Island Hopping and Path Colouring

Andrew McGregor  UPenn → UC San Diego
Bruce Shepherd  Bell Labs → McGill
Optical Network Design
Optical Network Design

- Route a set of signals, $s_i$ to $t_i$, in a graph $G$
Optical Network Design

- Route a set of signals, $s_i$ to $t_i$, in a graph $G$
- Advantages of optical communication:

  A single optical fiber can carry multiple signals if each is assigned a different wavelength.

  Decreased latency if signals can avoid expensive optical-electrical-optical-optical (OEO) conversions.
Optical Network Design

• Route a set of signals, $s_i$ to $t_i$, in a graph $G$

• Advantages of optical communication:

  A single optical fiber can carry **multiple signals** if each is assigned a different wavelength.

  **Decreased latency** if signals can avoid expensive optical-electrical-optical (OEO) conversions.

• Many interesting theory problems arise...
Minimizing Fiber Costs
Minimizing Fiber Costs

- In each link $e$ of $G$, a fiber that can carry a single signal of each wavelength $\{1, ..., \lambda\}$ can be installed with cost $c_e$. 
Minimizing Fiber Costs

- In each link $e$ of $G$, a fiber that can carry a single signal of each wavelength $\{1, ..., \lambda\}$ can be installed with cost $c_e$.

- **MinFiber**: Minimize the cost of fibers installed such that every signal can be routed monochromatically.
Minimizing Fiber Costs

- In each link $e$ of $G$, a fiber that can carry a single signal of each wavelength $\{1, \ldots, \lambda\}$ can be installed with cost $c_e$.

- **MinFiber**: Minimize the cost of fibers installed such that every signal can be routed monochromatically.

- **Approx**: $O(\log n)$ and $\Omega(\log^{1/4-\varepsilon} n)$  

  [Andrews, Zhang ’05]
Minimizing Fiber Costs

- In each link $e$ of $G$, a fiber that can carry a single signal of each wavelength $\{1, \ldots, \lambda\}$ can be installed with cost $c_e$.

- **MinFiber**: Minimize the cost of fibers installed such that every signal can be routed monochromatically.

- **Approx**: $O(\log n)$ and $\Omega(\log^{1/4-\varepsilon} n)$ [Andrews, Zhang ’05]

- **Exact solution if $G$ is a path** [Winkler, Zhang ’03]
Minimizing Fiber Costs

- In each link $e$ of $G$, a fiber that can carry a single signal of each wavelength $\{1, \ldots, \lambda\}$ can be installed with cost $c_e$.

- **MinFiber**: Minimize the cost of fibers installed such that every signal can be routed monochromatically.

- **Approx**: $O(\log n)$ and $\Omega(\log^{1/4-\epsilon} n)$ [Andrews, Zhang '05]

- **Exact solution if $G$ is a path** [Winkler, Zhang '03]

- **Our results**: Exact solution if $G$ is a directed tree and 3.55 approx if demands are single-source.
Minimizing Fiber Costs

• In each link $e$ of $G$, a fiber that can carry a single signal of each wavelength $\{1, ..., \lambda\}$ can be installed with cost $c_e$.

• **MinFiber**: Minimize the cost of fibers installed such that every signal can be routed monochromatically.

• **Approx**: $O(\log n)$ and $\Omega(\log^{1/4-\epsilon} n)$  
  [Andrews, Zhang '05]

• **Exact solution if $G$ is a path**  
  [Winkler, Zhang '03]

• **Our results**: Exact solution if $G$ is a directed tree and 3.55 approx if demands are single-source.
Minimizing “Hops”
Minimizing “Hops”

- At each node, can only switch signals optically within sets of $c$ incident fibers.
Minimizing “Hops”

- At each node, can only switch signals optically within sets of $c$ incident fibers.

![ROADM diagram](image-url)
Minimizing “Hops”

- At each node, can only switch signals optically within sets of $c$ incident fibers.
Minimizing “Hops”

- At each node, can only switch signals optically within sets of $c$ incident fibers.
- Any signal not switched optically requires an OEO conversion or “hop.”
Minimizing “Hops”

- At each node, can only switch signals optically within sets of $c$ incident fibers.
- Any signal not switched optically requires an OEO conversion or “hop.”
- $\text{MinHop}_c$: Given a single infinite capacity fiber in each link, route demands simply and set Roadms to minimize average number of hops.
Minimizing “Hops”

- At each node, can only switch signals optically within sets of $c$ incident fibers.
- Any signal not switched optically requires an OEO conversion or “hop.”
- $\text{MinHop}_c$: Given a single infinite capacity fiber in each link, route demands simply and set Roadms to minimize average number of hops.
Minimizing “Hops”

- At each node, can only switch signals optically within sets of $c$ incident fibers.

- Any signal not switched optically requires an OEO conversion or “hop.”

- $\text{MinHop}_c$: Given a single infinite capacity fiber in each link, route demands simply and set Roadms to minimize average number of hops.

- Approx: $O(\log n)$ and $>2$ for $c=2$  
  [Anshelevich, Zhang ’05]
Minimizing "Hops"

- At each node, can only switch signals optically within sets of $c$ incident fibers.

- Any signal not switched optically requires an OEO conversion or "hop."

- $\text{MinHop}_c$: Given a single infinite capacity fiber in each link, route demands simply and set Roadms to minimize average number of hops.

- Approx: $O(\log n)$ and $>2$ for $c=2$  
  [Anshelevich, Zhang '05]

- Our Results: $\Omega(\log^{1-\epsilon} n)$ for $c=2$...
1. Min-Hop
2. Min-Fiber
3. Min-Both?
MinHop\textsubscript{\textit{c}}

- **Input:** Supply network \( G=(V,E) \) and demands \( H \).
- **Solution:**
  
a) Decomposition of \( E \) into “transparent islands”

b) Simple routing path \( P_h \) for each demand \( h \)

- **Goal:** Minimize average number of times each \( P_h \) needs to hop between transparent islands.
An $O(\log n)$ Approx
Undirected Graphs & 2-arm Roadms
An $O(\log n)$ Approx

Undirected Graphs & 2-arm Roadms

• Choose any spanning tree $T$ of $G$ rooted at $r$
An $O(\log n)$ Approx

Undirected Graphs & 2-arm Roadms

- Choose any spanning tree $T$ of $G$ rooted at $r$
- Route signals along the spanning via $r$
An $O(\log n)$ Approx
Undirected Graphs & 2-arm Roadms

• Choose any spanning tree $T$ of $G$ rooted at $r$
• Route signals along the spanning via $r$
• Setting Roadms optimally ensures each signal requires at most $2 \log n$ hops.  [Anshelevich, Zhang ’05]
An $O(\log n)$ Approx
Undirected Graphs & 2-arm Roadms

- Choose any spanning tree $T$ of $G$ rooted at $r$
- Route signals along the spanning via $r$
- Setting Roadms optimally ensures each signal requires at most $2 \log n$ hops.  
  
[Anshelevich, Zhang ’05]
An $O(\log n)$ Approx

Undirected Graphs & 2-arm Roadms

- Choose any spanning tree $T$ of $G$ rooted at $r$
- Route signals along the spanning via $r$
- Setting Roadms optimally ensures each signal requires at most $2 \log n$ hops.  
  
  [Anshelevich, Zhang '05]
$\Omega(\log^{1-\varepsilon} n)$ Hardness

Undirected Graphs & 2-arm Roadms
\[ \Omega(\log^{1-\epsilon} n) \] Hardness

Undirected Graphs & 2-arm Roadms

- Reduction from **LongPath**:
  - Given a 3-regular Hamiltonian graph find a long path
  - Constant approximation is hard  
    [Bazgan, Santha, Tuza ’99]
\( \Omega(\log^{1-\varepsilon} n) \) Hardness

Undirected Graphs & 2-arm Roadms

- Reduction from **LongPath**:
  
  Given a 3-regular Hamiltonian graph find a long path
  
  Constant approximation is hard [Bazgan, Santha, Tuza '99]

- Let \( L \) be an instance of **LongPath** on \( t \) nodes:
  
  Replace each node \( u \) with \( K_{2,3} = \{u_1, u_2, v_1, v_2, v_3\} \) and match \( v_1, v_2, v_3 \) to neighbours of \( u \).
$\Omega(\log^{1-\varepsilon} n)$ Hardness
Undirected Graphs & 2-arm Roadms

- Reduction from **LongPath**:
  Given a 3-regular Hamiltonian graph find a long path
  Constant approximation is hard [Bazgan, Santha, Tuza '99]
- Let $L$ be an instance of **LongPath** on $t$ nodes:
  Replace each node $u$ with $K_{2,3} = \{u_1, u_2, v_1, v_2, v_3\}$ and
  match $v_1, v_2, v_3$ to neighbours of $u$. 

![Diagram of graph transformation](image)
\( \Omega(\log^{1-\varepsilon} n) \) Hardness

Undirected Graphs & 2-arm Roadms

Insert multiple copies of \( L' \) into \((t-1)\)-ary tree in which each edge is duplicated.
$\Omega(\log^{1-\epsilon} n)$ Hardness

Undirected Graphs & 2-arm Roadms

Consider demands from leaves to root

Insert multiple copies of $L'$ into $(t-1)$-ary tree in which each edge is duplicated.
$\Omega(\log^{1-\epsilon} n)$ Hardness

Undirected Graphs & 2-arm Roadms

Consider demands from leaves to root
$L$ is Hamiltonian so $\text{MinHop}_2(G) = 1$

Insert multiple copies of $L'$ into $(t-1)$-ary tree in which each edge is duplicated.
\( \Omega(\log^{1-\varepsilon} n) \) Hardness

Undirected Graphs & 2-arm Roadmats

Consider demands from leaves to root

\( L \) is Hamiltonian so \( \text{MinHop}_2(G) = 1 \)

Finding a solution of cost \( o(\log^{1-\varepsilon} n) \) requires finding length \( \Omega(t) \) path in \( L \).

Insert multiple copies of \( L' \) into \((t-1)\)-ary tree in which each edge is duplicated.
$\Omega(n^{1-\varepsilon})$ Hardness

Directed Graphs & 2-arm Roadms
Ω(n^{1-\varepsilon}) Hardness

Directed Graphs & 2-arm Roadms

- Reduction from 2DirPaths:
  For directed graph L and s_1, t_1, s_2, t_2, it is NP-hard to determine if there is edge disjoint paths between s_1 and t_1; and s_2 and t_2.

[Fortune, Hopcroft, Wyllie '80]
\( \Omega(n^{1-\varepsilon}) \) Hardness

Directed Graphs & 2-arm Roadms

- Reduction from 2DirPaths:
  For directed graph \( L \) and \( s_1, t_1, s_2, t_2 \), it is NP-hard to determine if there is edge disjoint paths between \( s_1 \) and \( t_1 \); and \( s_2 \) and \( t_2 \).
  [Fortune, Hopcroft, Wyllie ’80]

- Form supply graph \( G \) with demands \( (a,b) \) and \( (b,a) \)
$\Omega(n^{1-\epsilon})$ Hardness

Directed Graphs & 2-arm Roadms

- Reduction from 2DirPaths:
  For directed graph $L$ and $s_1,t_1,s_2,t_2$, it is NP-hard to determine if there is edge disjoint paths between $s_1$ and $t_1$; and $s_2$ and $t_2$.
  
  [Fortune, Hopcroft, Wyllie ’80]

- Form supply graph $G$ with demands $(a,b)$ and $(b,a)$

- If there exists edge disjoint paths then $\text{MinHop}_2(G) = 1$ and otherwise $\text{MinHop}_2(G) = \Omega(n^{1-\epsilon})$. 

**Ω(𝑛^{1-ε})** Hardness

**Directed Graphs & 2-arm Roadmaps**

- **Reduction from 2DirPaths:**
  For directed graph \( L \) and \( s_1, t_1, s_2, t_2 \), it is NP-hard to determine if there is edge disjoint paths between \( s_1 \) and \( t_1 \); and \( s_2 \) and \( t_2 \).
  
  [Fortune, Hopcroft, Wyllie '80]

- Form supply graph \( G \) with demands \((a,b)\) and \((b,a)\)

- If there exists edge disjoint paths then \( \text{MinHop}_2(G) = 1 \) and otherwise \( \text{MinHop}_2(G) = \Omega(n^{1-\epsilon}) \).

- Can assume \( G \) is strongly connected...
An $O(n^{1/2})$ Approx
Directed Acyclic Graphs & 2 arm Roadms
An $O(n^{1/2})$ Approx
Directed Acyclic Graphs & 2 arm Roadms

- Thm: An $O(n^{1/2})$ approximation for DAGS.
An $O(n^{1/2})$ Approx
Directed Acyclic Graphs & 2 arm Roadms

- **Thm:** An $O(n^{1/2})$ approximation for DAGS.

- **Lemma:** Call a sequence $a_1, ..., a_n$ boosted if $a_i \neq a_{i+1}$ and if $a_i = a_k$, then $a_j \leq a_k$ for all $i < j < k$. Length of a boosted sequence with alphabet $\{1, 2, ..., k\}$ is at most $2k$. 
An $O(n^{1/2})$ Approx
Directed Acyclic Graphs & 2 arm Roadms

• **Thm:** An $O(n^{1/2})$ approximation for DAGS.

• **Lemma:** Call a sequence $a_1, ..., a_n$ *boosted* if $a_i \neq a_{i+1}$ and if $a_i = a_k$, then $a_j \leq a_k$ for all $i < j < k$. Length of a boosted sequence with alphabet $\{1, 2, ..., k\}$ is at most $2k$.

**Proof:** Induction on $k$:

$k=1$ trivial!

Let $q$ be minimum repeated element and let sequence be of the form $S\, q \, I_1 \, q \, I_2 \, q \, ... \, I_j \, q \, P$.

Assume $I_1 \, I_2 \, ... \, I_j$ has length $r$ and so $I_1 \, q \, I_2 \, q \, ... \, I_j \, q$ has length at most $2r$.

Sequence $S\, q \, P$ is boosted and has alphabet size $k-r$ hence length is $2(k-r)$ by induction.
An $O(n^{1/2})$ Approx
Directed Acyclic Graphs & 2 arm Roadms

• Thm: An $O(n^{1/2})$ approximation for DAGS.

Proof (Sketch):

Define “long” paths $P_1, P_2, \ldots, P_k$ that route some demands

$P_j$: If shortest route in $G$ augmented by edges of distance $n^{-2}$ between all pairs of nodes in $P_i$ for all $i<j$ is length at least $n^{1/2}$ then let $P_j$ be this route.

Define transparent islands as maximal sub-paths of $P_j \setminus (P_1, \ldots, P_{j-1})$ and all remaining individual edges.

$G$ is a DAG implies that $k=O(n^{1/2})$

Boosting lemma implies every routing requires $O(k)$ hops.
Multi-arm Roadmaps
Multi-arm Roadms

- **Thm:** MinHop$_3$ has $O(\log n)$ approx for strongly connected graphs. (c.f. $\Omega(n^{1-\varepsilon})$ MinHop$_2$)
Multi-arm Roadms

• **Thm:** MinHop$_3$ has $O(\log n)$ approx for strongly connected graphs. (c.f. $\Omega(n^{1-\varepsilon})$ MinHop$_2$)

• **Thm:** For $G$ planar:

  $\text{MinHop}_2(G)=1$ if $G$ is 4-node connected.

  *Graph is Hamiltonian [Tutte ’56] and a degree-3 spanning tree can be found in polytime [Fürer, Raghavachari ’56].*
Multi-arm Roadms

- **Thm:** MinHop$_3$ has $O(\log n)$ approx for strongly connected graphs. (c.f. $\Omega(n^{1-\epsilon})$ MinHop$_2$)

- **Thm:** For $G$ planar:

  \[
  \text{MinHop}_2(G) = 1 \text{ if } G \text{ is 4-node connected.}
  \]

  *Graph is Hamiltonian [Tutte ’56] and a degree-3 spanning tree can be found in polytime [Fürer, Raghavachari ’56].*

  \[
  \text{MinHop}_3(G) = 1 \text{ if } G \text{ is 3-node connected.}
  \]

  *Graph has a degree-3 spanning tree and this can be found in polytime [Barnette ’66].*
Multi-arm Roadms

• **Thm:** MinHop\(_3\) has \(O(\log n)\) approx for strongly connected graphs. (c.f. \(\Omega(n^{1-\epsilon})\) MinHop\(_2\))

• **Thm:** For \(G\) planar:

\[
\text{MinHop}_2(G) = 1 \text{ if } G \text{ is 4-node connected.}
\]

*Graph is Hamiltonian* [Tutte '56] and *a degree-3 spanning tree can be found in polytime* [Fürer, Raghavachari '56].

\[
\text{MinHop}_3(G) = 1 \text{ if } G \text{ is 3-node connected.}
\]

*Graph has a degree-3 spanning tree and this can be found in polytime* [Barnette '66].

\[
\text{MinHop}_3(G) = \Omega(\log n) \text{ for some 2-node connected } G.
\]
Multi-arm Roadms

- **Thm:** MinHop\(_3\) has \(O(\log n)\) approx for strongly connected graphs. (c.f. \(\Omega(n^{1-\varepsilon})\) MinHop\(_2\))

- **Thm:** For \(G\) planar:

\[
\text{MinHop}_2(G) = 1 \text{ if } G \text{ is 4-node connected.}
\]

*Graph is Hamiltonian* [Tutte '56] *and a degree-3 spanning tree can be found in polytime* [Fürer, Raghavachari '56].

\[
\text{MinHop}_3(G) = 1 \text{ if } G \text{ is 3-node connected.}
\]

*Graph has a degree-3 spanning tree and this can be found in polytime* [Barnette '66].

\[
\text{MinHop}_3(G) = \Omega(\log n) \text{ for some 2-node connected } G.
\]
## Summary of MinHop

<table>
<thead>
<tr>
<th></th>
<th>2-arm Roadms</th>
<th>3-arm Roadms</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Algorithm</strong></td>
<td><strong>Hardness</strong></td>
<td><strong>Algorithm</strong></td>
</tr>
<tr>
<td><strong>Undirected</strong></td>
<td>$O(\log n)$</td>
<td>$\Omega(\log^{1-\varepsilon} n)$</td>
</tr>
<tr>
<td><strong>Strongly Connected</strong></td>
<td>$O(n)$</td>
<td>$\Omega(n^{1-\varepsilon})$</td>
</tr>
<tr>
<td><strong>DAG</strong></td>
<td>$O(n^{1/2})$</td>
<td>$\Omega(\log n)$</td>
</tr>
</tbody>
</table>
## Summary of MinHop

<table>
<thead>
<tr>
<th></th>
<th>2-arm Roadms</th>
<th>3-arm Roadms</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Algorithm</strong></td>
<td>O(log n)</td>
<td>O(log n)</td>
</tr>
<tr>
<td><strong>Hardness</strong></td>
<td>Ω(log^{1-ε} n)</td>
<td>Ω(n^{1-ε})</td>
</tr>
<tr>
<td><strong>Undirected</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strongly Connected</td>
<td>O(n)</td>
<td>O(log n)</td>
</tr>
<tr>
<td>DAG</td>
<td>O(n^{1/2})</td>
<td>O(n^{1/2})</td>
</tr>
</tbody>
</table>

- **Open Question:** Resolve the hardness of directed acyclic graphs
1. Min-Hop
2. Min-Fiber
3. Min-Both?
MinFiber

- **Input:** Supply network $G=(V,E)$, demand graph $H$, costs $c_e$ to install a fiber in link $e$, and fiber capacity $\lambda$.

- **Solution:**
  
  a) Multiple $l_e$ of fibers at link $e$
  
  b) Simple routing path $P_h$ for each demand
  
  c) Assignment of one of $\lambda$ colours to each $P_h$ such that the number of paths of the same colour using any edge is at most $l_e$.

- **Goal:** Minimize $\sum c_e l_e$
Integer Decomposition Property
A polyhedron $P$ has the **integer decomposition property (IDP)** if for any $x \in P$ and integer $k$ such that $kx$ is integral then we have

$$kx = \sum_{i \in [k]} x_i$$

where $x_i$ is an integral vector in $P$. 

**Integer Decomposition Property**
A polyhedron $P$ has the **integer decomposition property (IDP)** if for any $x \in P$ and integer $k$ such that $kx$ is integral, then we have

$$kx = \sum_{i \in [k]} x_i$$

where $x_i$ is an integral vector in $P$.

**Thm (Baum, Trotter):** Matrix $A$ is totally unimodular iff $\{x : Ax \leq b, x \geq 0\}$ has the IDP for every integer vector $b$. 

**Integer Decomposition Property**

- A polyhedron $P$ has the integer decomposition property (IDP) if for any $x \in P$ and integer $k$ such that $kx$ is integral then we have $kx = \sum_{i \in [k]} x_i$ where $x_i$ is an integral vector in $P$.

- **Thm (Baum, Trotter):** Matrix $A$ is totally unimodular iff $\{x : Ax \leq b, x \geq 0\}$ has the IDP for every integer vector $b$. 

WDM Flows on Directed Trees

- **Thm:** Exact solution MinFib on directed tree instances.
- **Proof (Sketch):**

  Let $B$ be the matrix with $B_{ah} = 1$ if routing for demand $h$ goes through arc $a$. $B$ and $[B^T l]^T$ are totally unimodular.

  Let $l$ an allocation of fibers that satisfies capacity requirements.

  Define $P_l = \{ x : B \cdot x \leq l, 0 \leq x \leq 1 \}$ and note $P_l$ is IDP.

  By assumption $(1/\lambda, 1/\lambda, ..., 1/\lambda)$ is in $P_l$ and hence there exists a decomposition of demands into $\lambda$ classes such that each class can be assigned the same colour.
1. Min-Hop
2. Min-Fiber
3. Min-Both?
Open Question

- **Incompatible Assumptions:**
  
  MinHop assumes an existing infinite capacity fiber in each link.
  
  MinFiber assumes full wavelength selective switching (i.e. infinite-arm Roadms)

- **How can we unify both problems?**
  
  In MinHop, consider purchasing extra fibers in each link at some cost.
  
  If we have to hop, can’t we get a wavelength conversion for free?
MinFiber:
Exact Solution for Directed Trees
3.55 Approximation for Single-Source
via “Fractional implies Integral” results

MinHop:

<table>
<thead>
<tr>
<th></th>
<th>2-arm Roadms</th>
<th>3-arm Roadms</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Algorithm</td>
<td>Hardness</td>
</tr>
<tr>
<td><strong>Undirected</strong></td>
<td>$O(\log n)$</td>
<td>$\Omega(\log^{1-\epsilon} n)$</td>
</tr>
<tr>
<td><strong>Strongly Connected</strong></td>
<td>$O(n)$</td>
<td>$\Omega(n^{1-\epsilon})$</td>
</tr>
<tr>
<td><strong>DAG</strong></td>
<td>$O(n^{1/2})$</td>
<td>$\Omega(\log n)$</td>
</tr>
</tbody>
</table>