# Explain the problem

# The throughput of information transmission within a data network is constrained by the network topology and link capacities. Traditional techniques in improving transmission throughput focus on strategically routing information flow along high bandwidth or multiple paths from the source to the destinations. Recently, it has been shown that such routing strategies alone may not be sufficient. Rather, it is necessary to consider encoding/decoding data on nodes in the network, in order to achieve the optimal throughput. Since these coding operations are not restricted to source or destination nodes, they are referred to as network coding. [from 2004 Network Coding in Undirected Networks]

# Review Multi-Commodity flow

There have been studies on achieving optimality with respect to computing oblivious routing strategies in data networks. The objectives are to maximize throughput for a source-destination pair and to minimize congestion on the network. Most notably, using linear programming techniques, polynomial time algorithms (with a polynomial number of variables and constraints in the LP formulation) can be constructed to compute strategies for optimal oblivious routing for any network, directed or undirected [16], [17]. Though we also employ linear optimization tools and study undirected networks, our problem domain is more general: while optimal oblivious routing focuses on origin-destination pairs of unicast sessions (possibly exploiting path diversity), we focus on a variety of communication sessions, including unicast, multicast, broadcast and group communication. We seek fundamental insights on how optimal a routing strategy may become, and what is the maximum achievable throughput in a communication session.

The theory of network flows [18] studies the transmission of commodities of the same type (unicommodity flows) through a network with known capacities. The maximum flow rate between the source and the destination is known to be equal to the minimum cut between them, which may be computed with efficient algorithms. When commodities to be transmitted are of different types (multicommodity flows), computing the maximum flow rate can be formulated as a linear optimization problem, and then solved using general linear program solvers [18], [19]. In both unicommodity and multicommodity flows, commodities may only be forwarded at intermediate nodes, comparable to all unicast in data networks. [from mflow]

To formulate the multi-commodity problem, represents the fraction of ith flow along the edge (u, v), with a value between 0 to 1. For flow conversation on transit nodes, the incoming amount of flow for every node is equal to the outcoming amount of flow, . Flow conversation at the source or destination is that flow must all come out from the source / all enter the sink.

# Conceptually can actually coded to do better

The concept of *network coding* extends the capabilities of network nodes in a communication session: from basic data forwarding (as in all unicast) and data replication (as in IP or overlay multicast), to *coding in Galois fields*. Fig. 1

illustrates a classic example of how network coding assists to improve end-to-end throughput. As R1 receives both a and a + b (encoded over GF(2)), it is able to decode and retrieve both a and b. If the link capacities are 1, it is apparent that the maximum achievable throughput with network coding is 2. Without coding, it can be computed that the optimal throughput is 1.875. If only one multicast tree is used (as in IP multicast), the achieved throughput is 1.

A diagram of a network code

Description automatically generatedA diagram of a complex algorithm

Description automatically generated with medium confidence

A classic butterfly multicast example that illustrates the power of network coding is shown in Fig. 1, where each link has unit capacity. Node \(s\) is assigned as the source node where the information will be distributed, while node \(t\_1\) and \(t\_2\) are the sink destinations. Consider the case that we want to transmit 2 bits, bit a and bit b, to both node \(t\_1\) and \(t\_2\). Fig 1(a) shows the traditional routing strategy we can do without network coding, and Fig 1(b) shows the case with network coding. In both network coding and traditional cases, s will send its packet to \(t\_1\) and \(t\_2\). However, the difference in performance occurs at node 3. Without network coding, node 3 will be required to do two rounds in order to deliver all received packets to node 4. Node 4 will then be able to broadcast to the two destination nodes. With network coding, however, node 3 could transmit only one packet (a+b) to both \(t\_1\) and \(t\_2\). Thus, a coding advantage of 2 could be found at this directed butterfly example.

# Multicast cases with network coding (batterfly computation+mflow)

A text and a diagram

Description automatically generated with medium confidence

Computation

And mflow code

For different network topologies, the network coding would result in different coding advantage in the throughput of information transmission. To investigate further in here, it involves the calculation of maximum throughput for both with network coding or without network coding (routing only).

In directed networks, the network gain could be infinitely large, with well-designed topology. As shown in \Cref{fig:directed}, a directed network of this pattern with \(2n\) unicast terminals, without network coding, has the maximum throughput bounded by 1 bit due to the bottleneck A-B edge. However, with network coding, each receiver is able to 1 bit of information from \(S\_i\), indicating a coding advantage of \(2n\) .

In 2004, \cite{undirected} studied, for the first time, the network coding advantage for single-session undirected network topology including unicast, boardcast and multicast scenarios. The auther suggested that the coding advantage in an undirected graph is always upper-bounded by a factor of 2, compared to the infinity for directed graph. There is an example in \Cref{fig:undirected}, which is the undirected butterfly network compared to the directed on in \Cref{fig:example}, the optimal throughput with arbitrary fractional routing could achieve at 1.875 in undirected one compared to 1 in directed one. Thus, with the coding optimal throughput of 2, the coding advantage on undirected butterfly network would be \(2 / 1.875 = 1.067\).

For the multiple session cases, there is a special case with multiple unicast sessions, with the researchers still holding conjecture that the use of network coding does not allow any advantage over standard routing, which was first conjectured in \cite{undirected}.

A diagram of a diagram

Description automatically generated A diagram of a triangle with black and white circles and black lines

Description automatically generated

# Multiple Unicast cases – with examples – no benefit

* Cflow mflow examples