# Interconnection Networks Assignment

**SLAT hours and load**: According to the Faculty Handbook, this sub-module has a load of 50 SLAT hours. Of these, 10 hours cover attending the 10 lectures. Consequently (as this module is coursework only), the remaining 40 hours cover the time taken to understand and digest the lecture material and to complete this assignment; this means devoting, on average, 4 hours per week to this sub-module during the first term (outside of the lectures).

**Helping your time management**: Questions are presented in roughly lecture-order and are released week by week in 5 batches so that you can work on the coursework throughout the whole of the term. However, when you answer any question, you should apply your *full course knowledge* (so, you might return to an early answer later in the term after you have learned more). Don't forget to revisit the videos of lectures as sometimes I might make a helpful comment verbally that does not appear in writing.

**Format**: The main document containing your written answers should be **in pdf form** and called `Int_Nets_text_abcd12.pdf` where 'abcd12' is your personal ID (I don't care whether the document is typeset or simply photos of legible handwritten pages but **it needs to be in pdf form**). You should also submit your implementations of routing algorithms, all defined as functions (in the form and named as explained below), in a single Python program entitled `Int_Nets_routing_abcd12.py` so that the functions are executable in Python 3.8. Moreover, there should be no non-standard imported modules. Both the answers to the questions and your Python code should be submitted in a zipped folder named `abcd12`.

***If you do not follow these instructions then you run the risk of losing marks!***

## Questions *roughly* covered by Lectures 1 and 2

**Question 1.** The graph on Slide 14 of Lecture 1 in the bottom right-hand corner (where there is an orange background) is actually called a *biswapped network* and is designed for use in machines with both electrical and optical connections. The electrical connections are the 'short' ones joining sets of nodes in cycles (5 such cycles are shown above and 5 below the red dotted line but in general there are more that are hidden) and the optical connections are the 'long' ones that cross the red dotted line.

Let us suppose that our biswapped network $B_n$ consists of $n$ cycles above the red dotted line and $n$ below and that each cycle has length $n$ ($B_5$ is shown on Slide 14). The intention in a biswapped network is that for any cycle, there is an edge joining a node of that cycle to a node of each of the other cycles that lie on the other side of the red dotted line (so, the optical connections form a matching in the graph).

($a$) Give a *precise algebraic description* of such a biswapped network $B_n$. [4 marks]

[Hint: the node-set of $B_n$ should be defined as $\{(i, j, k) : i, j = 0, 1, \ldots, n-1; k = 0, 1\}$. You need to describe the edges.]

($b$) Does $B_n$ have a Hamiltonian cycle? [6 marks]

[You should justify your answer via a formal argument.]

**total 10 marks**

**Question 2.** The graph on Slide 10 of Lecture 2 in the top right-hand corner is actually called a *star graph* $S_4$.

($a$) Give a *precise algebraic description* of $S_4$. [3 marks]

[You should phrase your definition so that the digits in the node names are $0, 1, 2, 3$ rather than $1, 2, 3, 4$ (it is the latter representation in the figure on Slide 10).]

($b$) Provide a general definition of an analogous star-graph $S_n$ where $n > 4$. [2 marks]

[Again, you should phrase your definition so that the digits in the node names are $0, 1, \ldots, n-1$.]

($c$) Comment on potential problems with the scalability of the family of interconnection networks $\{S_n\}$; in particular, on the construction of $S_{n+1}$ from $S_n$. [4 marks]

($d$) The $(n,k)$-*star* $S_{n,k}$, for $k \leq n$, has node-set $\{(p_1, p_2, \ldots, p_k) :$ each $p_i \in \{0, 1, \ldots, n-1\}; p_i \neq p_j$, for all $i \neq j\}$ and any node $(p_1, p_2, \ldots, p_k)$ has neighbours

- $(p_i, p_2, \ldots, p_{i-1}, p_1, p_{i+1}, \ldots, p_k)$, for all $2 \leq i \leq k$
- $(q, p_2, \ldots, p_k)$, for all $q \in \{0, 1, \ldots, n-1\} \setminus \{p_1, p_2, \ldots, p_k\}$.

Comment on the scalability of the family of interconnection networks $\{S_{n,k}\}$ in comparison with the family $\{S_n\}$. [2 marks]

[There is no need to supply full quantitative data.]

($e$) Devise a routing algorithm that will enable you to compute a route from the node $\mathbf{x} = (x_1, x_2, \ldots, x_k)$ of the $(n,k)$-star $S_{n,k}$ to the node $\mathbf{e} = (0, 1, \ldots, k-1)$. [9 marks]

[You need only describe your algorithm using pseudocode or natural language; though in both cases you should ensure that your account is clear, readable and unambiguous.]

**total 20 marks**

## Interconnection Networks Assignment

**Questions *roughly* covered by Lectures 3 and 4**

**Question 3.** Are the highlighted sets of links in Fig. 1 cuts? (Justify your answer.)

**total 5 marks**

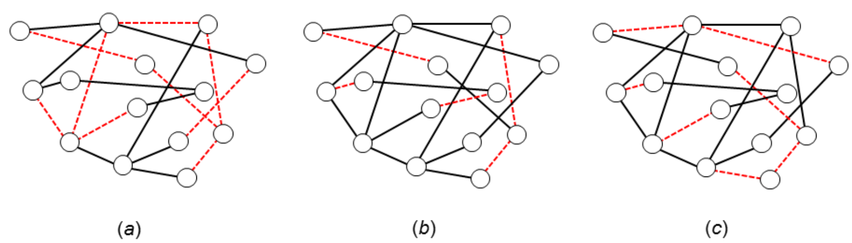


Figure 1: Some potential cuts shown in dotted red.

**Question 4.** Suppose that we have an interconnection network $N$ on $n$ nodes, where $n$ is even, and a function $\varphi$ that takes the nodes of the complete digraph $D_n$ on $n$ nodes to the nodes of $N$. Suppose further that $\varphi$ is one-to-one and for every directed edge $(u, v)$ of $D_n$, there is a directed path $\rho_{u,v}$ in $N$ from $\varphi(u)$ to $\varphi(v)$ ($\varphi$ is an *embedding* of $D_n$ into $N$ of *load* 1; see Slide 2 in Lecture 4). Suppose that no directed edge of $N$ is such that it appears on more than $\gamma$ $\rho$-paths (the embedding $\varphi$ has *congestion* at most $\gamma$). If the bisection width of $N$ is denoted by $\beta$, prove that $\beta\gamma \geq \frac{n^2}{2}$.

**total 10 marks**

**Question 5.** Consider the Petersen Graph $P$ in Fig. 2. By finding an embedding of the complete digraph $D_{10}$ into $P$ with load 1 and congestion 5, prove that the bisection width of $P$ is exactly 10.

**total 15 marks**

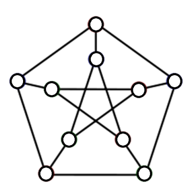

Figure 2: Petersen Graph.

## Interconnection Networks Assignment

**<u>Questions *roughly* covered by Lectures 5 and 6</u>**

**Question 6.** Prove that the $3 \times 6$ mesh has bisection width 6.

**total 5 marks**

**Question 7.** Suppose that we have 900 processors, each with 32 pins and so that the signal frequency is 1 GHz. Our intention is to build a distributed-memory multiprocessor whose interconnection network is a $k$-ary $n$-cube $Q_n^k$, with $n \geq 3$ and $k \geq 4$, or a cube-connected cycles $CCC_n$ but so that for each interconnection network, we use as many of our processors as possible.

($a$) Analyse and compare each of the possible candidate designs with regard to the maximum possible ideal throughput $\theta_{ideal}$. Which design is preferable: a $Q_n^k$ or a $CCC_n$? [7 marks]

[You should take $H_{ave}$ as $n\lfloor \frac{k^2}{4} \rfloor / k$ for $Q_n^k$ and as $\frac{7n}{4}$ for $CCC_n$.]

($b$) Give three factors that are ignored when the upper bounds on $\theta_{ideal}$ are derived. [3 marks]

**total 10 marks**

**Question 8.** Prove that the $(4, 2)$-star $S_{4,2}$ is node-symmetric.

**total 15 marks**

# Interconnection Networks Assignment

**Questions *roughly* covered by Lectures 7 and 8**

**Question 9.** Consider the *circulant graph* $C_{n,3}$ where there are $n = 6m + 4$ vertices $\{0, 1, \ldots, n-1\}$, with $m \geq 1$, and where there are edges $\{(x, y) : x = y \pm 1 \text{ or } x = y \pm 3\}$, with addition mod $n$. The graph $C_{16,3}$ can be visualized as in Fig. 1 (here, $m = 2$).

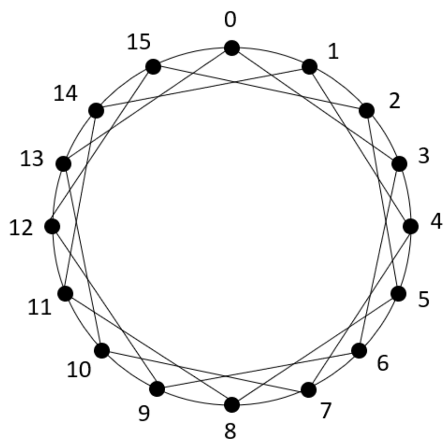


Figure 1: The circulant $C_{16,3}$.

$(a)$ Calculate $H_{ave}$ for $C_{n,3}$. [7 marks]

[You need only sketch a proof.]

$(b)$ Compare $C_{124,3}$ and the cycle (or ring) $C_{124}$ of length 124 in terms of average zero-load latency when every router has delay $t_r = 20$ ns, every channel has bandwidth 5 GHz and every packet has length $L = 1024$ bits. [3 marks]

[You can ignore the time of flight.]

**total 10 marks**

**Question 10.** Let $S_{n,k}$ be the $(n, k)$-star as defined in Question 2 where $3 \leq k \leq n - 1$. Let $\mathbf{x} = (x_1, x_2, \ldots, x_{k-1}, z)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_{k-1}, z)$ be distinct nodes of $S_{n,k}$. You may assume that given any two distinct nodes of $S_{n-1,k-1}$, there are $n - 2$ node-disjoint paths joining them.

($a$) Prove that the nodes of $S_{n,k}$ can be partitioned into $n$ sets so that the subgraph induced by the nodes of any of these sets is isomorphic to $S_{n-1,k-1}$. [8 marks]

[Be sure to provide a full explanation.]

($b$) Use your partition to prove that there are $n-1$ node-disjoint paths joining $\mathbf{x}$ and $\mathbf{y}$ in $S_{n,k}$. [12 marks]

**total 20 marks**

# Interconnection Networks Assignment

## Questions *roughly* covered by Lectures 9 and 10

**Question 11.** Implement in Python and submit an extension of your routing algorithm from Question 2($e$) but where the routing algorithm works for any source and any destination of $S_{n,k}$. What are the loads on each node and each channel of $S_{7,4}$ in an all-to-all traffic pattern that result from an execution of your algorithm? Also, what is the maximal path-length and average-path length produced by your algorithm? Finally, you should use your data to make predictions as to the likely symmetry properties of $S_{7,4}$.

**total 30 marks**

[Implementation notes:

- Your implementation should be in the form of a function called `nkstar_routing` and take four input variables:
  - `n` - an integer giving the digit-count for $S_{n,k}$
  - `k` - an integer giving the dimension of $S_{n,k}$
  - `source` - a list of length $k$ detailing the source node
  - `destination` - a list of length $k$ detailing the destination node.
- Your function `nkstar_routing` should return a list of nodes (where a node is a list of length $k$) starting with `source` and ending with `destination` and which details a path from `source` to `destination` (if `source` = `destination` then the list should be [`source`]).
- Your implementation should work for any given values of $n$ and $k$ with $2 \leq k < n$. Marks will be awarded for: a correctly functioning implementation, which will be tested for various values of $k$ and $n$; good loadings and path-lengths within $S_{7,4}$; and a good execution time. This data should be supplied as part of the answer to this question (although I will calculate it myself when I execute your code).
- When calculating the loads on the nodes, only the nodes on a path that are different to the source and destination contribute to the loads; so, a path of length 1 contributes no loads to any node.

- In order to obtain the values requested, you will need to use the code I supply, namely `alltoall_traffic`, so as to generate an all-to-all traffic pattern. The function `alltoall_traffic` takes integers `n` and `k`, which define $S_{n,k}$, as input and outputs two lists, `list_of_sources` and `list_of_destinations`, each of length $[\frac{n!}{(n-k)!}]^2$, so that

$$\{(\texttt{list\_of\_sources}[\texttt{i}], \texttt{list\_of\_destinations}[\texttt{i}]) \\ : i = 0, 1, \ldots, [\tfrac{n!}{(n-k)!}]^2 - 1$$

  consists of all possible source-destination pairs (hence, an item of one of these lists is a list of length $k$ detailing distinct elements of $\{0, 1, \ldots, n-1\}$). You may use any functions in `alltoall_traffic` in your own code.

- You will need to present the loads requested in a sensible format so as to convey the key information (you will certainly *not* present the load on every node and every channel in $S_{7,4}$!). I will execute your code and if there is variance between your claimed values and the ones obtained by my execution then marks will be lost.

- There are $705,600$ source-destination pairs in an all-to-all traffic pattern for $S_{7,4}$ and so you should expect your routing algorithm to take a non-trivial amount of time to terminate.]