electronics

MDPI

Article

Unified Graph Theory-Based Modeling and Control Methodology of Lattice Converters

Jingyang Fang

Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA; jf274@duke.edu

Abstract: Lattice converters combine the merits of both cascaded-bridge converters and multi-paralleled converters, leading to infinitely large current and voltage capabilities with modularity and scalability as well as small passive components. However, lattice converters suffer from complexity, which poses a serious threat to their widespread adoption. By use of graph theory, this article proposes a unified modeling and control methodology for various lattice converters, resulting in the satisfaction of their key control objectives, including selected inputs/outputs, desired voltages, current sharing, dynamic voltage balancing, and performance optimization. In addition, this article proposes a plurality of novel lattice converter topologies, which complement state-of-the-art options. Simulation and experimental results verify the effectiveness and superiority of the proposed methodology and lattice converters.

Keywords: algorithms; cascaded-bridge converters; graph theory; lattice converters; modular multi-level converters (MMCs); multi-paralleled converters; power converters

Citation: Fang, J. Unified Graph Theory-Based Modeling and Control Methodology of Lattice Converters. *Electronics* **2021**, *10*, 2146. https://doi.org/10.3390/electronics10172146

Academic Editor: M. Tariq Iqbal

Received: 6 July 2021
Accepted: 23 July 2021
Published: 3 September 2021

1. Introduction

Higher power is one of the primary drivers behind the research and development of novel power converters [1–4]. Typical high-power applications include, but are not limited to, high-voltage dc/ac transmissions (e.g., dc/ac power converters and circuit breakers) [5], motor drives [6], renewable generation (such as photovoltaic plants and wind farms) [7], power quality conditioners [8], and special power supplies (e.g., for fusion reactors [9], medical magnetic stimulators [10], and power amplifiers [11]). Along the trajectory of high-power converters, wide-bandgap devices are proven to be effective in pushing up switching frequencies, and hence the simplification of circuit structures [12]. However, they still suffer from inherent thermal limitations, thus necessitating novel circuit topologies.

In general, high power can be achieved via a high voltage, a large current, or both. To increase voltage ratings, we can connect simple yet basic power converters (known as submodules) in series, which collectively share a high voltage, giving rise to the invention of cascaded-bridge converters, such as the well-known cascaded H-bridge converter shown in Figure 1a [1]. As compared with other practically viable multilevel converters (e.g., neutral-point-clamped converters [2], T-type converters [13], and flying capacitor converters [4]), cascaded-bridge converters excel in modularity and scalability [1]. Since modular multilevel converters (MMCs) utilize cascaded-bridge converters as their arms, MMC arms inherently enjoy modularity and scalability [14]. However, from a macro-level point of view, MMCs follow conventional two-level converters, thereby losing modularity and scalability when each MMC arm is treated as a simple component [15]. However, the serial connection of submodules enables only voltage sharing rather than current sharing, and thus each submodule has to handle the entire load current.

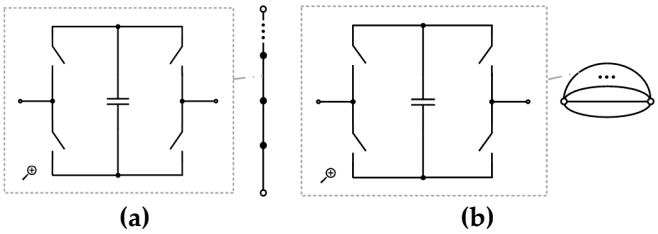

Figure 1. Schematic diagrams of (**a**) cascaded-bridge converters; (**b**) multi-paralleled converters.

Alternatively, we achieve current sharing by paralleling a plurality of submodules (see Figure 1b), whereby bulky power systems implement power sharing among synchronous generators and/or multiple power converters [16]. Notably, multi-paralleled converters (with identical submodules) also reap the benefits of modularity and scalability, thereby facilitating the design of individual submodules [17]. From a macro-level perspective, multi-paralleled converters are dual- to cascaded-bridge converters, and their voltages and currents correspond to each other. As such, all the submodules of multi-paralleled converters should undertake rated voltages. Noticeably, both cascaded-bridge and multi-paralleled converters connect submodules in one direction (1D), either in series or in parallel, leading to 1D modularity and scalability. However, they are incapable of handling both high voltages and large currents.

Two-dimensional (2D) connections (or extensions) of submodules are necessary when high power comprises a marriage of high voltages and large currents. This can be achieved through the parallelization of cascaded-bridge converters or the serial connection of multi-paralleled converters, yet at the expense of modularity and scalability from the macro-level perspective. Inspired by lattice graphs, which regularly extend themselves and tile the 2D plane or three-dimensional (3D) space, the author has proposed lattice converters (see Figure 2). Lattice converters replace each edge of a lattice graph with a converter submodule (e.g., an H-bridge converter), while each vertex represents a connection point of different converters [18]. Lattice converters feature high-voltage and large-current capabilities, which are up to infinity as the size of lattices grows. In addition to unparalleled voltage, current, and power ratings, lattice converters benefit from modularity and scalability as well as extremely small passive components [18].

Although serving as promising candidates for high-power applications, lattice converters suffer from their inherent circuit complexity, which implies a serious challenge to modeling and control. On top of this, as will be pointed out in this article, lattice converters feature various circuit topologies, thereby further complicating the design of controllers. Nevertheless, circuit complexity and variations also illuminate some of the potential flexibility and opportunities, which will be seized and disclosed by this article.

This article proposes a unified modeling and control methodology of lattice converters based on the classic algorithms of graph theory. The proposed methodology applies equally well to existing lattice converters and novel lattice converters proposed in this article. The proposed control methodology achieves the desired voltages and currents of selected input/output ports, dynamic voltage balancing, and performance optimizations. The remainder of this article is organized as follows. Section 2 presents the fundamentals and topologies of lattice converters. Section 3 introduces the proposed unified graph theory-based modeling methodology. Section 4 details the relevant graph theory control algorithms. Section 5 provides simulation and experimental results for verification purposes. Finally, Section 6 gives the concluding remarks.

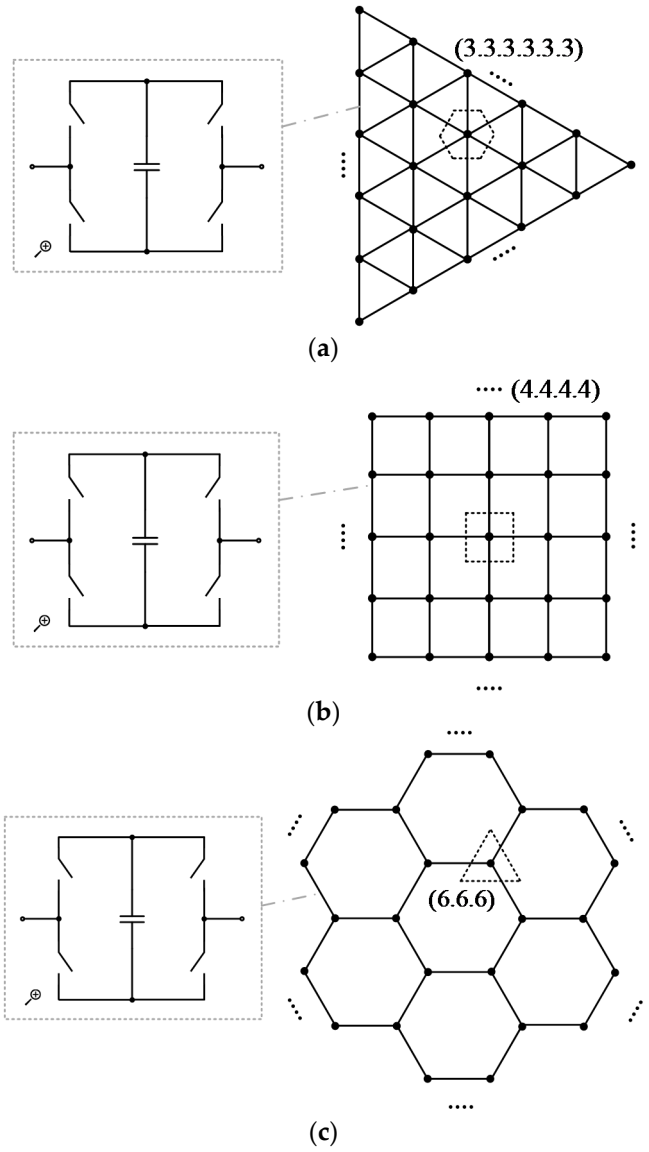

Figure 2. Three existing lattice converters based on regular tilings: (**a**) (3.3.3.3.3.3); (**b**) (4.4.4.4); (**c**) (6.6.6).

2. Topologies of Lattice Converters

This section introduces the fundamental knowledge and topologies of lattice converters. In addition, a plurality of novel lattice converters are proposed and described.

2.1. Fundamentals of Lattice Converters

Major motivations behind the invention of lattice converters include (1) high voltages, (2) large currents, and (3) modularity and scalability. As mentioned, cascaded-bridge or multi-paralleled converters achieve high voltages or large currents through 1D connections or extensions, respectively. To achieve both high voltages and large currents, lattice converters extend themselves in the 2D plane or 3D space.

Speaking of modularity and scalability, we duplicate a part (called a submodule) in order to form the entire converter circuit. Abstractly, modularity and scalability refer to translational symmetries, whose corresponding mathematical language is group theory [18]. Notably, wallpaper groups describe all plane symmetries, including translations, rotations, reflections, and their combinations, while space groups focus on space symmetries, as detailed in [18]. When restricted to modularity and scalability (namely, translational

symmetries) with a focus on topologies, we ignore the length of edges and use regular polygons as basic patterns. This translates modularity and scalability into tilings (also known as tessellations, pavings, or mosaics) by regular polygons, where basic patterns (or tiles) cover the plane or space without gaps or overlaps [19]. As a pioneer, J. Kepler investigated tiling more than four centuries ago in 1691, but his findings are still insightful today [19].

As for lattice converters, we follow the assumptions of (1) regular polygons, (2) edge-to-edge (where every edge is shared by precisely two polygons), and (3) convex polygons.

2.2. Topologies of Lattice Converters

Figure 2 shows the topologies of three existing lattice converters based on regular tiling, where each lattice converter employs exactly one type of polygon with identical vertices. It should be emphasized that the lattice converter is named by the combinations of polygons (abbreviated by the number of their edges) related to each vertex. For example, one vertex touches six equilateral triangles in Figure 2a, thereby named (3. 3. 3. 3. 3. 3). Similarly, (4. 4. 4. 4) and (6. 6. 6) represent the lattice converters formed by squares and regular hexagons, respectively. Due to multiple relationships between 2π (i.e., 360°) and the angles of polygons, there are only three lattice converters on the basis of regular tiling.

Furthermore, this article proposes eight novel lattice converters based on Archimedean tiling (see Figure 3), where each lattice converter contains at least two different types of polygons with identical vertices. For brevity, we remove the details of H-bridge submodules (represented by edges) in Figure 3. Among the proposed lattice converters, only (4. 6. 12) involves three different polygons, while the others consist of two different polygons. Notably, (3. 3. 3. 4. 4) and (3. 3. 4. 3. 4) consist of identical polygons yet with different connection sequences. Once again, these eight lattice converters exhaust all the possibilities.

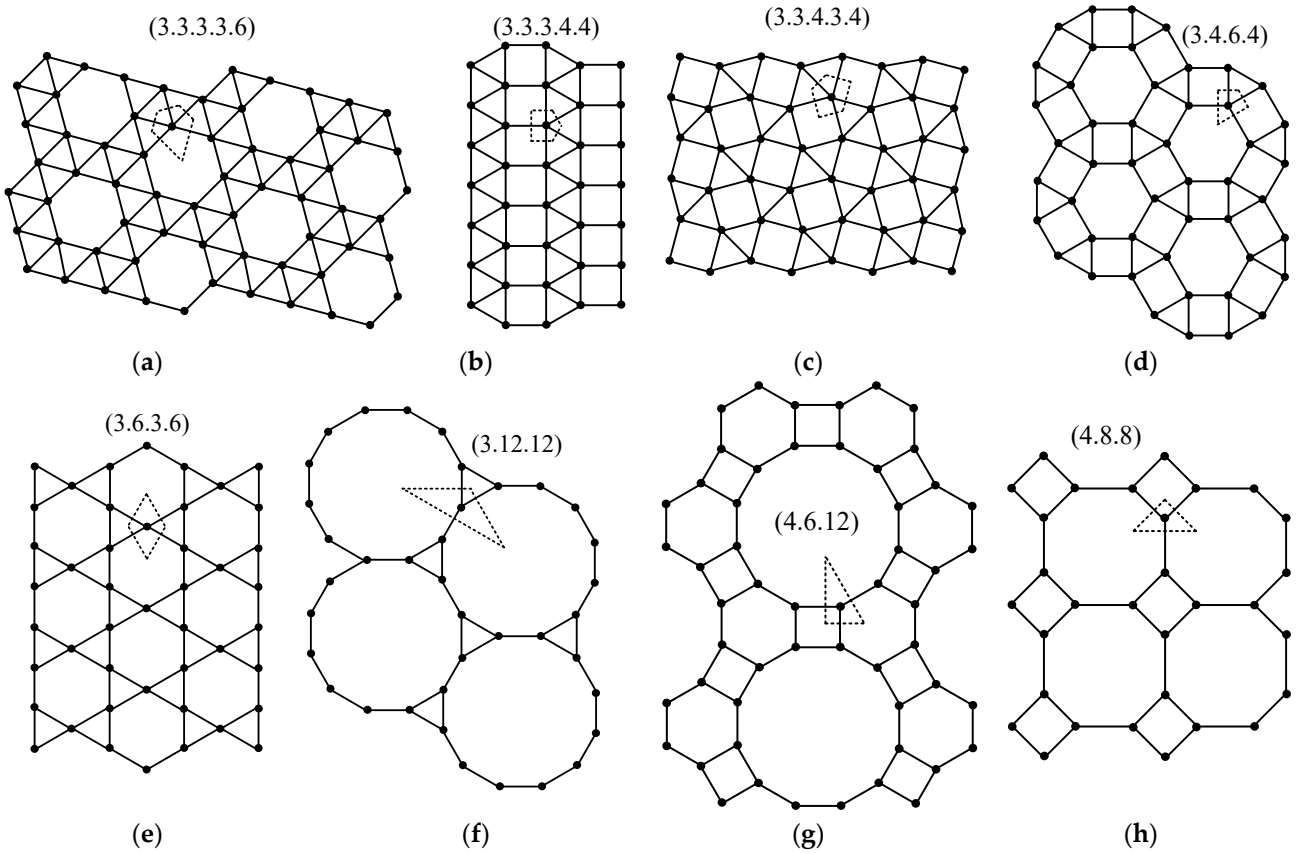

Figure 3. Proposed eight lattice converters based on Archimedean tilings: (**a**) (3.3.3.3.6); (**b**) (3.3.3.4.4); (**c**) (3.3.4.3.4); (**d**) (3.4.6.4); (**e**) (3.6.3.6); (**f**) (3.12.12); (**g**) (4.6.12); (**h**) (4.8.8).

Without the requirement that all the vertices are identical, we can derive novel lattice converters based on k-uniform tilings by dividing vertices into k individual groups. Along this research direction, there exist 20 lattice converters on the basis of 2-uniform tilings, while it is unknown how many k-uniform tilings exist in the case of $k \geq 3$ [19]. This aspect will not be considered here due to page limits.

In 3D and higher-dimensional spaces, a tiling of polyhedral is also known as a honeycomb. Under the assumptions of (1) regular polyhedra (being edge-transitive, vertex-transitive, and face-transitive), (2) face-to-face (where every face is shared by precisely two polyhedra), and (3) convex polyhedra (note that there are only five 3D regular convex polyhedra, i.e., the five Platonic solids: tetrahedron, cube, octahedron, dodecahedron, and icosahedron), only the cubic honeycomb remains, resulting in the proposed cubic lattice converter shown in Figure 4. If the requirements of regular polyhedra were relaxed to vertex-transitive (or uniform), it gives rise to 28 novel lattice converters based on the Archimedean honeycombs, which are excluded here [20]. Clearly, the variations in lattice converter topologies pose a challenge to their controller design.

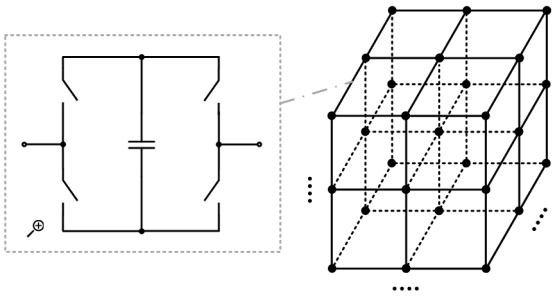

Figure 4. Proposed lattice converter based on the cubic honeycomb.

3. Proposed Graph-Theory-Based Modeling

This section introduces the background knowledge of graph theory. Subsequently, we propose the graph-theory-based modeling of lattice converters and their operating status.

3.1. Background Knowledge of Graph Theory

According to graph theory, we define a graph $\mathbf{G_x}$ by the following triplet:

$$\mathbf{G_x} \triangleq (\mathbf{N_x}, \mathbf{E_x}, \mathbf{A_x}), \tag{1}$$

where $\mathbf{N_x}$, $\mathbf{E_x}$, and $\mathbf{A_x}$ represent the set of nodes, set of edges, and adjacency matrix, respectively [21]. Notably, the set of edges (i.e., $\mathbf{E_x}$) contain the node pairs, in which the source node and destination node appear complimentarily and sequentially. On top of this, we define a path as an ordered sequence of edges that links two nodes [15].

3.2. Modeling Methodology of Lattice Converters

As mentioned, lattice converters feature many topological variations, which provide added incentives for the adoption of a unified modeling methodology for all lattice converters. By use of graph theory, we propose to model lattice converters by their respective lattice graphs.

Taking the 3×3 lattice converter (4. 4. 4. 4) in Figure 5 as an example, we first number the nodes sequentially from 0 (the left-bottom corner) to 8 (the right-top corner) so that the relevant set of nodes becomes

$$\mathbf{N_{base}} = \{0, 1, 2, 3, 4, 5, 6, 7, 8\} \tag{2}$$

According to Figure 5, we obtain the set of edges as

$$\mathbf{E_{base}} = \left\{ \begin{array}{l} (0,1),(0,3),(1,0),(1,2),(1,4),(2,1),(2,5),(3,0) \\ (3,4),(3,6),(4,1),(4,3),(4,5),(4,7),(5,2),(5,4) \\ (5,8),(6,3),(6,7),(7,4),(7,6),(7,8),(8,5),(8,7) \end{array} \right\}, \tag{3}$$

where each undirected edge incorporates two node pairs, e.g., (0, 1) and (1, 0). Furthermore, we derive the corresponding adjacency matrix $\mathbf{A_{base}}$ after replacing the elements related to (3) in a 9×9 zero matrix by one, namely,

$$\mathbf{A_{base}} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \tag{4}$$

In general, we model a $n \times n$ lattice converter through the following steps:

(1) Treat each vertex as a node and every converter as an edge.
(2) Number the nodes sequentially from 0 (e.g., the left-bottom corner) to $n^2 - 1$ (e.g., the right-top corner).
(3) Generate a $n^2 \times n^2$ zero matrix $\mathbf{A_{base}}$.
(4) Consider vertical edges and update $\mathbf{A_{base}}$ accordingly. Specifically, if the absolute value of $(i - j)$ equals n, $\mathbf{A_{base}}(i, j) = 1$, where $i, j \in 1, 2, \ldots, n^2$.
(5) Consider horizontal edges and modify $\mathbf{A_{base}}$ accordingly. Specifically, if abs$(i - j) = 1$ and abs(floor(i/n) − floor $(j/n)) \neq 1$, $\mathbf{A_{base}}(i, j) = 1$, where $i, j \in 1, 2, \ldots, n^2$. Note that the function floor () rounds the element to the nearest integer (less than or equal to that element).

As a result, we obtain a $n^2 \times n^2$ base matrix $\mathbf{A_{base}}$ (e.g., (4)) for further analyses.

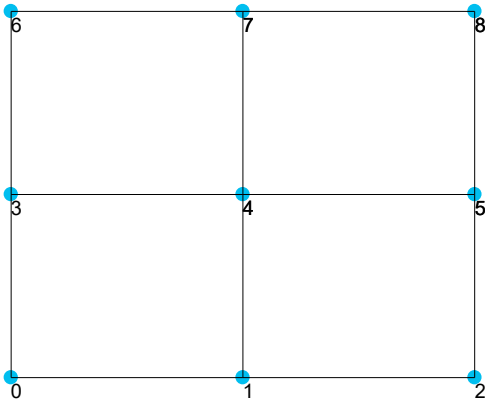

Figure 5. 3×3 lattice converter (4. 4. 4. 4).

3.3. Modeling of Operating Status

The lattice graph, together with its associated matrix, varies as the switching of submodules. In this article, we propose to model the operating status of each submodule via several different types of edges. Figure 6 illustrates the four operating statuses of H-bridge submodules. As shown, the weight of each edge represents the per unit voltage across two output terminals, where the rated submodule capacitor voltage serves as a nominal value. To differentiate the zero-output status from the off status, we use the

symbol ε to represent a small conduction voltage drop. As for asymmetrical half-bridge submodules, three statuses remain, including +1, 0, and ε [22]. In contrast, symmetrical half-bridge submodules maintain also three yet different statuses: +1, −1, and 0 [23].

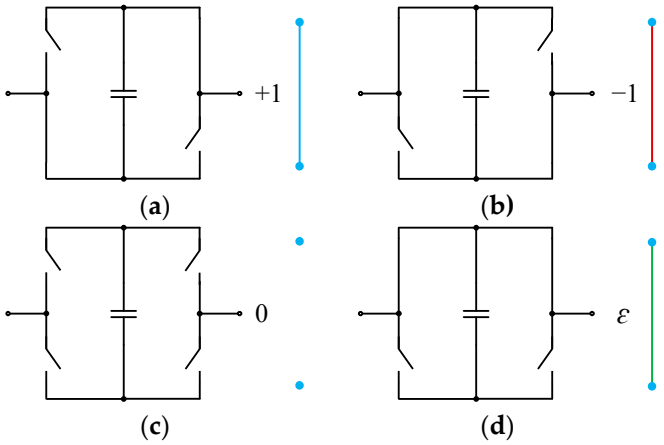

Figure 6. Operating status of lattice converter H-bridge submodules: (**a**) Positive output +1; (**b**) Negative output −1; (**c**) Off 0; (**d**) Zero-output ε.

4. Proposed Graph-Theory-Based Control Methodology

This section first presents the control objectives of lattice converters. Next, the proposed control methodology and algorithms of lattice converters are detailed.

4.1. Control Objectives

Generally, power converters should output (at least one pair of) desired voltages/currents [24]. For multilevel converters, submodules are expected to balance the state of charges of their energy storage units, e.g., capacitors and batteries [25,26]. On top of this, it is desirable to further optimize key performance indices, such as efficiency, dynamics, and costs, etc. [27]. In summary, the major control objectives of lattice converters comprise (1) selected input/output ports, (2) desired output voltage(s), (3) desired output current(s), (4) dynamic voltage balancing, and (5) performance optimizations.

4.2. Control Algorithms

Bearing the aforementioned control objectives in mind, we propose the graph-theory-based control methodology and algorithms of lattice converters. First, we select an output port consisting of two nodes. By referring to Figure 5, the nodes 8 and 0 are chosen as the positive and negative output terminals, respectively.

4.2.1. Searching for All Possible Paths between Two Selected Nodes

After node selection, we propose an important algorithm to search for all possible paths between two selected nodes. In brief, the searching of graph paths is possible through either Breadth First Search (BFS) or Depth First Search (DFS). The following **AllPathGeneration** function details the DFS algorithm of possible paths.

(1) Figure out the input and output variables. The input variables include the base matrix $\mathbf{A_{base}}$, source node s_{node}, destination node d_{node}, visited node vector $\mathbf{v_{node}} = []$, and an initial path number $p_{\text{no}} = 0$. The only output variable refers to the resultant path number p_{num}. In consequence, the overall function takes the form of *function* p_{num} = **AllPathGeneration** ($\mathbf{A_{base}}$, s_{node}, d_{node}, $\mathbf{v_{node}}$, p_{no}).

(2) Let $p_{\text{num}} = p_{\text{no}}$. Incorporate the source node into the visited node vector, namely, $\mathbf{v_{node}} = [\mathbf{v_{node}}\ s_{\text{node}}]$.

(3) If $s_{\text{node}} = d_{\text{node}}$, $p_{\text{num}} = p_{\text{num}} + 1$. Meanwhile, we print the path with all the visited nodes through the single path generation function **SinglePathGeneration** (n, $\mathbf{v_{node}}$).

(4) Update the indicator vector of the current node, i.e., $\mathbf{c_{pointer}} = \mathbf{A_{base}}(s_{\text{node}} + 1, :)$.
(5) For $i = 0{:}1{:}n^2 - 1$, continue to iterate the function **AllPathGeneration** with the next unvisited node. Specifically, if $[\mathbf{c_{pointer}}(i + 1) = 1]$ and $i \notin \mathbf{v_{node}}$, $p_{\text{num}} =$ **AllPathGeneration**$(\mathbf{A_{base}}, i, d_{\text{node}}, \mathbf{v_{node}}, p_{\text{num}})$.

Notably, we have employed another function, **SinglePathGeneration**, in the above steps. The purpose of this function lies in the generation and visualization of a single path, as detailed below:

(1) Figure out the input and output variables. The input variables include the length of lattice converters n and visited node vector $\mathbf{v_{node}}$. The output variable refers to the relevant graph adjacency matrix $\mathbf{A_{single}}$. As such, the function is written as *function* $\mathbf{A_{single}} =$ **SinglePathGeneration** $(n, \mathbf{v_{node}})$.
(2) Generate a $n^2 \times n^2$ zero matrix $\mathbf{A_{single}}$.
(3) Draw all the nodes at the background graph.
(4) Draw a path according to $\mathbf{v_{node}}$.
(5) Print the length of this path and its node vector $\mathbf{v_{node}}$.

Once again, we use the 3×3 lattice converter (4. 4. 4. 4) in Figure 5 as an example to test the function **AllPathGeneration** that searches for all possible paths between the positive (i.e., Node 8) and negative (i.e., Node 0) output terminals. Figure 7 illustrates the results of searching, where 12 possible paths are displayed, together with the lengths of paths, i.e., [0 1 2 5 4 3 6 7 8] with a length of 8, [0 1 2 5 4 7 8] with a length of 6, [0 1 2 5 8] with a length of 4, [0 1 4 3 6 7 8] with a length of 6, [0 1 4 5 8] with a length of 4, [0 1 4 7 8] with a length of 4, [0 3 4 1 2 5 8] with a length of 6, [0 3 4 5 8] with a length of 4, [0 3 4 7 8] with a length of 4, [0 3 6 7 4 1 2 5 8] with a length of 8, [0 3 6 7 4 5 8] with a length of 6, and [0 3 6 7 8] with a length of 4. As long as all the paths are available, it is easy to find the longest path and shortest path. Importantly, the longest path also implies the highest output voltage. Among the paths listed above, [0 1 2 5 4 3 6 7 8] and [0 3 6 7 4 1 2 5 8] are the longest paths, both with a length of 8.

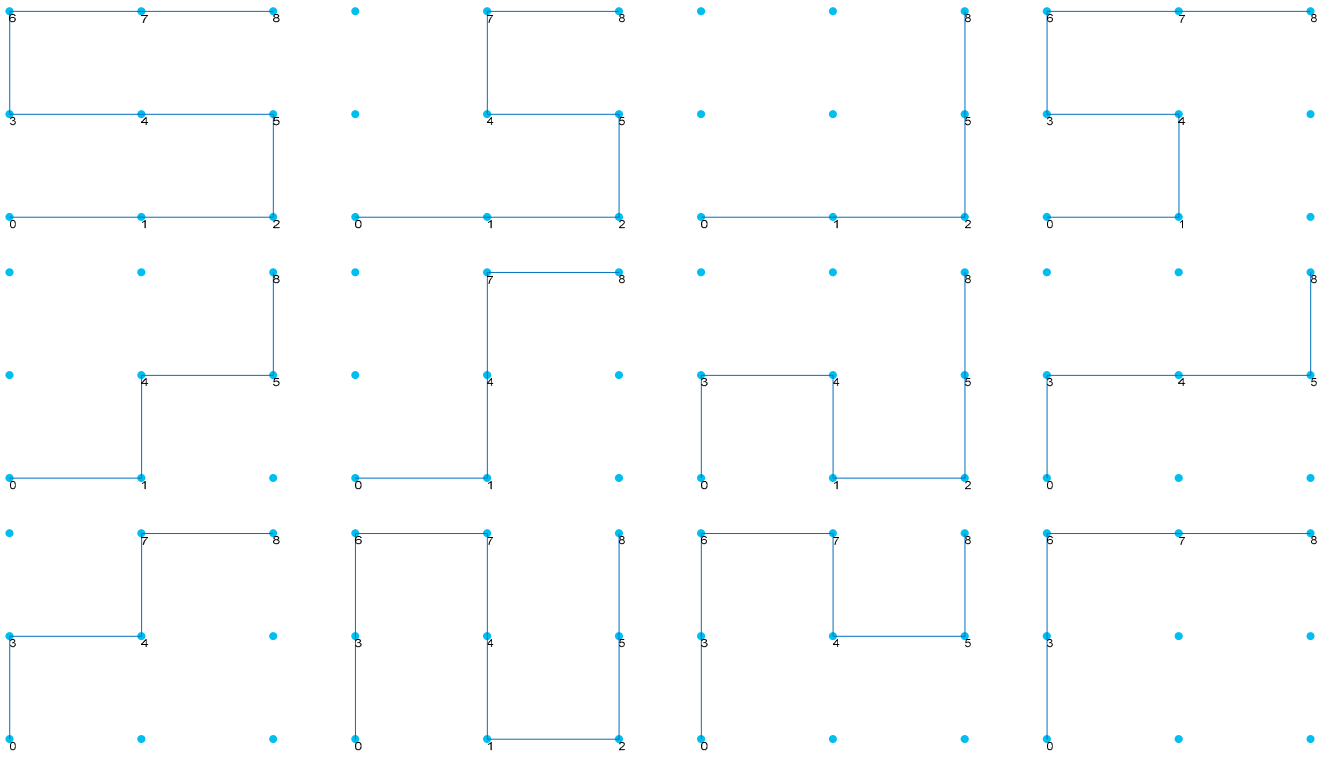

Figure 7. All possible paths between Node 8 and Node 0 of a 3×3 lattice converter (4. 4. 4. 4).

4.2.2. Searching for All Possible Sub-Paths of a Given Path

By use of the **AllPathGeneration** function, we obtain all possible paths between two selected nodes. Assuming that the output voltage requirement is $\pm k$ (i.e., $\pm kV_{\mathrm{dc}}$, where V_{dc} denotes the submodule dc voltage, and k represents a natural number), we can easily pick up those paths with a length of k to satisfy the voltage requirement. In addition, it should be emphasized that the paths with lengths greater than k can also satisfy the voltage requirement by making several submodules output a zero voltage. To achieve this, we propose a function **SubPathGeneration** that searches and visualizes all possible sub-paths affiliated to a given path, as detailed below.

(1) Figure out the input variables, including the length of lattice converters n, length of shorted sub-paths n_{sub}, shorted sub-path node vector $\mathbf{v_{sub}} = []$, path node vector $\mathbf{v_{node}}$, and initial node vector $\mathbf{v_{ini}} = \mathbf{v_{node}}$. The function takes the form of *function* **SubPathGeneration** (n, n_{sub}, $\mathbf{v_{sub}}$, $\mathbf{v_{node}}$, $\mathbf{v_{ini}}$).

(2) If $n_{\mathrm{sub}} = 0$, **SinglePathGeneration** (n, $\mathbf{v_{ini}}$). For i = 1:1:the length of $\mathbf{v_{sub}}$, draw a sub-path with shorted submodules via the function **PaintPath** (n, **[**$\mathbf{v_{ini}}$(find($\mathbf{v_{ini}} = \mathbf{v_{sub}}(i)$))**)** $\mathbf{v_{ini}}$(find($\mathbf{v_{ini}} = \mathbf{v_{sub}}(i)$) + 1)**]**), and $\mathbf{v_{sub}} = []$. Note that the find () function can yield the correct indices of $\mathbf{v_{ini}}$.

(3) If $n_{\mathrm{sub}} \neq 0$, find the next sub-path. For i = 1:1:the length of $\mathbf{v_{node}} - 1$, $\mathbf{v_{sub1}} = [\mathbf{v_{sub}}\ \mathbf{v_{node}}(i)]$, and $\mathbf{v_{node1}} = []$. For j = 1:1:the length of $\mathbf{v_{node}} - i$, $\mathbf{v_{node1}} = [\mathbf{v_{node1}}\ \mathbf{v_{node}}(i + j)]$. Finally, iterate in the loop of i: **SubPathGeneration** (n, $n_{\mathrm{sub}} - 1$, $\mathbf{v_{sub1}}$, $\mathbf{v_{node1}}$, $\mathbf{v_{ini}}$).

Notably, another function, **PaintPath**, appears in the above steps. This function aims to highlight the shorted submodules of a sub-path:

(1) Figure out the input variables, including the length of lattice converters n and shorted sub-path node vector $\mathbf{v_{sub}}$. The function is described by *function* **PaintPath** (n, $\mathbf{v_{sub}}$).

(2) Use the line () function to draw the shorted sub-path.

Figure 8 presents all the sub-paths of a given path yielded by the function **SubPathGeneration** (3, 2, [], [0 1 2 5 8], [0 1 2 5 8]). Clearly, these sub-paths all feature a length of 2, thereby providing more options in the case of $k = \pm 2$.

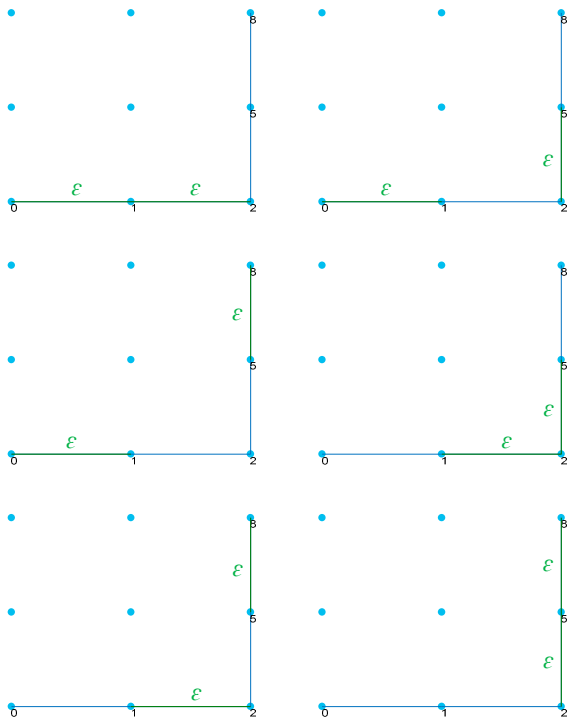

Figure 8. All the length-2 sub-paths of the path [0 1 2 5 8].

4.2.3. Merging Nodes for Enhanced Current Capabilities

Thus far, we have introduced how to find all the available paths that satisfy voltage requirements. Moreover, it is necessary to parallel several paths according to the current requirement. However, as each node has a limited number of neighboring nodes, the current capability of lattice converters is limited. To enhance current capabilities, we can merge nodes together, where the involved submodules output zero voltages. As an example, Figure 9 shows the principle of node merging, where four nodes are lumped into one with a doubled current capability [18]. Accordingly, the dimension of the base matrix $\mathbf{A_{base}}$ shrinks from 4×4 to 2×2.

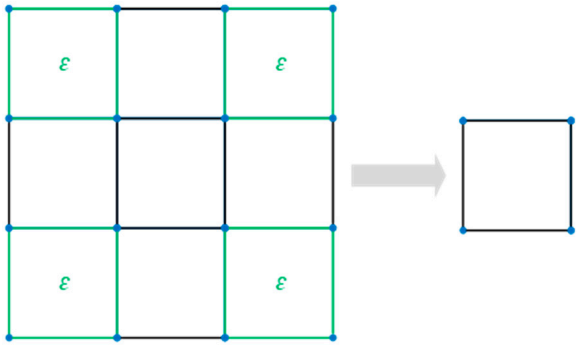

Figure 9. Node merging for enhanced current capabilities.

4.2.4. Removing Selected Nodes and/or Edges in the Case of Multiple Input/Output Ports

The proposed lattice converters allow operations with multiple input/output ports. When adding a new port (e.g., two new terminals) and the associated paths, we should guarantee that they have no overlaps with existing ones. This can be achieved by modification of the base matrix $\mathbf{A_{base}}$ via the two functions **NodeBreak** and **EdgeBreak**, respectively. The **NodeBreak** function is detailed as follows.

(1) Figure out the input and output variables. The base matrix $\mathbf{A_{base}}$ and the broken node b_{node} serve as the input variables, while $\mathbf{A_{base}}$ also acts as the output variable. The function is defined as *function* $\mathbf{A_{base}}$ = **NodeBreak** ($\mathbf{A_{base}}$, b_{node}).
(2) For i = 1:1:the number of nodes in $\mathbf{A_{base}}$, $\mathbf{A_{base}}(b_{\text{node}} + 1, i) = 0$, and $\mathbf{A_{base}}(i, b_{\text{node}} + 1) = 0$.

Similarly, the **EdgeBreak** function is given as follows.

(1) Figure out the input and output variables. The input variables comprise the base matrix $\mathbf{A_{base}}$, the source node s_{node}, and the destination node d_{node}. $\mathbf{A_{base}}$ is also the output variable. The function can be written as *function* $\mathbf{A_{base}}$ = **EdgeBreak**($\mathbf{A_{base}}$, s_{node}, d_{node}).
(2) $\mathbf{A_{base}}(s_{\text{node}} + 1, d_{\text{node}} + 1) = 0$, and $\mathbf{A_{base}}(d_{\text{node}} + 1, s_{\text{node}} + 1) = 0$.

In the case of multiple input/output ports, we search for the paths related to the first port by use of the functions **AllPathGeneration** and **SubPathGeneration** as before. Next, we employ the **NodeBreak** or **EdgeBreak** functions to remove existing nodes or edges, respectively. Finally, we can search for paths with updated base matrices $\mathbf{A_{base}}$ again. For instance, Figure 10 illustrates the two possible paths between Node 8 and Node 0 with Node 4 removed.

4.2.5. Dynamic Submodule Voltage Balancing through Parallelization

The balancing of capacitor voltage (or state-of-the charges for batteries) serves as one major control objective of multilevel converters [26,28]. For multilevel converters with parallel connectivity or switched-capacitor converters, hardware-based voltage balancing can be achieved by parallelization of submodules, leading to the great simplification of control efforts [15]. Fortunately, lattice converters allow parallel connectivity, and hence

hardware-based dynamic voltage balancing. Specifically, we propose the four statuses of voltage balancing, as shown in Figure 11. Clearly, any three statuses can collectively achieve dynamic voltage balancing.

4.2.6. Performance Optimizations

Furthermore, we select from the available paths or status to optimize the performance of lattice converters, such as efficiency. To increase system efficiency, we maintain the operating status of submodules unchanged as much as possible so that fewer switches will operate. This can be achieved by comparing the matrices of the current and next status using the function **MatrixCompare**, as described below.

(1) Figure out the input and output variables. The input variables refer to two matrices $\mathbf{A_1}$ and $\mathbf{A_2}$, and the output variable quantifies their differences $n_{\text{diff}} = 0$. The function takes the form of *function* n_{diff} = **MatixCompare** ($\mathbf{A_1}$, $\mathbf{A_2}$).

(2) Compare the corresponding elements of $\mathbf{A_1}$ and $\mathbf{A_2}$. If unequal, $n_{\text{diff}} = n_{\text{diff}} + 1$.

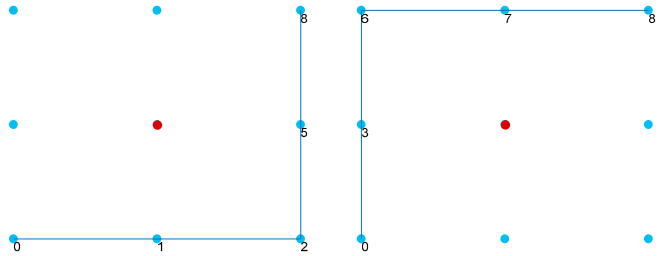

Figure 10. Two paths between Node 8 and Node 0 with Node 4 removed.

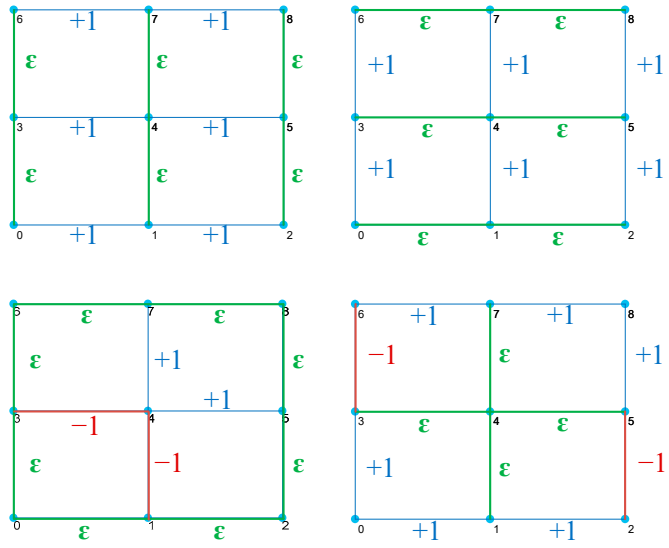

Figure 11. Statuses of dynamic voltage balancing.

5. Simulation and Experimental Results

This section provides the simulation and experimental results of lattice converters for verification purposes.

5.1. Simulation Results

We performed simulations under the Matlab/Simulink environment (R2016b) with the parameters listed in Table 1, where the 3 × 3 lattice converter shown in Figure 5 is involved.

To demonstrate the effectiveness of the proposed unified graph-theory-based control methodology, Figure 12 illustrates the simulation waveforms of the load voltage and current in the case of harmonic elimination modulation. Clearly, nine programmed output voltage levels with deliberately shifted phase angles (i.e., $\alpha_1 = 0.8572^\circ$, $\alpha_2 = 24.8571^\circ$, $\alpha_3 = 35.1429^\circ$, and $\alpha_4 = 60.8571^\circ$ [29]) are alternatively used to remove the third, fifth, seventh, and ninth harmonics. Notably, the 3rd and 12th paths in Figure 7, together with their sub-paths, provide sufficient options.

Table 1. Simulation parameters of lattice converters.

Descriptions	Symbols	Values
Lattice size	$n \times n$	3×3
Number of submodules	$2n \times (n-1)$	12
Load inductance	L_{load}	3 mH
Load capacitance	C_{load}	1 µF
Load resistance	R_{load}	20 Ω
Module dc voltage	V_{dc}	200 V
Module inductance	L_{module}	2 µH
Module resistance	R_{module}	0.1 Ω
Switching frequency	f_{sw}	5 kHz

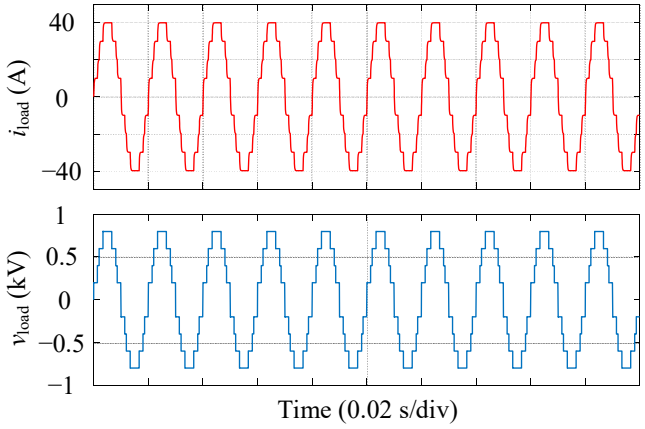

Figure 12. Simulation results of the lattice converter with harmonic elimination modulation.

Moreover, with the 3rd and 12th paths in Figure 7, Figure 13 shows the results of the pulse-width-modulated (PWM) lattice converter. Comparing Figure 13a with Figure 13b, it is noted that the lattice converter allows not only voltage sharing but also current sharing among submodules.

Figure 14 provides the simulation results of multiple outputs, where the nodes related to the 12th path in Figure 7 (or the second path in Figure 10) serve as ac nodes, while the remaining four nodes enable a dc output with two paralleled paths. As validated, lattice converters enable successful operations with multiple programmed input/output ports.

Figure 15 presents the simulation results of dynamic voltage balancing, where the statuses of lattice converters in Figure 11 are alternatively employed. Thanks to the proposed hardware-based voltage balancing, all the submodule dc voltages equalize within one fundamental period (i.e., 0.02 s). The above simulation results verify the effectiveness and feasibility of the proposed control methodology and lattice converters.

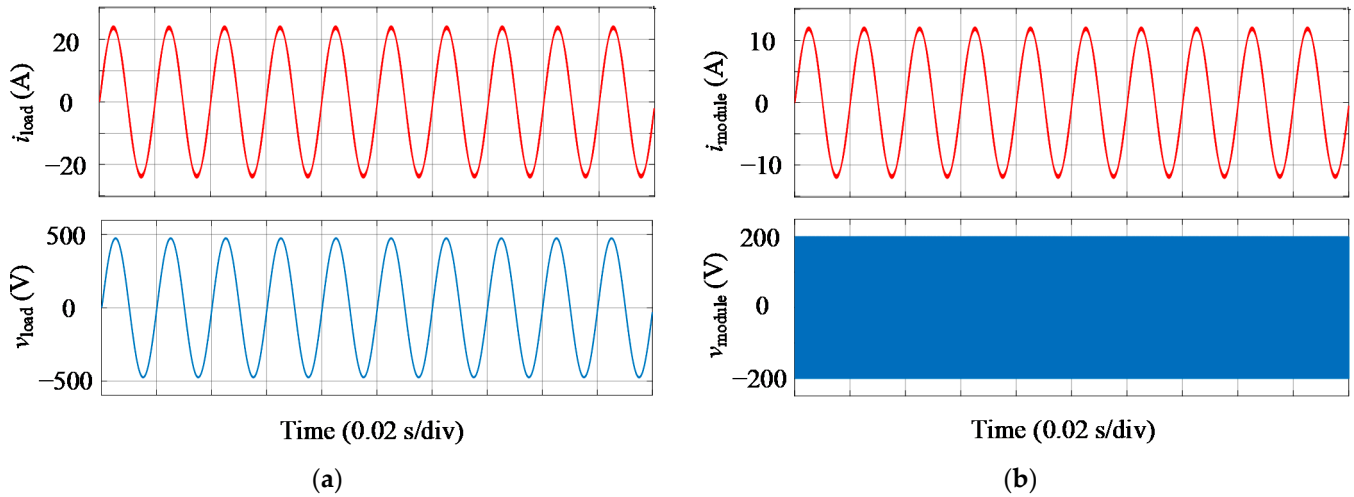

Figure 13. Simulation results of the lattice converter with pulse-width modulation: (**a**) The load current iload and load voltage vload; (**b**) The module current imodule and module voltage vmodule.

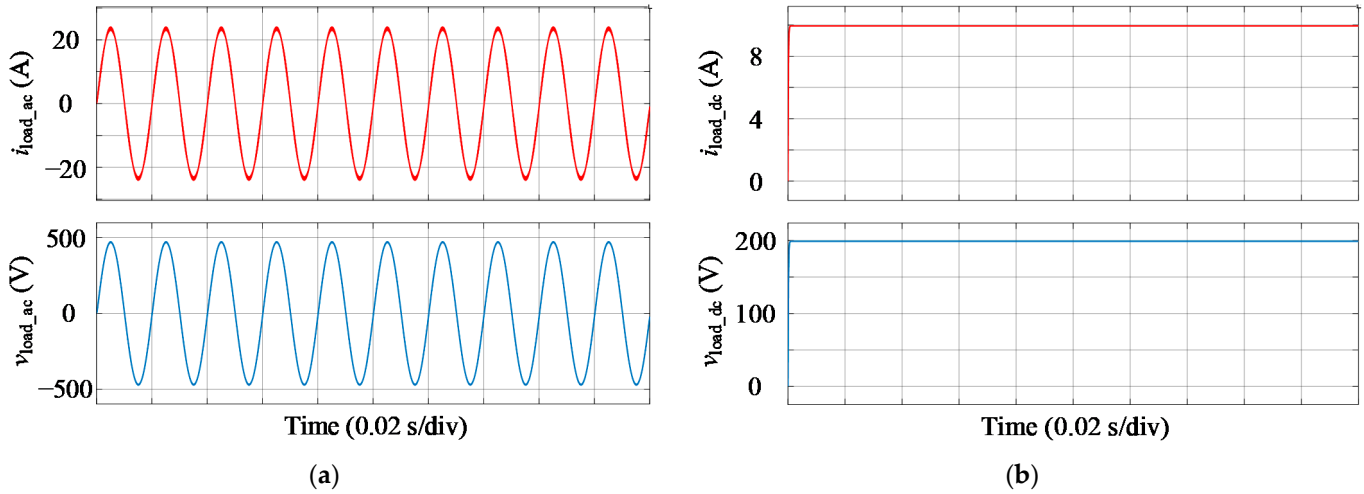

Figure 14. Simulation results of the lattice converter with multiple output ports: (**a**) ac output port; (**b**) dc output port.

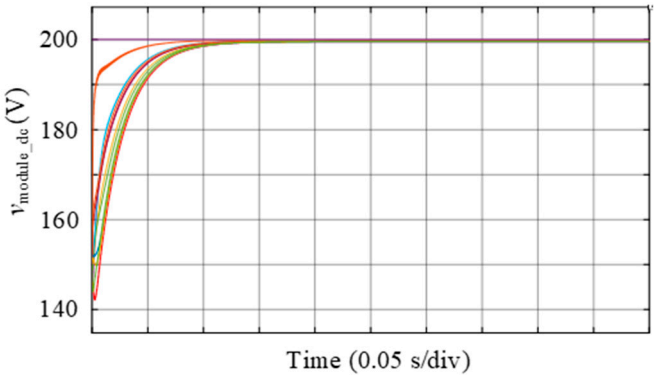

Figure 15. Simulation results of the lattice converter with dynamic voltage balancing.

5.2. Experimental Results

To further validate the effectiveness of lattice converters, we built an experimental setup, shown in Figure 16, where the lattice converter is regulated by a dSPACE Microlabbox controller. Figure 17 illustrates the experimental results of the lattice converter with harmonic elimination modulation, where the third, fifth, seventh, and ninth harmonics are

eliminated. Figure 18 presents the experimental results of the lattice converter with pulse-width modulation. Clearly, current sharing among submodules is achieved. Figure 19 shows the case of multiple outputs, including one ac output and one dc output. As shown, both output ports work properly as desired. Finally, Figure 20 shows the dynamics of dc voltage balancing. It is clear that we achieved a very fast voltage balancing. The above experimental results demonstrate the effectiveness and superiority of the proposed lattice converter modeling and control methodology.

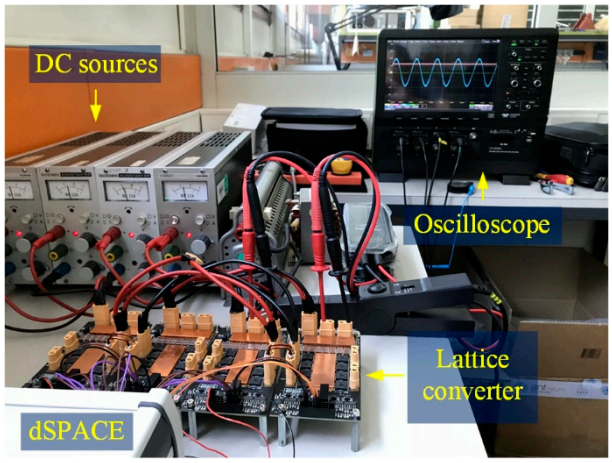

Figure 16. Photo of the experimental setup.

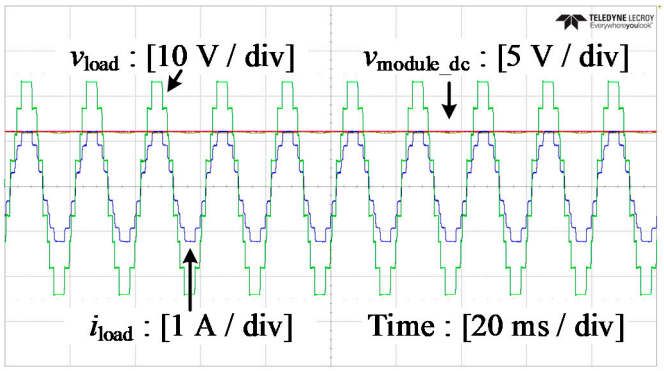

Figure 17. Experimental results of the lattice converter with harmonic elimination modulation.

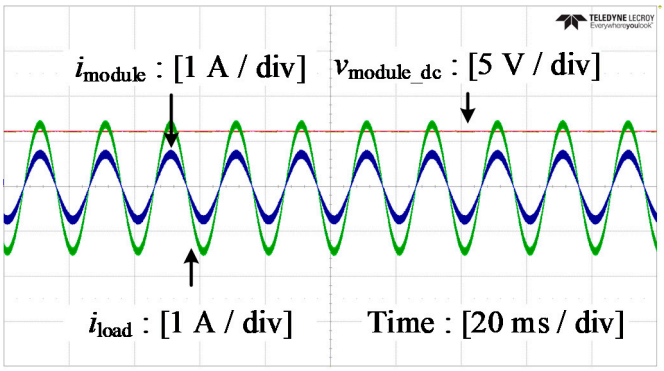

Figure 18. Experimental results of the lattice converter with pulse-width modulation.

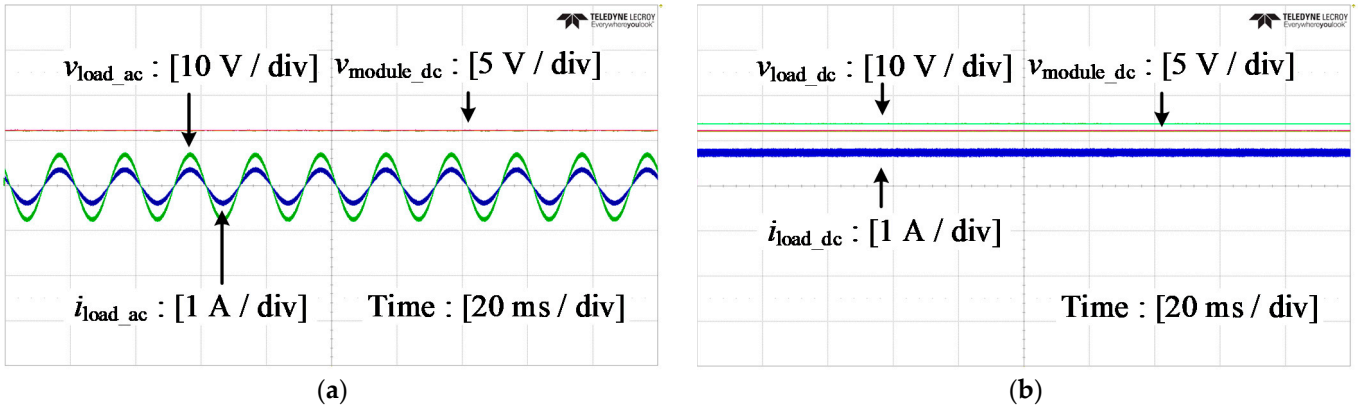

Figure 19. Experimental results of the lattice converter with multiple outputs: (**a**) ac output port; (**b**) dc output port.

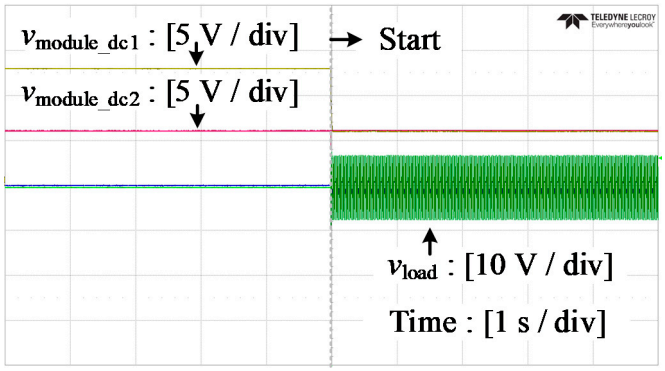

Figure 20. Experimental results of the lattice converter with dynamic voltage balancing.

6. Conclusions

This article has proposed a unified modeling and control methodology that applies to both existing and newly proposed lattice converters. The proposed methodology models each lattice converter as a base lattice graph. Moreover, we model the operating status of submodules as different edges for analyses and syntheses. On top of this, the article proposes lattice converter controllers according to graph-theory-based algorithms. As a result, the control objectives of desired voltages, current sharing, programmed multiple inputs/outputs, dynamic voltage balancing, and performance optimization are guaranteed. Finally, the simulation and experimental results validate the feasibility of the proposed modeling and control methodology as well as the benefits of lattice converters, including high voltage and large current capabilities, micro- and macro-level modularity and scalability, and small passive components.

Funding: This research received no external funding.

Conflicts of Interest: The author declare no conflict of interest

References

1. Baker, R.H.; Bannister, L.H. Electric Power Converter. U.S. Patent 3,867,643, 18 February 1975.
2. Nabae, A.; Takahashi, I.; Akagi, H. A New Neutral-Point-Clamped PWM Inverter. *IEEE Trans. Ind. Appl.* **1981**, *IA-17*, 518–523. [CrossRef]
3. Meynard, T.; Foch, H. Multi-level conversion: High voltage choppers and voltage-source inverters. In Proceedings of the PESC'92 Record. 23rd Annual IEEE Power Electronics Specialists Conference, Toledo, Spain, 29 June–3 July 1992; pp. 397–403.
4. Marquardt, R.; Lesnicar, A. A new modular voltage source inverter topology. In Proceedings of the European Conference on Power Electronics and Applications, Bologna, Italy, 23–26 June 2003; pp. 1–10.
5. Flourentzou, N.; Agelidis, V.G.; Demetriades, G.D. VSC-Based HVDC Power Transmission Systems: An Overview. *IEEE Trans. Power Electron.* **2009**, *24*, 592–602. [CrossRef]

6. Lai, J.; Peng, F.Z. Multilevel converters-a new breed of power converters. *IEEE Trans. Ind. Appl.* **1996**, *32*, 509–517. [CrossRef]
7. Villanueva, E.; Correa, P.; Rodriguez, J.; Pacas, M. Control of a Single-Phase Cascaded H-Bridge Multilevel Inverter for Grid-Connected Photovoltaic Systems. *IEEE Trans. Ind. Electron.* **2009**, *56*, 4399–4406. [CrossRef]
8. Peng, F.Z.; Lai, J. Dynamic performance and control of a static var generator using cascade multilevel inverters. *IEEE Trans. Ind. Appl.* **1996**, *32*, 509–517. [CrossRef]
9. Yue, L.; Lee, I.; Yao, X. Tokamak vertical stability coil power supply based on modular multilevel converter. In Proceedings of the 2016 IEEE International Power Modulator and High Voltage Conference (IPMHVC), San Francisco, CA, USA, 6–9 July 2016; pp. 447–452.
10. Goetz, S.M.; Pfaeffl, M.; Huber, J.; Singer, M.; Marquardt, R.; Weyh, T. Circuit topology and control principle for a first magnetic stimulator with fully controllable waveform. In Proceedings of the 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, San Diego, CA, USA, 28 August–1 September 2012; pp. 4700–4703.
11. Vasic, M.; Garcia, O.; Oliver, J.A.; Alou, P.; Diaz, D.; Cobos, J.A. Multilevel power supply for high-efficiency RF amplifiers. *IEEE Trans. Power Electron.* **2010**, *25*, 1078–1089. [CrossRef]
12. Fang, J.; Yang, X.; Zhang, L.; Tang, Y. An optimal digital pulse-width-modulated dither technique to enhance the resolution of high frequency power converters. *IEEE Trans. Power Electron.* **2017**, *32*, 7222–7232. [CrossRef]
13. Schweizer, M.; Kolar, J.W. Design and Implementation of a Highly Efficient Three-Level T-Type Converter for Low-Voltage Applications. *IEEE Trans. Power Electron.* **2013**, *28*, 899–907. [CrossRef]
14. Marquardt, R. Modular Multilevel Converters: State of the Art and Future Progress. *IEEE Power Electron. Mag.* **2018**, *5*, 24–31. [CrossRef]
15. Fang, J.; Blaabjerg, F.; Liu, S.; Goetz, S. A Review of Multilevel Converters with Parallel Connectivity. *IEEE Trans. Power Electron.* **2021**, in press. [CrossRef]
16. Kundur, P. *Power System Stability and Control*; McGraw-Hill: New York, NY, USA, 1994.
17. Agorreta, J.L.; Borrega, M.; López, J.; Marroyo, L. Modeling and Control of *N*N-Paralleled Grid-Connected Inverters With LCL Filter Coupled Due to Grid Impedance in PV Plants. *IEEE Trans. Power Electron.* **2010**, *26*, 770–785. [CrossRef]
18. Fang, J.; Goetz, S.M. Symmetries in power electronics. In Proceedings of the IEEE Energy Conversion Congress and Exposition, Vancouver, BC, Canada, 10–14 October 2021.
19. Grünbaum, B.; Shephard, G. A Review of Multilevel Converters with Parallel Connectivity. *Math. Mag.* **1977**, *50*, 227–247. [CrossRef]
20. Coxeter, H.S.M. *Regular Polytopes*; Dover Publications: New York, NY, USA, 1973.
21. Dorfler, F.; Simpson-Porco, J.W.; Bullo, F. Breaking the hierarchy: Distributed control &economic optimality in microgrids. *IEEE Trans. Control Netw. Syst.* **2016**, *3*, 241–253.
22. Fang, J. *More-Electronics Power Systems: Power Quality and Stability*; Springer Science and Business Media LLC: Berlin, Germany, 2021; p. 8.
23. Fang, J.; Li, Z.; Goetz, S.M. Multilevel Converters With Symmetrical Half-Bridge Submodules and Sensorless Voltage Balance. *IEEE Trans. Power Electron.* **2021**, *36*, 447–458. [CrossRef]
24. Erickson, R.W.; Maksimovic, D. *Fundamentals of Power Electronics*; Springer: New York, NY, USA, 2001.
25. Fang, J.; Yang, S.; Wang, H.; Tashakor, N.; Goetz, S.M. Reduction of MMC Capacitances Through Parallelization of Symmetrical Half-Bridge Submodules. *IEEE Trans. Power Electron.* **2021**, *36*, 8907–8918. [CrossRef]
26. Goetz, S.M.; Li, Z.; Liang, X.; Zhang, C.; Lukic, S.; Peterchev, A.V. Control of Modular Multilevel Converter With Parallel Connectivity—Application to Battery Systems. *IEEE Trans. Power Electron.* **2017**, *32*, 8381–8392. [CrossRef]
27. Fang, J.; Tang, Y.; Li, H.; Li, X. A Battery/Ultracapacitor Hybrid Energy Storage System for Implementing the Power Management of Virtual Synchronous Generators. *IEEE Trans. Power Electron.* **2018**, *33*, 2820–2824. [CrossRef]
28. Fang, J.; Li, Z.; Goetz, S.M.; Yang, S.; Wang, H. H-Bridge MMCs with Symmetrical Half-Bridge Submodules. In Proceedings of the 2020 IEEE Energy Conversion Congress and Exposition (ECCE), Portland, OR, USA, 11–15 October 2020; pp. 2904–2908.
29. Holmes, D.G.; Lipo, T.A. *Pulse Width Modulation for Power Converters: Principles and Practice*; John Wiley & Sons: Hoboken, NJ, USA, 2003.