and out-degrees of the input graph to two edges by adding buffer nodes, as shown in Figure 19, followed by the graph balancing step. Next, the placement utilizes a breadth-first search to find a feasible solution for the routing phase, which utilizes the *Euclidean Distance* heuristic as a driver for the A* algorithm. Diagonal wire connections between tiles are allowed, and it keeps track of previous solutions to limit wire occupation in each tile. Furthermore, it avoids wire routing in tiles with logic elements. Since the placement and routing are decoupled, this strategy is highly dependant on the efficiency of the placement algorithm. Poor placement leads to a high count of wire crossings.

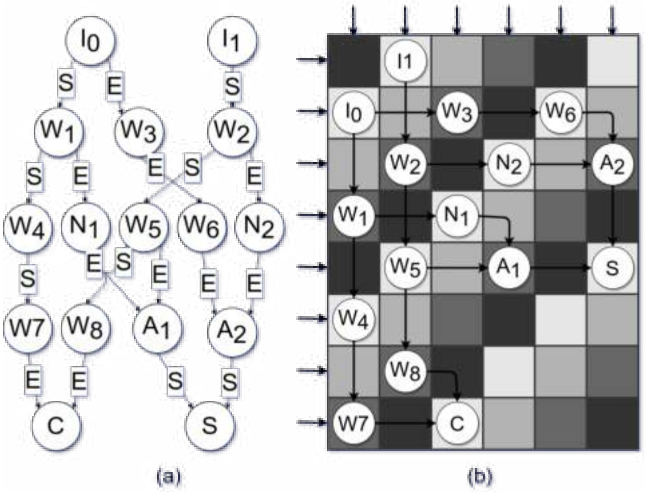

Fig. 18: P&R for the half adder example. (a) Edge labels. (b) P&R for orthogonal drawing.

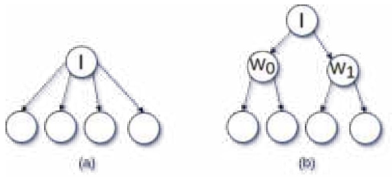

Fig. 19: (a) Node *I* with four fan-outs. (b) Insertion of wire nodes to reduce the fan-out count of *I*.

Recently, [26] presents an alternative approach to ignore clocking constraints, a *synchronization element*, which can stall a signal for a given number of clock cycles, and split the forward and receive functions of a tile into two distinct clocking phases.

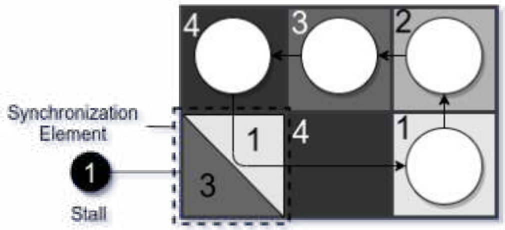

Fig. 20 Sequential circuit implemented with a synchronization element.

Figure 20 depicts a sequential circuit from [26], the synchronization element is highlighted with the dotted square, it acts as tile 1 when the signal is received, and as tile 3, when the signal is sent to tile 4. To the left of the element, is a number representing the cycles the signal is stalled for correct circuit synchronization, i.e., it ignores the signal from the tile 4 above, for one clock cycle. The design receives the signal controlled by tile 4. Subsequently, the signal is held for a cycle and transferred to the tile 4 to the right. In this

Strategy	Req. Balanced Graph	Clocking Scheme	Layout	Scalable	Sequential	Majority Support
S.A. Column-Based [36, 46, 50]	Yes	Column-based	Yes	Yes	No	Yes
Sparse [52]	Yes	USE	No	No	No	Yes
S.A. Sparse [21]	Yes	USE, BANCS	No	No	No	Yes
Divide-and-conquer [27]	No	USE	Yes	Yes	No	Yes
Orthogonalization [25]	No	Column-Based, 2DDWave, RES, USE, BANCS	Yes	Yes	No	No
S.A. Synchronization Elem. [26]	Yes	Column-Based, 2DDWave, RES, USE, BANCS	Yes	Yes	Yes	Yes

scenario, the element acts as a tile of 3. In their work, the authors present waveform results of the circuit simulation. The process utilizes the simulated annealing with a modified version of the A* algorithm in a custom version of the QCADesigner tool [59], which correctly propagates the signals with the proposed element. The work is concluded with a series of benchmarks for combinational and sequential circuits and holds feasible time complexity on over $1,500$ logic gates designs.

Table V presents all approaches, its characteristics of pre-processing stages, the P&R approaches, and the layout generation. We dedicate the first entry of the table to the column-based approaches [36, 46, 50], where a feasible solution requires a KLBG for synchronization and scalable designs. The clocking schemes are sequential with no tiling characteristics and generate scalable layouts in the *cell placement phase*. The second entry refers to the non-deterministic approach [52] for placement and routing of each level sequentially. This algorithm is aimed at the USE clocking scheme. It does not specify cell placement for QCA cells, the non-deterministic attribute, lack of direction enforcement and, a heuristic to handle fan-in or fan-out proximity makes the approach not scalable. The latter is enforced in the third approach [21] with the simulated annealing meta-heuristic to find more feasible solutions. Still, only this modification is not sufficient to generate scalable layouts, which are feasible in the fourth approach [27], where the problem is solved with a divide-and-conquer strategy, with solution overlaps to each partition to generate the final layout. Lastly, the OGD problem is explored to yield scalable designs [25] for a range of clocking schemes. Moreover, this approach offers layout generation for QCA approaches and theoretical sequential circuit handling setting future research [26] to introduce a synchronization element to handle sequential circuits with high complexity.

VI Conclusion

In this paper, we have reviewed the current state-of-the-art approaches to solve the P&R problem for circuits of FCN technologies. We exhibited the challenges and methodologies for the pre-processing of circuit graphs. Moreover, we show examples of the initially proposed solutions for combinational circuits, with its pre-processing phases of *zone partitioning*, *zone placement* and *cell placement*. Right after, we discuss subsequent proposals for the arrangement of layouts in tiled clocking schemes with feedback support, with a range of examples for possible arrangements before technology synthesis, and conclude our work with a table that summarizes the targeted problems each proposal aimed to solve.

References

[1] R. H. Dennard, F. H. Gaensslen, H. Yu, V. L. Rideout, E. Bassous, and A. R. LeBlanc, "Design of ion-implanted mosfet's with very small physical dimensions," *IEEE Journal of Solid-State Circuits*, vol. 9, no. 5, pp. 256–268, 1974.

[2] G. E. Moore, "Cramming more components onto integrated circuits, reprinted from electronics, volume 38, number 8, april 19, 1965, pp.114 ff." *IEEE Solid-State Circuits Society Newsletter*, vol. 11, no. 3, pp. 33–35, 2006.

[3] T. N. Theis and H. . P. Wong, "The end of moore's law: A new beginning for information technology," *Computing in Science Engineering*, vol. 19, no. 2, pp. 41–50, 2017.

[4] N. S. Kim, T. Austin, D. Baauw, T. Mudge, K. Flautner, J. S. Hu, M. J. Irwin, M. Kandemir, and V. Narayanan, "Leakage current: Moore's law meets static power," *Computer*, vol. 36, no. 12, pp. 68–75, 2003.

[5] N. G. Anderson and S. Bhanja, *Field-Coupled Nanocomputing*, 1st ed. Springer-Verlag Berlin Heidelberg: Springer-Verlag Berlin Heidelberg, 2014.

[6] G. Csaba and W. Porod, "Simulation of field coupled computing architectures based on magnetic dot arrays," *Journal of Computational Electronics*, vol. 1, pp. 87–91, 07 2002.

[7] R. Cowburn and M. Welland, "Cowburn, r. p. & welland, m. e. room temperature magnetic quantum cellular automata. science 287, 1466-1468," pp. 1466–8, 03 2000.

[8] L. O. Luz, J. A. M. Nacif, R. Ferreira, and O. P. V. Neto, "Nmlib: A nanomagnetic logic standard cell library," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021, pp. 1–5.

[9] G. Csaba, P. Lugli, and W. Porod, "Power dissipation in nanomagnetic logic devices," pp. 346–348, 2004.

[10] C. S. Lent and P. D. Tougaw, "A device architecture for computing with quantum dots," pp. 541–557, 1997.

[11] J. Timler and C. Lent, "Power gain and dissipation in quantum-dot cellular automata," pp. 823–831, 01 2002.

[12] D. J. Griffiths and D. F. Schroeter, "Introduction to quantum mechanics," 2018.

[13] G. Pasandi and M. Pedram, "A dynamic programming-based, path balancing technology mapping algorithm targeting area minimization," in *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. IEEE, 2019, pp. 1–8.

[14] M. Vieira, M. Canesche, L. Bragança, J. Campos, M. Silva, R. Ferreira, and J. A. M. Nacif, "Reshape: A run-time dataflow hardware-based mapping for cgra overlays," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021, pp. 1–5.

[15] W. C. Oliveira, M. Canesche, L. Reis, J. Nacif, and R. Ferreira, "Design exploration of machine learning data-flows onto heterogeneous reconfigurable hardware," *XXI Simpósio em Sistemas Computacionais de Alto Desempenho*, 2020.

[16] O. Carvalho, M. Canesche, L. Reis, F. Torres, P. Jamieson, L. Silva, J. Nacif, and R. Ferreira, "A design exploration of scalable mesh-based fullypipelined accelerators," in *2020 International Conference on Field-Programmable Technology (ICFPT)*. IEEE, 2020.

[17] M. Walter, R. Wille, D. Große, F. Sill Torres, and R. Drechsler, "Placement and routing for tile-based field-coupled nanocomputing circuits is np-complete (research note)," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 15, pp. 1–10, 04 2019.

[18] M. Canesche, M. Menezes, W. Carvalho, F. Torres, P. Jamieson, J. A. Nacif, and R. Ferreira, "Traversal: A fast and adaptive graph-based placement and routing for cgras," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2020.

[19] R. Ferreira, L. Rocha, A. Santos, J. Nacif, S. Wong, and L. Carro, "A run-time graph-based polynomial placement and routing algorithm for virtual fpgas," in *2013 23rd International Conference on Field programmable Logic and Applications*. IEEE, 2013, pp. 1–8.

[20] R. Ferreira, L. Rocha, A. G. Santos, J. A. Nacif, S. Wong, and L. Carro, "A runtime fpga placement and routing using low-complexity graph traversal," *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, vol. 8, no. 2, pp. 1–16, 2015.

[21] R. E. Formigoni, R. S. Ferreira, and J. A. M. Nacif, "Ropper: A placement and routing framework for field-coupled nanotechnologies," pp. 1–6, 2019.

[22] W. Chung, B. Smith, and S. Lim, "Node duplication and routing algorithms for quantum-dot cellular automata circuits," *Circuits, Devices and Systems, IEE Proceedings -*, vol. 153, pp. 497 – 505, 11 2006.

[23] M. Bubna, S. Roy, N. Shenoy, and S. Mazumdar, "A layout-aware physical design method for constructing feasible qca circuits," pp. 243–248, 01 2008.

[24] A. Chaudhary, D. Chen, K. Whitton, M. Niemier, and R. Ravichandran, "Eliminating wire crossings for molecular quantum-dot cellular automata implementation," *IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers, ICCAD*, vol. 2005, pp. 565– 571, 12 2005.

[25] M. Walter, R. Wille, F. S. Torres, D. Große, and R. Drechsler, "Scalable design for field-coupled nanocomputing circuits," New York, NY, USA, p. 197–202, 2019. [Online]. Available: https://doi.org/10.1145/3287624.3287705

[26] R. Wille, M. Walter, F. Sill Torres, D. Große, and R. Drechsler, "Ignore clocking constraints: An alternative physical design methodology for field-coupled nanotechnologies," in *2019 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, 2019, pp. 651–656.

[27] G. Fontes, P. A. R. L. Silva, J. A. M. Nacif, O. P. V. Neto, and R. Ferreira, "Placement and routing by overlapping and merging qca gates," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1–5.

[28] V. Vankamamidi, M. Ottavi, and F. Lombardi, "Two-dimensional schemes for clocking/timing of qca circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 1, pp. 34–44, 2008.

[29] A. Imre, G. Csaba, L. Ji, A. Orlov, G. H. Bernstein, and W. Porod, "Majority logic gate for magnetic quantum-dot cellular automata," *Science*, vol. 311, no. 5758, pp. 205–208, 2006. [Online]. Available: http://science.sciencemag.org/content/311/5758/205

[30] L. Amarú, P. Gaillardon, and G. De Micheli, "Majority-inverter graph: A novel data-structure and algorithms for efficient logic optimization," in *2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2014, pp. 1–6.

[31] P. A. R. Silva, J. R. S. Alves, R. S. Ferreira, O. P. V. Neto, and J. A. M. Nacif, "A novel five-input multiple-function qca threshold gate," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2018, pp. 1–5.

[32] G. Csaba and W. Porod, "Behavior of nanomagnet logic in the presence of thermal noise," in *2010 14th International Workshop on Computational Electronics*, 2010, pp. 1–4.

[33] J. Atulasimha and S. Bandyopadhyay, "Bennett clocking of nanomagnetic logic using multiferroic single-domain nanomagnets," *Applied Physics Letters*, vol. 97, pp. 173 105–173 105, 10 2010.

[34] D. Bhowmik, L. You, and S. Salahuddin, "Spin hall effect clocking of nanomagnetic logic without magnetic field," *Nature nanotechnology*, vol. 9, 11 2013.

[35] S. Breitkreutz, J. Kiermaier, I. Eichwald, X. Ju, G. Csaba, D. Schmitt-Landsiedel, and M. Becherer, "Majority gate for nanomagnetic logic with perpendicular magnetic anisotropy," *IEEE Transactions on Magnetics*, vol. 48, no. 11, pp. 4336–4339, 2012.

[36] J. Nguyen, R. Ravichandran, S. K. Lim, and M. T. Niemier, "Global placement for quantum-dot cellular automata based circuits," 2003.

[37] F. Riente, G. Turvani, M. Vacca, M. R. Roch, M. Zamboni, and M. Graziano, "Topolinano: A cad tool for nano magnetic logic," pp. 1061–1074, 2017.

[38] V. Vankamamidi, M. Ottavi, and F. Lombardi, "Clocking and cell placement for qca," pp. 343 – 346, 07 2006.

[39] C. Campos, A. Marciano, O. Vilela Neto, and F. Sill Torres, "Use: A universal, scalable and efficient clocking scheme for qca," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, pp. 513–517, 02 2016.

[40] M. Goswami, A. Mondal, M. H. Mahalat, B. Sen, and B. K. Sikdar, "An efficient clocking scheme for quantum-dot cellular automata," *International Journal of Electronics Letters*, vol. 0, no. 0, pp. 1–14, 2019.

[41] F. deng, G.-J. Xie, x. cheng, z. zhang, and Y. Zhang, "Cfe: a convenient, flexible, and efficient clocking scheme for quantum-dot cellular automata," 11 2019.

[42] R. E. Formigoni, O. P. Vilela Neto, and J. A. M. Nacif, "Bancs: Bidirectional alternating nanomagnetic clocking scheme," pp. 1–6, 2018.

[43] L. Wang, G. Xie, R. Zhu, and C. Yu, "An optimized clocking scheme for nanoscale quantum-dot cellular automata circuit," in *2019 IEEE 14th International Conference on Nano/Micro Engineered and Molecular Systems (NEMS)*, 2019, pp. 336–341.

[44] F. Sill Torres, P. A. Silva, G. Fontes, J. A. Nacif, R. Santos Ferreira, O. Paranaiba Vilela Neto, J. Chaves, and R. Drechsler, "Exploration of the synchronization constraint in quantum-dot cellular automata," in *2018 21st Euromicro Conference on Digital System Design (DSD)*, 2018, pp. 642–648.

[45] C. M. Fiduccia and R. M. Mattheyses, "A linear-time heuristic for improving network partitions," in *19th Design Automation Conference*, 1982, pp. 175–181.

[46] D. A. Antonelli, D. Z. Chen, T. J. Dysart, X. S. Hu, A. B. Kahng, P. M. Kogge, R. C. Murphy, and M. T. Niemier, "Quantum-dot cellular automata (qca) circuit partitioning: Problem modeling and solutions," New York, NY, USA, p. 363–368, 2004. [Online]. Available: https://doi.org/10.1145/996566.996671

[47] X. Hu, R. G. Harber, and S. C. Bass, "Minimizing the number of delay buffers in the synchronization of pipelined systems," New York, NY, USA, p. 758–763, 1991. [Online]. Available: https://doi.org/10.1145/127601.127765

[48] K. Sugiyama, S. Tagawa, and M. Toda, “Methods for visual understanding of hierarchical system structures,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 11, no. 2, pp. 109–125, 1981.

[49] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of state calculations by fast computing machines,” *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[50] R. Ravichandran, N. Ladiwala, J. Nguyen, M. Niemier, and S. K. Lim, “Automatic cell placement for quantum-dot cellular automata,” New York, NY, USA, p. 332–337, 2004. [Online]. Available: https://doi.org/10.1145/988952.989033

[51] M. Walter, R. Wille, D. Große, F. Sill Torres, and R. Drechsler, “An exact method for design exploration of quantum-dot cellular automata,” pp. 503–508, 03 2018.

[52] A. Trindade, R. Ferreira, J. A. M. Nacif, D. Sales, and O. P. V. Neto, “A placement and routing algorithm for quantum-dot cellular automata,” pp. 1–6, 2016.

[53] M. Eiglsperger, S. Fekete, and G. Klau, “Orthogonal graph drawing,” *Drawing Graphs: Methods and Models, 121-171 (2001)*, vol. 2025, 01 1999.

[54] T. C. Biedl, “Improved orthogonal drawings of 3-graphs,” in *Proceedings of the 8th Canadian Conference on Computational Geometry*. Carleton University Press, 1996, p. 295–299.

[55] Rumi Zhang, K. Walus, Wei Wang, and G. A. Jullien, “A method of majority logic reduction for quantum cellular automata,” *IEEE Transactions on Nanotechnology*, vol. 3, no. 4, pp. 443–450, 2004.

[56] L. Amarú, P. Gaillardon, and G. De Micheli, “Majority-inverter graph: A new paradigm for logic optimization,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 5, pp. 806–819, 2016.

[57] K. Kong, Y. Shang, and R. Lu, “An optimized majority logic synthesis methodology for quantum-dot cellular automata,” *IEEE Transactions on Nanotechnology*, vol. 9, no. 2, pp. 170–183, 2010.

[58] P. A. R. L. Silva, O. P. V. Neto, and J. A. M. Nacif, “Toward nanometric scale integration: An automatic routing approach for nml circuits,” in *2019 32nd Symposium on Integrated Circuits and Systems Design (SBCCI)*, 2019, pp. 1–6.

[59] K. Walus and G. A. Jullien, “Design tools for an emerging soc technology: Quantum-dot cellular automata,” *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1225–1244, 2006.