On The Mathematics of Data Centre Network Topologies

Iain A. Stewart*

School of Engineering and Computing Sciences, Durham University,
Science Labs, South Road, Durham DH1 3LE, U.K.
email: i.a.stewart@durham.ac.uk

Abstract. In a recent paper, combinatorial designs were used to construct switch-centric data centre networks that compare favourably with the ubiquitous (enhanced) fat-tree data centre networks in terms of the number of servers within (given a fixed server-to-server diameter). Unfortunately there were flaws in some of the proofs in that paper. We correct these flaws here and extend the results so as to prove that the core combinatorial construction, namely the 3-step construction, results in data centre networks with optimal path diversity.

1 Introduction

Data centres are expanding both in terms of their physical size and their reach and importance as computational platforms for cloud computing, web search, social networking and so on. There is an increasing demand that data centres incorporate more and more servers but so that computational efficiency is not compromised. A key contributor as to the eventual performance of a data centre is the *data centre network* (*DCN*). New topologies are continually being developed so as to incorporate more servers and best utilize the additional computational power. It is with topological aspects of DCNs that concern us here.

The traditional design of a DCN is *switch-centric* whereby all routing intelligence resides amongst the switches. In such DCNs, there are no direct server-to-server links; only server-to-switch and switch-to-switch links. Switch-centric DCNs are traditionally tree-like with servers located at the 'leaves' of the tree-like structure, e.g., Fat-Tree [1], VL2 [3] and Portland [5]. Whilst it is generally acknowledged that tree-like, switch-centric DCNs have their limitations when it comes to, for example, scalability (with the core switches at the 'roots' quickly becoming bottlenecks), tree-like switch-centric DCNs remain popular and can usually be constructed from commodity hardware. A more recent paradigm, namely *server-centric* DCNs, has emerged so that deficiencies of tree-like, switch-centric DCNs might be ameliorated. In a server-centric DCN, routing intelligence resides within the servers with switches operating only as dumb crossbars; as

* Supported by EPSRC grant EP/K015680/1 'Interconnection Networks: Practice unites with Theory (INPUT)'.

such, there are only server-to-switch and server-to-server links. However, server-centric DCNs also suffer from deficiencies such as packet relay overheads caused by the need to route packets within the server (see [4] for the DCN state of the art). Both switch-centric and server-centric DCNs are abstracted as undirected graphs where the set of nodes is partitioned into a set of servers and a set of switches with edges depending upon the DCN type. It is with switch-centric DCNs that we are concerned here.

It is difficult to design computationally efficient DCNs so as to incorporate large numbers of servers as there are additional design considerations. For example, switches and (especially) servers have a limited number of ports; so, the more servers there are, the greater the average or worst-case link-count between two distinct servers and, consequently, there is a packet latency overhead to be borne. Also, so as to better support routing, fault-tolerance and load-balancing, we would prefer that there is path diversity in the form of numerous alternative (short) paths within the DCN joining any two distinct servers. There are many other design parameters to bear in mind (see, e.g., [7]).

A recent proposal in [6] advocated the use of *combinatorial design theory* in order to design switch-centric DCNs which incorporate more servers, have short server-to-server paths and possess path diversity. The use of combinatorial designs within the study of general interconnection networks is not new and originated in [2] where the targeted networks involved processors communicating via buses. A hypergraph framework was developed in [2] where the hypergraph nodes represented the processors and the hyperedges the buses, and likewise an analogous framework was developed in [6] where the hypergraph nodes represented the servers and the hyperedges represented the switches. However, some of the results derived in [6] are incorrect in that for some of the results there were errors in the proofs while for other results the actual claims are not true.

In this paper we provide correct proofs for some of the results from [6] and we also extend and improve the results from [6]. In particular, using the general construction for building switch-centric DCNs from bipartite graphs and transversal designs as adopted in [6], we prove that in the resulting switch-centric DCNs, there is the maximal number of internally disjoint paths joining any two distinct servers and provide a bound on the length of the longest such path. As can be seen from our proofs, the situation is far more subtle than was assumed in [6].

2 Basic Concepts

Hypergraphs provide the original framework for the 3-step construction as employed in [2] and [6]. A *hypergraph* $H = (V, E)$ consists of a finite set V of *nodes* together with a finite set E of *hyperedges* where each hyperedge is a non-empty set of nodes and each node appears in at least one hyperedge. The *degree* of a node is the number of hyperedges containing it and the *rank* of a hyperedge is its size as a subset of V. A hypergraph is *regular* (resp. *uniform*) if every node has the same degree (resp. every hyperedge has the same rank) with this degree (resp. rank) being the *degree* (resp. *rank*) of the hypergraph. Every graph

$G = (V, E)$ has a natural representation as a hypergraph: the nodes of the hypergraph are V; and the hyperedges are E, where the hyperedge e consists of the pair of nodes incident with the edge e of G.

We can represent a hypergraph $H = (V, E)$ as a bipartite graph: the node set of the bipartite graph is $V \cup E$; and there is an edge (v, D), for $v \in V$ and $D \in E$, in the bipartite graph iff $v \in D$ in the hypergraph. It is clear that this yields a one-to-one correspondence between hypergraphs and bipartite graphs (without isolated nodes) that come complete with a partition of the elements into a 'left-hand side', which will correspond to the nodes of the hypergraph, and a 'right-hand side', which will correspond to the hyperedges of the hypergraph. We assume (henceforth) that every bipartite graph comes equipped with such a partition and for clarity we henceforth refer to the nodes on the left-hand side as *nodes* and the nodes on the right-hand side as *blocks*. Likewise, we refer to the degree of a node as its *degree* and the degree of a block as its *rank*. A bipartite graph corresponding to a regular, uniform hypergraph of degree d and rank Δ is called a (d, Δ)-*bipartite graph*. Every bipartite graph (and so every hypergraph) also describes its *dual hypergraph* where the roles of the nodes on the left-hand side and the blocks on the right-hand side of the partition are reversed in the definition of the hypergraph. With regard to our one-to-one correspondence between bipartite graphs and hypergraphs described above, if G is a bipartite graph then it corresponds to a hypergraph via this correspondence and it also corresponds to a (different) hypergraph via the natural representation highlighted in the previous paragraph.

A *path* in some hypergraph $H = (V, E)$ is an alternating sequence of nodes and hyperedges so that all nodes are distinct, all hyperedges are distinct and a node $v \in V$ follows or preceeds a hyperedge $D \in E$ in the sequence only if $v \in D$ in the hypergraph. The *length* of any path is its length in the bipartite graph corresponding to the hypergraph, and the *distance* between two distinct elements of $V \cup E$ is the length of a shortest path joining these two elements in the corresponding bipartite graph. The *diameter* of H is the maximum of the distances between every pair of distinct nodes of V, and the *line-diameter* of H is the maximum of the distances between every pair of distinct hyperedges of E.

We have two remarks. First, we have analogous notions of diameter and line-diameter in any bipartite graph. Note that our notion of diameter (which ignores node-to-block and block-to-block paths) is different from the usual graph-theoretic notion of diameter in a bipartite graph (and likewise for line-diameter). Second, our graph-theoretic notion of path length in a hypergraph differs from that in [6] where the focus is on the number of hyperedges in a hyperedge-to-hyperedge path in some hypergraph. We shall soon move to an exclusively graph-theoretic formulation in which our notion of length is the natural one.

We shall be interested in building sets of paths in some hypergraph H so that all paths have the same (distinct) source and destination nodes or hyperedges; moreover, we shall require that these paths do not 'interfere' with one another. We say that a set of paths in H joining two distinct elements of $V \cup E$ is: *pairwise internally disjoint* if every node and every hyperedge different from the source

and destination lies on at most one path from this set; or *pairwise internally edge-disjoint* if every pair $(v, D) \in V \times E$ is such that v follows or precedes D on at most one path from this set. The reason we make the above differentiation as regards path disjointness is as follows. Given some hypergraph, our intention is to ultimately consider the nodes as servers and the hyperedges as switches (as it happens, we shall go on to compose such hypergraphs so that servers morph into switches but more later when we discuss composing DCNs). This intention is best appreciated by working with the corresponding bipartite graph where the nodes are to denote servers and the blocks switches. Consequently, we can regard a hypergraph as modelling a switch-centric DCN where there is one layer of switches. A set of pairwise internally disjoint (resp. edge-disjoint) paths is required if we want to enable simultaneous data transfer when the corresponding servers and switches are blocking (resp. non-blocking).

The notion of a transversal design is crucial to what follows.

Definition 1. *Let $k, \Delta \geq 2$. A* $[\Delta, k]$-transversal design T *is a triple (X, D, V) where: $|X| = \Delta k$; $D = (D_1, D_2, \ldots, D_\Delta)$ is a partition of X into Δ equal-sized* groups *(each of size k); and $V = \{V_j : j = 1, 2, \ldots, k^2\}$ is a family of k^2 subsets of X, each of size Δ and called a* block*, so that*

- $|D_i \cap V_j| = 1$*, for $i = 1, 2, \ldots, \Delta$, $j = 1, 2, \ldots, k^2$*
- *each pair of elements $\{x_i, x_j\}$, where $x_i \in D_i$, $x_j \in D_j$ and $i \neq j$, is contained in exactly* 1 *block (we say that the unique block containing x_i and x_j is the block generated by x_i and x_j).*

We adopt a graph-theoretic perspective on transversal designs as defined in Definition 1: we think of the $[\Delta, k]$-transversal design T as a bipartite graph where the elements of X (resp. V) lie on the left-hand side (resp. right-hand side) of the partition, and so are called nodes (resp. blocks) within the bipartite graph, and so that in this bipartite graph there is an edge (p, Q), for $p \in X$ and $Q \in V$, iff in the transversal design the element p is in the block Q. Note that the bipartite graph corresponding to the transversal design from Definition 1 is a (k, Δ)-bipartite graph. Henceforth, we regard both hypergraphs and transversals as bipartite graphs unless we state otherwise.

3 The 3-step Construction and its Extensions

We begin by describing the 3-*step construction* (originating in [2] and used in [6]) for building bipartite graphs by using a base bipartite graph and a transversal design. We'll then explain how one might iterate it and then compose bipartite graphs to obtain more complex DCNs (as was done in [6]).

Step 1: Let H_0 be a (d, Δ)-bipartite graph so that there are n nodes (on the left-hand side of the partition, each of degree d) and e blocks (on the right-hand side, each of rank Δ). Such an H_0 can be visualized as in Fig. 1(a).

Step 2: Let T be a $[\Delta, k]$-transversal design. In particular, there are Δ groups of k nodes (on the left-hand side) as well as k^2 blocks (on the right-hand side). Such

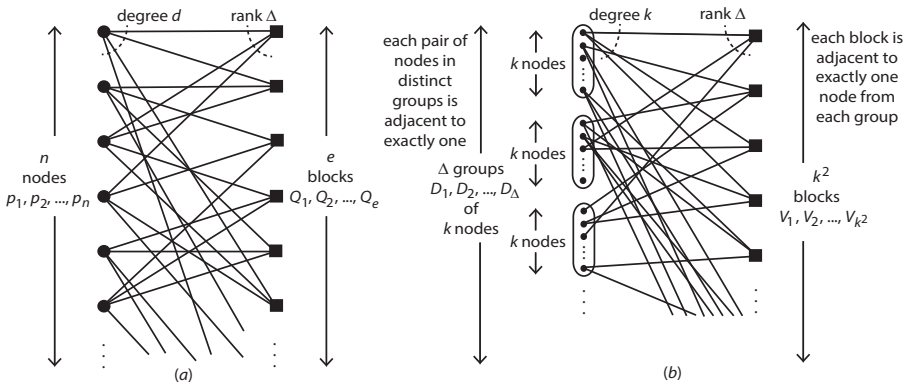

Fig. 1. A (d, Δ)-bipartite graph H_0 and a $[\Delta, k]$-transversal.

a T can be visualized as in Fig. 1(b). Build the bipartite graph H as follows. For every node p of H_0, introduce a group G_p of k nodes of H. For every block Q of H_0, adjacent to the nodes $p_1, p_2, \ldots, p_\Delta$ in H_0, introduce a copy of T, denoted T_Q, rooted on the Δ groups of nodes $G_{p_1}, G_{p_2}, \ldots, G_{p_\Delta}$ (so, corresponding to the block Q of H_0, we have introduced k^2 blocks in H). We refer to the Δ groups of nodes $G_{p_1}, G_{p_2}, \ldots, G_{p_\Delta}$ as the *roots* of the copy T_Q of T in H. Such a bipartite graph H can be visualized as in Fig. 2 where two of the copies of T are partially shown. The bipartite graph H_0 essentially provides a template as to where we introduce copies of T to form H.

Note that: each node of H can be indexed as $a_{p,j}$, where $p \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, k\}$, so that p is the node of H_0 to which the group G_p in which $a_{p,j}$ sits corresponds and j is the index of $a_{p,j}$ in this group; and each block of H can be indexed as $B_{Q,V}$, where $Q \in \{1, 2, \ldots, e\}$ and $V \in \{1, 2, \ldots, k^2\}$, so that Q is the block of H_0 to which the set of blocks in which $B_{Q,V}$ sits corresponds and V is the block of T to which $B_{Q,V}$ corresponds. In addition, each node of T can be indexed $u_{i,j}$, where $i \in \{1, 2, \ldots, \Delta\}$ and $j \in \{1, 2, \ldots, k\}$, so that D_i is the group of nodes in which $u_{i,j}$ sits and j is the index of $u_{i,j}$ in that group.
Step 3: Let H^* be the bipartite graph obtained from the bipartite graph H by reversing the roles of nodes and blocks (so, H^* is the dual bipartite graph of H). Note that the bipartite graph H^* is regular of degree Δ and uniform of rank dk.

We refer to the (dk, Δ)-bipartite graph H (resp. the (Δ, dk)-bipartite graph H^*) constructed above as having been constructed by the 2-step (resp. 3-step) method using the (d, Δ)-bipartite graph H_0 and the $[\Delta, k]$-transversal T.

Our intention with our constructions is to ultimately design switch-centric DCNs with beneficial properties. Whilst there are many properties we would like our DCNs to have, it is important that DCNs can integrate a large number of servers so that the server-to-server distances are short and so that there is redundancy as to which short server-to-server routes we choose to use. In the parlance of bipartite graphs, this translates as building bipartite graphs with a large number of nodes and with redundant, short node-to-node paths. The

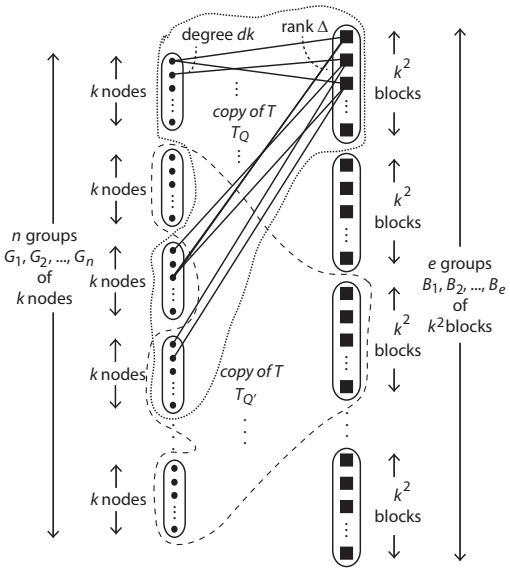

Fig. 2. Amalgamating H_0 and T to get H.

following result was proven in [2] (it is actually derivable from the proofs of our upcoming results) and allows us control over the length of block-to-block paths in 2-step constructions (and so node-to-node paths in 3-step constructions).

Theorem 1 ([2]). *Suppose that the (dk, Δ)-bipartite graph H has been constructed by the 2-step method using the (d, Δ)-bipartite graph H_0 and the $[\Delta, k]$-transversal T. If H_0 has line-diameter $\lambda \geq 2$ then H has line-diameter λ.*

We can iterate the 3-step construction (as was done in [6]). Note that if H_0 is a (d, Δ)-bipartite graph of line-diameter λ then the bipartite graph H_1 resulting from the 2-step construction (using H_0 and some $[\Delta, k]$-transversal design T) is a (dk, Δ)-bipartite graph of line-diameter λ. So, repeating the 2-step construction but with H_1 replacing H_0 (we keep the same T, though) yields a (dk^2, Δ)-bipartite graph H_2 of line-diameter λ. By iterating this construction, we can clearly obtain a (dk^i, Δ)-bipartite graph H_i of line-diameter λ. Converting H_i into H_i^* results in a bipartite graph with ek^{2i} nodes, with dk^i blocks, with diameter λ and that is regular of degree Δ and uniform of rank dk^i.

In [2], the 3-step construction was the focus as the application there was to build bus interconnection networks of large size but so as to limit the diameter of the resulting network. Similarly, in [6], the 3-step construction was the focus as the intention was to interpret nodes as servers and blocks as switches; were we to focus on the 2-step method and allow the server degree to grow (in H_i, above, the degree is dk^i), this would result in practically infeasible DCNs.

New methods of composing bipartite graphs (built using the 3-step construction) so as to obtain switch-centric DCNs were also derived in [6] where 4 such

methods were given: Methods M_1, M_2 and M_3 are different cases of our Method A; and Method M_4 is our Method B. Let H be a (σ, ω)-bipartite graph which we think of as a DCN with the nodes as servers and the blocks as switches, and where $\sigma < \omega$.
Method A: We take c copies of H where $\omega - c\sigma > 0$ and $c \geq 1$. For each server (node) u of H: we remove the corresponding server in each of the c copies of H and introduce a new switch (this switch is common to all copies of H); we make all of the $c\sigma$ links incident with the c original servers incident with this new switch; and we attach $\omega - c\sigma$ pendant servers to the new switch (in [6], the new switches are termed *level*-1 *switches* with the original switches *level*-2 *switches*). So, the new DCN is such that: all switches have ω ports; there are links from (new) servers to level-1 switches; and links joining level-1 and level-2 switches. Note that there is some choice as regards the parameter c. The case where $c \geq 1$ corresponds to Method M_2 of [6]; the case when $c = 1$ corresponds to Method M_1; and the case when $c = \lfloor \frac{\lfloor \frac{\omega}{2} \rfloor}{\sigma} \rfloor$ corresponds to Method M_3 (here, the aim is to ensure that every level-1 switch is adjacent to roughly the same number of level-2 switches as it is nodes).
Method B: We now work with a switch-centric DCN as constructed by Method A. Let every level-1 switch have n_e adjacent servers. Suppose that there is an even number of level-1 switches. Partition the set of level-1 switches into pairs. For each pair of switches (S', S''): remove $\lfloor \frac{n_e}{2} \rfloor$ servers that are adjacent to S' and remove $\lceil \frac{n_e}{2} \rceil$ servers that are adjacent to S''; and make every server that is adjacent to the switch S' or the switch S'' also adjacent to the other switch.

In [6], various switch-centric DCNs were constructed using the 3-step construction allied with Methods A and B and were favourably compared with the ubiquitous 3-level fat-tree with regard to the number of servers therein when the diameter and the switch radix are held constant (see Tables 2–4 in [6]).

4 Constructions of Paths

We are now in a position to use transversal designs to build switch-centric DCNs, similarly to as was done in [6]. However, in [6] there were a number of problems with the proofs (so much so that some claimed results are false). We begin by highlighting these problems and then we provide not only correct proofs but also extend some of the claimed results in [6] with regard to path diversity.

In order to detail the difficulties in [6], we adopt the terminology of [6]. In Subcases (1.1) and (2.1) of the proof of Theorem 2 in [6], the situation when $r_j = s_j$, for some j where $p \neq t_j$, was not considered (although this is trivial to remedy). However, and more importantly, in Subcases (1.2) and (2.2) the construction does not work when $j = i$ as r_i, s_i, t_i all lie in the same group G_i^E and consequently we cannot infer the existence of R_i and S_i.

An attempt was also made in [6] to extend Theorem 2 of [6]; see Theorem 3 of [6]. Assumptions concerning the connectivity of H_0 are made and the existence of additional paths to those constructed in the proof of Theorem 2 are claimed in the situation when the two blocks $B_{Q,V}$ and $B_{Q',V'}$ are such that $Q \neq Q'$.

However, there are serious flaws in the proof of Theorem 3 of [6], so much so that the theorem is untrue. In short, Theorem 3 of [6] claims that if there are ω pairwise internally disjoint paths in H_0 from Q to Q' then there are $\min\{\Delta\omega, k\omega\}$ pairwise internally disjoint paths in H from $B_{Q,V}$ to $B_{Q',V'}$. This does not make sense: the maximum number of pairwise internally disjoint paths in H from $B_{Q,V}$ to $B_{Q',V'}$ is Δ (as the bipartite graph H has rank Δ) and so we must have that $\min\{\Delta\omega, k\omega\} \leq \Delta$. For instance, in Example 1 of [6], where the bipartite graph H_0 is the cycle of length 10, so that $d = \Delta = 2$ and $n = e = 5$, and a $[2,3]$-transversal T is used, the bipartite graph H built by the 2-step method has degree 6 and rank 2. However, there are 2 paths from any block of H_0 to any other block of H_0 and so if Theorem 3 of [6] were true then there would be 4 pairwise disjoint paths from $B_{Q,V}$ to $B_{Q',V'}$ in H which clearly cannot be the case.

We now resurrect (some of) the proofs from [6] and extend the results claimed in that paper. We use the following easy-to-prove lemma repeatedly.

Lemma 1. *Let T be some $[\Delta, k]$-transversal with groups of nodes $\{D_1, D_2, \ldots, D_\Delta\}$. Let U be some block of T. For each $i \in \{1, 2, \ldots, \Delta\}$, let $r_i \in D_i$ be the unique node of D_i that is adjacent to U. Set $R = \{r_i : i = 1, 2, \ldots, \Delta\}$. Let P be a set of distinct pairs of nodes so that: exactly one node of any pair in P is in R and no node of R is in more than one pair of P; and no pair in P is such that both nodes lie in the same group. The blocks generated by the pairs in P are all distinct and different from U.*

Theorem 2. *Let $k, \Delta, d \geq 2$ but where $(k, \Delta) \notin \{(2,3), (2,5), (2,7)\}$. Let H be built by the 2-step method from the (d, Δ)-bipartite graph H_0 using the $[\Delta, k]$-transversal T.*

(a) If Q and Q' are distinct blocks of H_0 so that there are $\lambda \geq 1$ pairwise internally disjoint paths in H_0 from Q to Q', each of length at most μ, then there are $\min\{\Delta, k\lambda\}$ pairwise internally disjoint paths from any block $B_{Q,V}$ of H to any other block $B_{Q',V'}$ of H, each of length at most $\mu + 4$.

(b) If $B_{Q,V}$ and $B_{Q,V'}$ are distinct blocks of H then there are Δ pairwise internally disjoint paths from $B_{Q,V}$ to $B_{Q,V'}$, each of length at most 6 and lying entirely within T_Q.

Proof. (a) We may assume that $\lambda \leq \lceil \frac{\Delta}{k} \rceil$. Consider the λ pairwise internally disjoint paths from Q to Q' in H_0. We may clearly assume that either every path has length 2 or that every common neighbour of Q and Q' in H_0 lies on one of the λ paths (with each of these paths having length 2).

Suppose that $b + c = \lambda$, where $b \geq 1$ and $c \geq 0$, and that the nodes $p_1, p_2, \ldots, p_b$ are common neighbours in H_0 of Q and Q' (the case when there are no common neighbours is easy). As stated above, we may assume that either: $b = \lambda$; or $c > 0$ and $\{p_1, p_2, \ldots, p_b\}$ consists of all common neighbours of Q and Q' in H_0. In the case when $c > 0$, let the nodes $q_1, q_2, \ldots, q_c$ be neighbours of Q but not of Q' in H_0, and let the nodes $q'_1, q'_2, \ldots, q'_c$ be neighbours of Q' but not of Q in H_0 so that the remaining c paths from Q to Q' in H_0 are of the form $Q, q_i, \ldots, q'_i, Q'$, for $i = 1, 2, \ldots, c$.

We begin with an involved construction. Set $k' = \Delta - k(\lceil \frac{\Delta}{k} \rceil - 1)$; so $1 \leq k' \leq k$. We can batch groups of nodes of T_Q and $T_{Q'}$ in H as follows:

- for $i \in \{1, 2, \ldots, b\}$, define $G_0^i = G_{p_i} = H_0^i$
- for $i \in \{1, 2, \ldots, c\}$ (where $c > 0$), define $G_0^{b+i} = G_{q_i}$ and $H_0^{b+i} = G_{q'_i}$
- for $i \in \{1, 2, \ldots, b + c - 1\}$, choose groups $G_1^i, G_2^i, \ldots, G_{k-1}^i$ within T_Q and groups $H_1^i, H_2^i, \ldots, H_{k-1}^i$ within $T_{Q'}$, and choose groups $G_1^{b+c}, G_2^{b+c}, \ldots, G_{k'-1}^{b+c}$ within T_Q and groups $H_1^{b+c}, H_2^{b+c}, \ldots, H_{k'-1}^{b+c}$ within $T_{Q'}$ so that:
  - all G_j^i, where $j > 0$, are distinct and different from $G_0^1, G_0^2, \ldots, G_0^{b+c}$
  - all H_j^i, where $j > 0$, are distinct and different from $H_0^1, H_0^2, \ldots, H_0^{b+c}$
  - any G_j^i, where $j > 0$, corresponds to some node p of H_0 that is adjacent to both Q and Q' iff the group H_j^i corresponds to the same node p of H_0, i.e, G_j^i and H_j^i are identical.

We have three remarks: each G_j^i, where $j \geq 0$, is in T_Q and each H_j^i, where $j \geq 0$, is in $T_{Q'}$, so that if $c > 0$ then the only groups common to both T_Q and $T_{Q'}$ are $G_{p_1}, G_{p_2}, \ldots, G_{p_b}$; the bound $b + c \leq \lceil \frac{\Delta}{k} \rceil$ means that there are enough groups available in both T_Q and $T_{Q'}$ for us to be able to choose as above; and if some group of the form G_j^i, where $j > 0$, is identical to the group H_j^i then it must be the case that both are rooted at the same node p of H_0 that is a common neighbour of Q and Q' in H_0, and consequently that $c = 0$.

For each $i \in \{1, 2, \ldots, b + c\}$ and each $j \in \{0, 1, \ldots, k - 1\}$, if $i \neq b + c$, or each $j \in \{0, 1, \ldots, k' - 1\}$, if $i = b + c$, let $r_j^i \in G_j^i$ (resp. $s_j^i \in H_j^i$) be the unique node of G_j^i (resp. H_j^i) that is adjacent to $B_{Q,V}$ (resp. $B_{Q',V'}$) in H. Note that the pair r_j^i and s_j^i lie in the same group of H iff both G_j^i and H_j^i are rooted at the same node of H_0 and this node is adjacent to both Q and Q' in H_0.

For each $i \in \{1, 2, \ldots, b+c\}$, let $G_0^i = \{r_0^i, t_1^i, \ldots, t_{k-1}^i\}$ and $H_0^i = \{s_0^i, w_1^i, \ldots, w_{k-1}^i\}$ so that in the case when $G_0^i = H_0^i$: if $r_0^i = s_0^i$ then $t_j^i = w_j^i$, for $j \in \{1, 2, \ldots, k - 1\}$; and if $r_0^i \neq s_0^i$ then $r_0^i = w_1^i$ and $s_0^i = t_1^i$, with $t_j^i = w_j^i$, for $j \in \{2, 3, \ldots, k - 1\}$.

We are now ready to generate some blocks within T_Q and $T_{Q'}$ in H. For each $i \in \{1, 2, \ldots, b + c\}$ and each $j \in \{1, 2, \ldots, k - 1\}$, if $i \neq b + c$, or each $j \in \{1, 2, \ldots, k' - 1\}$, if $i = b + c$: let $B_{r_j^i, t_j^i}$ be the block of T_Q in H generated by $r_j^i \in G_j^i$ and $t_j^i \in G_0^i$; and let $B'_{s_j^i, w_j^i}$ be the block of $T_{Q'}$ in H generated by $s_j^i \in H_j^i$ and $w_j^i \in H_0^i$. So, we have generated $\Delta - \lambda$ blocks in T_Q and $\Delta - \lambda$ blocks in $T_{Q'}$. Note that any block of T_Q is necessarily distinct from any block of $T_{Q'}$. By Lemma 1 applied twice to both T_Q and $T_{Q'}$: all blocks of $\{B_{r_j^i, t_j^i} : i = 1, 2, \ldots, b+c-1$ and $j = 1, 2, \ldots, k-1$, or $i = b+c$ and $j = 1, 2, \ldots, k' - 1\}$ are distinct and different from $B_{Q,V}$; and all blocks of $\{B'_{s_j^i, w_j^i} : i = 1, 2, \ldots, b+c-1$ and $j = 1, 2, \ldots, k - 1$ or $i = b + c$ and $j = 1, 2, \ldots, k' - 1\}$ are distinct and different from $B_{Q',V'}$; call these two sets of blocks our working sets of blocks.

Now we build some paths from $B_{Q,V}$ to $B_{Q',V'}$ in H. For each $i \in \{1, 2, \ldots, b\}$: if $r_0^i = s_0^i$ then define the paths:

- π_0^i as $B_{Q,V}, r_0^i, B_{Q',V'}$

– π_1^i as $B_{Q,V}, r_1^i, B_{Q',V'}$, if $r_1^i = s_1^i$, and as $B_{Q,V}, r_1^i, B_{r_1^i,t_1^i}, t_1^i, B'_{s_1^i,w_1^i}, s_1^i$, $B_{Q',V'}$, if $r_1^i \neq s_1^i$ (note that $t_1^i = w_1^i$)

and if $r_0^i \neq s_0^i$ then define the paths:

– π_0^i as $B_{Q,V}, r_0^i, B'_{s_1^i,w_1^i}, s_1^i, B_{Q',V'}$ (note that $w_1^i = r_0^i$)
– π_1^i as $B_{Q,V}, r_1^i, B_{r_1^i,t_1^i}, s_0^i, B_{Q',V'}$ (note that $t_1^i = s_0^i$).

The above definition of π_0^i and π_1^i presupposes that both paths exist; that is, that it is not the case that $\Delta = k(b-1)+1$ and $r_0^b \neq s_0^b$ (as otherwise it is not clear how we build only π_0^b without having recourse to G_1^b; note that if $\Delta = k(b-1)+1$ and $r_0^b = s_0^b$ then π_0^b exists). We shall return to this special case later.

For each $i \in \{1, 2, \ldots, b\}$ and each $j \in \{2, 3, \ldots, k-1\}$, if $i < b+c$, or each $j \in \{2, 3, \ldots, k'-1\}$, if $i = b$ and $c = 0$: if $r_j^i \neq s_j^i$ then define the path π_j^i as $B_{Q,V}, r_j^i, B_{r_j^i,t_j^i}, t_j^i, B'_{s_j^i,w_j^i}, s_j^i, B_{Q',V'}$; and if $r_j^i = s_j^i$ then define the path π_j^i as $B_{Q,V}, r_j^i, B_{Q',V'}$.

Note that out of all the 'π-paths' constructed above, the only way that we can have that two of our paths are not internally disjoint is when $r_0^i \neq s_0^i$ but $r_1^i = s_1^i$, for some $i \in \{1, 2, \ldots, b\}$. In every such case, choose $x_1^i \in G_1^i \setminus \{r_1^i\}$. Let $B_{r_0^i,x_1^i}$ be the block of T_Q in H generated by $r_0^i \in G_0^i$ and $x_1^i \in G_1^i$, and let $B'_{s_0^i,x_1^i}$ be the block of $T_{Q'}$ in H generated by $s_0^i \in G_0^i$ and $x_1^i \in G_1^i$ (in essence, we have dispensed with the blocks $B_{r_1^i,t_1^i}$ and $B'_{s_1^i,w_1^i}$ and replaced them with the blocks $B_{r_0^i,x_1^i}$ and $B'_{s_0^i,x_1^i}$ in our working sets of blocks; we reiterate that we do this for every $i \in \{1, 2, \ldots, b\}$ for which $r_0^i \neq s_0^i$ and $r_1^i = s_1^i$). The conditions of Lemma 1 still hold and so the blocks in our working sets of blocks are all distinct and different from $B_{Q,V}$ and $B_{Q',V'}$. For each $i \in \{1, 2, \ldots, b\}$ for which $r_0^i \neq s_0^i$ and $r_1^i = s_1^i$, redefine the paths: π_0^i as $B_{Q,V}, r_1^i, B_{Q',V'}$; and π_1^i as $B_{Q,V}, r_0^i, B_{r_0^i,x_1^i}, x_1^i, B'_{s_0^i,x_1^i}, s_0^i, B_{Q',V'}$. The paths from the resulting set of π-paths are now pairwise internally disjoint.

Let us now return to the situation where $\Delta = k(b-1)+1$ and $r_0^b \neq s_0^b$ (so, necessarily, $c = 0$). In this case, we proceed exactly as we did above but without building the path π_0^b. We need to build a path of the form $B_{Q,V}, r_0^b, \ldots, s_0^b, B_{Q',V'}$ (that is internally disjoint from all of the above $\Delta - 1$ π-paths). Suppose that $k \geq 3$; so, there is a node $x_0^{b-1} \in G_0^{b-1} \setminus \{r_0^{b-1}, s_0^{b-1}\}$. Generate the block $B_{r_0^b,x_0^{b-1}}$ of T_Q within H and the block $B_{s_0^b,x_0^{b-1}}$ of $T_{Q'}$ within H. By Lemma 1, these blocks are different from $B_{Q,V}$, $B_{Q',V'}$ and all other blocks so generated within T_Q and $T_{Q'}$. Define the path π_0^b as $B_{Q,V}, r_0^b, B_{r_0^b,x_0^{b-1}}, x_0^{b-1}, B_{s_0^b,x_0^{b-1}}, s_0^b, B_{Q',V'}$. This path is internally disjoint from all other π-paths. Suppose that $k = 2$. So, there are 4 blocks in T and consequently $\Delta \in \{3, 5, 7\}$ which yields a contradiction.

If $c > 0$ then we can define additional paths in H from $B_{Q,V}$ to nodes of $G_0^{b+1}, G_0^{b+2}, \ldots, G_0^{b+c}$ and from $B_{Q',V'}$ to nodes of $H_0^{b+1}, H_0^{b+2}, \ldots, H_0^{b+c}$ (note that in this scenario $G_j^i \neq H_j^i$ unless $i \in \{1, 2, \ldots, b\}$ and $j = 0$). For each $i \in \{b+1, b+2, \ldots, b+c\}$, define the paths: η_0^i as $B_{Q,V}, r_0^i$; and ν_0^i as $B_{Q',V'}, s_0^i$. For each $i \in \{b+1, b+2, \ldots, b+c\}$ and each $j \in \{1, 2, \ldots, k-1\}$, if $i \neq b+c$, or

each $j \in \{1, 2, \ldots, k'-1\}$, if $i = b+c$, define the paths: η^i_j as $B_{Q,V}, r^i_j, B_{r^i_j,t^i_j}, t^i_j$; and ν^i_j as $B_{Q',V'}, s^i_j, B'_{s^i_j,w^i_j}, w^i_j$. Any 2 distinct paths from our collection of π-paths, η-paths and ν-paths clearly have no nodes in common and the only block in common is $B_{Q,V}$, $B_{Q',V'}$ or both.

If we can find a path in H from r^i_0 or t^i_j to s^i_0 or w^i_j, respectively, for each $i \in \{b+1, b_2, \ldots, b+c\}$ and each $j \in \{1, 2, \ldots, k-1\}$, if $i < b+c$, or each $j \in \{1, 2, \ldots, k'-1\}$, if $i = b+c$, so that no node or block of any of these paths, apart from the source and destination nodes, lies in T_Q or $T_{Q'}$ and so that the resulting paths are pairwise internally disjoint then we are done. Fix $i \in \{1, 2, \ldots, c\}$ and let $Q, q_i, Q_1, q^2_i, Q_2, q^3_i, \ldots, q^m_i, Q_m, q'_i, Q'$, be one of our remaining c paths from Q to Q' in H_0; in particular, $m \in \{1, 2, \ldots, \frac{1}{2}(\mu-2)\}$. In H: there are k paths of length 2, each path having a source in G_{q_i} and a destination in $G_{q^2_i}$ so that all sources are distinct as are all destinations and lying entirely within T_{Q_1}; there are k paths of length 2, each path having a source in $G_{q^2_i}$ and a destination in $G_{q^3_i}$ so that all sources are distinct as are all destinations and lying entirely within T_{Q_2}; ...; and there are k paths of length 2, each path having a source in $G_{q^m_i}$ and a destination in $G_{q'_i}$ so that all sources are distinct as are all destinations and lying entirely within T_{Q_m}. We are done.

Now return to the case ignored at the beginning of the proof, namely the case when $b = 0$ and $c = \lambda$. The above construction of paths from $B_{Q,V}$ to each node of G_{q_i}, concatenated with paths from each node of G_{q_i} to each node of $G_{q'_i}$, concatenated with paths from each node of $G_{q'_i}$ to $B_{Q',V'}$ still works.

(b) Consider the case when our two blocks are $B_{Q,V}$ and $B_{Q,V'}$. Suppose that the block Q of H_0 is adjacent to the nodes $p_1, p_2, \ldots, p_\Delta$. For each $i \in \{1, 2, \ldots, \Delta\}$, let $r_i \in G_{p_i}$ be adjacent to $B_{Q,V}$ in H and let $s_i \in G_{p_i}$ be adjacent to $B_{Q,V'}$ in H. W.l.o.g. suppose that $r_i \neq s_i$, for $i = 1, 2, \ldots, b$, and that $r_i = s_i$, for $i = b+1, b+2, \ldots, \Delta$.

Suppose that $b \geq 2$. For each $i \in \{1, 2, \ldots, b-1\}$, let $B_{r_i,s_{i+1}}$ be the unique block of T_Q that is generated by r_i and s_{i+1}, and let B_{r_b,s_1} be the unique block of T_Q that is generated by r_b and s_1. By Lemma 1, all blocks $B_{r_1,s_2}, B_{r_2,s_3}, \ldots, B_{r_{b-1},s_b}, B_{r_b,s_1}$ are distinct and different from $B_{Q,V}$ and $B_{Q,V'}$. Hence, if π_i is the path $B_{Q,V}, r_i, B_{r_i,s_{i+1}}, s_{i+1}, B_{Q,V'}$, for $i \in \{1, 2, \ldots, b-1\}$, π_b is the path $B_{Q,V}, r_b, B_{r_b,s_1}, s_1, B_{Q,V'}$, and π_i is the path $B_{Q,V}, r_i, B_{Q,V'}$, for $i \in \{b+1, b+2, \ldots, \Delta\}$, then the set of paths are pairwise internally disjoint.

If $b = 0$ then the above construction trivially yields Δ paths of length 2 from $B_{Q,V}$ to $B_{Q,V'}$. Suppose that $b = 1$. Choose $x_2 \in G_{p_2} \setminus \{r_2\}$ and let B_{r_1,x_2} (resp. B_{s_1,x_2}) be the block of T_Q generated by r_1 and x_2 (resp. s_1 and x_2). By Lemma 1, B_{r_1,x_2}, B_{s_1,x_2}, $B_{Q,V}$ and $B_{Q,V'}$ are all distinct. So, if π_1 is the path $B_{Q,V}, r_1, B_{r_1,x_2}, x_2, B_{s_1,x_2}, s_1, B_{Q,V'}$ and π_i is the path $B_{Q,V}, r_i, B_{Q,V'}$, for $i \in \{2, 3, \ldots, \Delta\}$ then we obtain a pairwise internally disjoint set of paths. □

Note that Theorem 2 is optimal in the sense that if H_0 has blocks Q and Q' so that there are exactly λ pairwise internally disjoint paths from Q to Q' in H_0 then we can do no better than $\min\{\Delta, k\lambda\}$ pairwise internally disjoint paths from any block $B_{Q,V}$ to any block $B_{Q',V'}$ in H, as by Menger's Theorem we can remove λ nodes from H_0 so as to disconnect Q and Q', and so $k\lambda$ nodes

from H so as to disconnect $B_{Q,V}$ and $B_{Q',V'}$. Note also that irrespective of the erroneous proofs in [6], Theorem 2 extends any claimed results in [6] by deriving $\min\{\Delta, k\lambda\}$ pairwise internally disjoint paths from any block $B_{Q,V}$ in H to any block $B_{Q',V'}$ where not only might we have $Q \neq Q'$ but also $Q = Q'$.

5 Conclusion

In this paper we have extended the use of mathematical techniques within the design of data centre networks. We feel that theoretical computer science has a lot to offer more practical areas such as data centre design and hope that this work provides some impetus to theoreticians. Naturally, our work provokes some directions for further research, both theoretical and applied. Whilst we have developed an optimal analysis of path diversity as regards using the 3-step construction, we have yet to use the additional path diversity so obtained. In order to do this we would need use bipartite graphs H_0 (with reference to the 3-step construction) with additional connectivity properties. In future we shall seek to build and use such bipartite graphs. Also, our constructions form part of a wider as yet untouched topic, analogous to the well-established study of Moore graphs, namely the analysis of (not graphs but) 'switch-server graphs' (the models of DCNs) as to the maximal number of servers that can be incorporated under 'degree and diameter' constraints. Finally, we would like to use techniques similar to those here so as to build not just switch-centric DCNs but also server-centric DCNs.

References

1. M. Al-Fares, A. Loukissas and A. Vahdat, A scalable, commodity data center network architecture, *SIGCOMM Comput. Commun. Rev.*, 38(4), 63–74 (2008)
2. J.C. Bermond, J. Bond and S. Djelloul, Dense bus networks of diameter 2, *Proc. Workshop on Interconnection Networks*, DIMACS Series, 21, 9–18 (1995)
3. A. Greenberg, J.R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D.A. Maltz, P. Patel and S. Sengupta, VL2: a scalable and flexible data center network, *SIGCOMM Comput. Commun. Rev.*, 39(4), 51–62 (2009)
4. Y. Liu, J.K. Muppala, M. Veeraraghavan, D. Lin and J. Katz, *Data Centre Networks: Topologies, Architectures and Fault-Tolerance Characteristics*, Springer (2013)
5. R.N. Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya and A. Vahdat, Portland: a scalable fault-tolerant layer 2 data center network fabric, *SIGCOMM Comput. Commun. Rev.*, 39 (4), 39–50 (2009)
6. G. Qu, Z. Fang, J. Zhang and S.-Q. Zheng, Switch-centric data center network structures based on hypergraphs and combinatorial block designs, *IEEE Trans. on Par. Distrib. Sys.*, 26 (4), 154–1164 (2015)
7. K. Wu, J. Xiao and L.M. Ni, Rethinking the architecture design of data center networks, *Frontiers of Comput. Sci.*, 6 (5), 596–603 (2012)