Pegasus: The second connectivity graph for large-scale quantum annealling hardware

Nike Dattani[*]
Harvard-Smithsonian Center for Astrophysics

Szilard Szalay[†]
Strongly Correlated Systems "Lendület" Research Group,
Wigner Research Centre for Physics of the Hungarian Academy of Sciences,
29-33, Konkoly-Thege Miklós út, Budapest, H-1121, Hungary

Nicholas Chancellor[‡]
Durham University

Pegasus is a graph which offers substantially increased connectivity between the qubits of quantum annealing hardware compared to the graph Chimera. It is the first fundamental change in the connectivity graph of quantum annealers built by D-Wave since Chimera was introduced in 2011 for D-Wave's first commercial quantum annealer. In this article we describe an algorithm which defines the connectivity of Pegasus and we provide what we believe to be the best way to graphically visualize Pegasus in 2D and 3D in order to see which qubits couple to each other. As Supplemental Material, we provide open source codes for generating Pegasus graphs.

The 128 qubits of the first commercial quantum annealer (D-Wave One, released in 2011) were connected by a graph called Chimera, which is rather easy to describe: A 2D array of $K_{4,4}$ graphs, with one partition of each $K_{4,4}$ being also connected to the same corresponding partition on the $K_{4,4}$ unit cell above it, and the other partition being connected to the same corresponding partition on the $K_{4,4}$ to the right of it (see Figure 1). The degree of the graph is six, since each qubit couples to four qubits within its $K_{4,4}$ unit cell, and to one qubit in a $K_{4,4}$ above it and one qubit in a $K_{4,4}$ to the right of it. All commercial quantum annealers built to date follow this graph connectivity, with just larger and larger numbers of $K_{4,4}$ unit cells (See Table 1).

Figure 1. The Chimera graph, with open edges to show that the pattern repeats.

fig_ChimeraLL_crop.pdf

Table I. Chimera graphs in all commercial quantum computers to date.

	Array of $K_{4,4}$ unit cells	Total # of qubits
D-Wave One	4×4	128
D-Wave Two	8×8	512
D-Wave 2X	12×12	1152
D-Wave 2000Q	16×16	2048

In 2018, D-Wave announced the construction of a (not yet commercial) quantum annealer with a greater connectivity than Chimera offers, and a publicly available program (NetworkX) which allows users to generate Pegasus graphs of arbitrary size. However, no explicit description of the graph connectivity in Pegasus has been published yet, so we have had to apply the process of reverse engineering to determine it, and the following section describes our algorithm for generating Pegasus.

I. ALGORITHM FOR GENERATING PEGASUS

Start with K layers of Chimera graphs, each being an $I \times J$ array of $K_{4,4}$ unit cells (therefore we have an $I \times J \times K$ array of $K_{4,4}$ unit cells). The indices (i, j, k) will be used to describe the location of each unit cell along the indices corresponding to the dimension picked from (I, J, K). (In the concrete realization by DWave, $I = J =?$, and $K = 3$.) Each $K_{4,4}$ cell has two parts, labeled $l \in \{0, 1\}$, so that there are 4 qubits (vertices) for every (i, j, k, l). We will arbitrarily label these 4 qubits using two more labels: $(m, n) \in \{0, 1\}^2$. Therefore every qubit is associated with 6 indices: (i, j, k, l, m, n), with their ranges and descriptions given in Table 2.

[*] n.dattani@cfa.harvard.edu
[†] szalay.szilard@wigner.mta.hu
[‡] nicholas.chancellor@durham.ac.uk

Table II. Indices used to describe each qubit (vertex) in Pegasus

i	0 to $I-1$	Row within a Chimera layer
j	0 to $J-1$	Column within a Chimera layer
k	0 to $K-1$	Chimera layer
l	$0, 1$	Bi-partition within $K_{4,4}$
m	$0, 1$	Top partition within each bi-partition of $K_{4,4}$
n	$0, 1$	Bottom partition within each bi-partition of $K_{4,4}$

The $K_{4,4}$ Chimera unit cells are given by the equation

$$(i, j, k, l = 0, m, n) \to (i, j, k, l = 1, m', n'). \quad (1)$$

(In our convention, such an equation is understood to all valid indices.) The vertical and horizontal couplings connect unit cells being nearest neighbors with respect to the (i, j) coordinates,

$$(i, j, k, l = 0, m, n) \to (i, j + 1, k, l = 0, m, n), \quad (2)$$

$$(i, j, k, l = 1, m, n) \to (i + 1, j, k, l = 1, m, n), \quad (3)$$

where $j \neq J - 1$ in the first case, and $i \neq I - 1$ in the second. Equations (1), (2) and (3) define K Chimera graphs. (These edges are drawn in gray color in the figures.) One Chimera layer (fixed k) is depicted in Figure ....

Pegasus first adds connections within each $K_{4,4}$ cell, given by the equation

$$(i, j, k, l, m, n = 0) \to (i, j, k, l, m, n = 1). \quad (4)$$

(These edges are drawn in black color in the figures.) The rest of the new connections in Pegasus come from connecting the unit cells between different layers of Chimera graphs. The qubits of a unit cell (i, j, k) from a specific Chimera layer will be connected to two times three different unit cells on the neighboring Chimera layers, and each pair of unit cells connected in this way will have two times eight different connections in the form of $K_{2,2}$. For the $m = 0$ vertexes of a unit cell, there will be connections to the unit cell with the same (i, j), but in the next layer, these are defined as

$$(i, j, k, l = 0, m = 0, n) \to (i, j, k + 1, l = 1, m', n'), \quad (5)$$

$$(i, j, k, l = 1, m = 0, n) \to (i, j, k + 1, l = 0, m', n'). \quad (6)$$

(These edges are drawn in blue and red colors in the figures.) Note that we have periodicity for the layers: addition for index k is understood modulo K. For the $m = 1$ vertexes, there will be connections to the next unit cells in the next layer, these are defined as

$$(i, j, k, l = 1, m = 1, n) \to (i, j + 1, k + 1, l = 0, m', n'), \quad (7)$$

$$(i, j, k, l = 0, m = 1, n) \to (i + 1, j, k + 1, l = 1, m', n'), \quad (8)$$

where $j \neq J - 1$ in the first case, and $i \neq I - 1$ in the second. (These edges are drawn in orange and green colors in the figures.) The above four equations give four types of edges from each unit cell. One Pegasus unit cell with the edges given above (that is, for fixed (i, j, k)) is depicted in Figure ....,

Applying Eqs. (4), (5), (6), (7) and (8) to obtain additional connections to the Chimera layers given by Eqs. (1), (2) and (3) will produce a Pegasus graph, without producing the same edge twice. A representative part of the Pegasus graph (for $K = 3$) is depicted in Figure ....,

To count the number of edges of a vertex in a generic unit cell (not on the boundary of the layers), we have to reverse the “outgoing” edges given in Eqs. (5), (6), (7) and (8). We call these “incoming” edges,

$$(i, j, k, l = 1, m, n) \leftarrow (i, j, k - 1, l = 0, m' = 0, n'), \quad (9)$$

$$(i, j, k, l = 0, m, n) \leftarrow (i, j, k - 1, l = 1, m' = 0, n'), \quad (10)$$

$$(i, j, k, l = 0, m, n) \leftarrow (i, j - 1, k - 1, l = 1, m' = 1, n'), \quad (11)$$

$$(i, j, k, l = 1, m, n) \leftarrow (i - 1, j, k - 1, l = 0, m' = 1, n'), \quad (12)$$

where $j \neq 0$ in the third case, and $i \neq 0$ in the fourth. From these rules we are now able to count the number of edges incident on a vertex contained in a unit cell which lies not on the boundary of the graph. To start off, each vertex will have 6 edges coming from its chimera graph, Eqs. (1), (2) and (3). The vertex will also have an additional edge from Eq. (4), leading to a total of 7 edges so far. The vertex will additionally have four edges from either Eq. (5) or (6) if $m = 0$, and from either Eq. (7) or (8) if $m = 1$. Moreover, it has additionally another two plus two edges from either (10) and (11), or (9) and (12), leading to a total degree of 15. One Pegasus unit cell with all the edges given above (that is, both outgoing and incoming edges, for fixed (i, j, k)) is depicted in Figure ....,

Figure 2. The Pegasus graph, with open edges to show that the pattern repeats.

fig_PegasusLLfull_allBlue_0_crop.pdf

II. ALTERNATIVE GRAPHICAL REPRESENTATIONS OF PEGASUS

There are many ways that Pegasus can be drawn, so we show some of these in Figures 3 and 4.

III. COMPARISON TO CHIMERA

A. Graph degree

As the yellow vertex in Fig. ? indicates, the vertices of Pegasus (except for vertices on an edge) have a degree 15, which is 2.5 times larger than the maimum degree achieved in Chimera.

B. Non-planarity

We note that certain binary optimization problems forming planar graphs can be solved on a classical computer with a number of operations that scales polynomially with the number of binary variables, with the blossom algorithm [**?** **?**]. Therefore it is important that the qubits of a quantum annealer are connected by a non-planar graph. The $K_{4,4}$ "unit cells" of Chimera are already sufficient to make all commercial D-Wave annealers non-planar. However, if each $K_{4,4}$ unit cell of Chimera physical qubits were to encode one logical qubit (in for example, an extreme case of minor embedding), then Chimera would be planar. While all red, blue, and green lines added in Pegasus are of the form $K_{2,4}$, which itself is planar; these $K_{2,4}$ lines connect unit cells of different planes of chimeras in a non-planar way, such that even if each unit cell were to represent one logical qubit, *these logical qubits would still form a non-planar graph in Pegasus.* This should expand the number of binary optimization problems that can be embedded onto a D-Wave annealer, and cannot be solved on a classical computer with a number of operations that scales polynomially with the number of binary variables.

C. Embedding of quadratization gadgets

We have written an entire paper on the embedding of quadratization gadgets onto Chimera and Pegasus. One highlight of that work is the fact that *all* quadratization gadgets for single cubic terms which require one auxiliary qubit, can be embedded onto Pegasus with no further auxiliary qubits because Pegasus contains K_4,which means that all three logical qubits and the auxiliary qubit can be connected in any way, without any minor-embedding.

Figure 3.

fig_Chimera.pdf

fig_Chimera (1).pdf

fig_Chimera (1).pdf

fig_Pegasus.pdf

fig_Pegasus (1).pdf

fig_Pegasus (2).pdf

Figure 4.

fig_Chimera.pdf

fig_Chimera (1).pdf

fig_Chimera (1).pdf

fig_PegasusLL.pdf

fig_PegasusLL_1.pdf

fig_PegasusLLd.pdf