Lecture 3:
The Traveling Salesman Problem: Inequalities and Separation

Outline:

1. The ILP formulation of the symmetric TSP
2. Survey of valid inequalities and facets
3. Exact separation based on templates
4. Heuristic separation based on templates
5. Non-template-based separation
6. The asymmetric TSP
7. Conclusions and open problems

1. ILP formulation of the symmetric TSP

Standard formulation due to Dantzig, Fulkerson & Johnson (1954):

Let V be the vertex set and E the edge set.

Define a 0-1 variable x_e for each $e \in E$.

For any $S \subset V$, let:

$E(S)$ = edges with both end-vertices in S.
$\delta(S)$ = edges with one end-vertex in S.

For any $F \subset E$, $x(F)$ denotes $\sum_{e \in F} x_e$.

If c_e is the cost of edge e, we have:

Minimise $\sum_{e \in E} c_e x_e$

Subject to:

$x(\delta(i)) = 2 \quad (\forall i \in V)$ (degree equations)

$x(E(S)) \leq |S| - 1 \quad (\forall S \subset V)$ (SECs)

$x \in \{0, 1\}^{|E|}$ (binary condition)

2. Survey of valid inequalities and facets.

The degree equations define the affine hull.
The trivial bounds $0 \leq x_e \leq 1$ induce facets.

The SECs induce facets. Note that there are an exponential number of them.

(An upper bound $x_e \leq 1$ is equivalent to an SEC with $|S| = 2$.)

The polyhedron defined by the degree equations, non-negativity inequalities, and SECs is called the *subtour elimination polytope* and denoted by SEP(n).

The convex hull of integer solutions is called the *symmetric traveling salesman polytope* and denoted by STSP(n).

For $3 \leq n \leq 5$, SEP(n) = STSP(n).

However, for $n \geq 6$, STSP(n) is strictly contained in SEP(n) and more inequalities are needed.

2.1. 2-matching inequalities

Discovered by Edmonds (1965) in the context of matching problems.

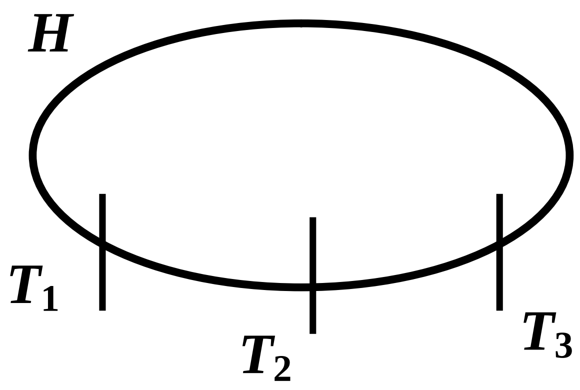
For any set $H \subset V$ and any edge set $F \subset \delta(H)$ such that $|F|=p$ is odd, we have:

$$x(E(H)) + x(F) \leq |H| + \lfloor p/2 \rfloor.$$

Can be written in various other ways, for example:

$$x(\delta(H) \setminus F) \geq x(F) - |F| + 1.$$

The set H is now called the *handle* and the edges in F are called *teeth*.

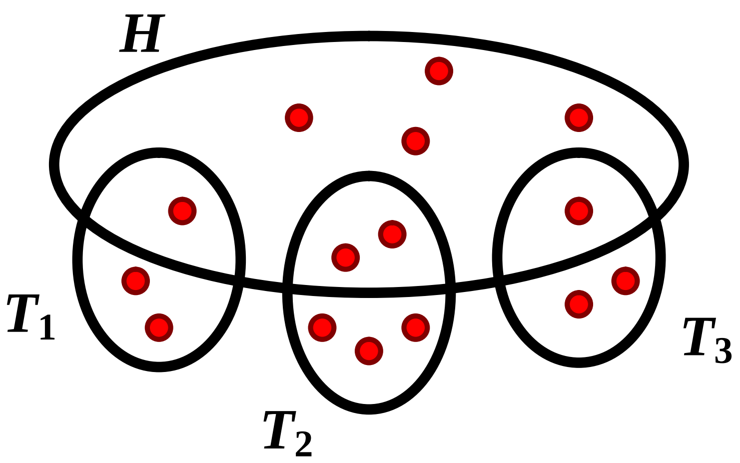


2.2. Comb inequalities

Due to Grötschel and Padberg (1979).

A comb consists of a vertex set H (the handle) and vertex sets $T_1, \ldots, T_p$ (the teeth) such that:

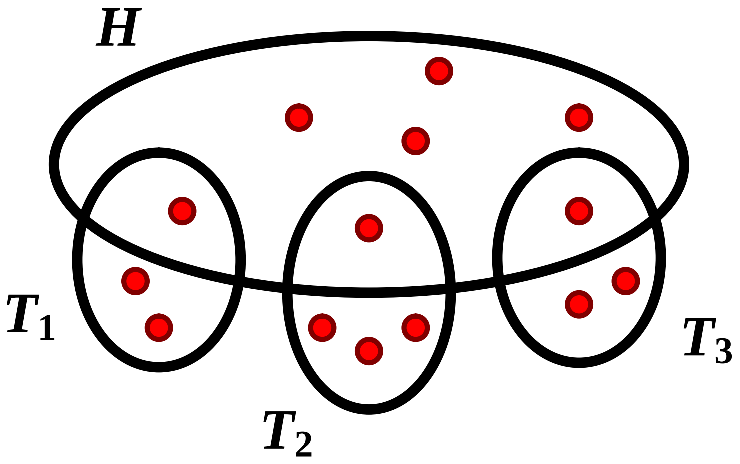
> $p \geq 3$ and odd;
> all teeth are disjoint;
> $H \cap T_j$ and $T_j \setminus H$ are non-empty for all j.



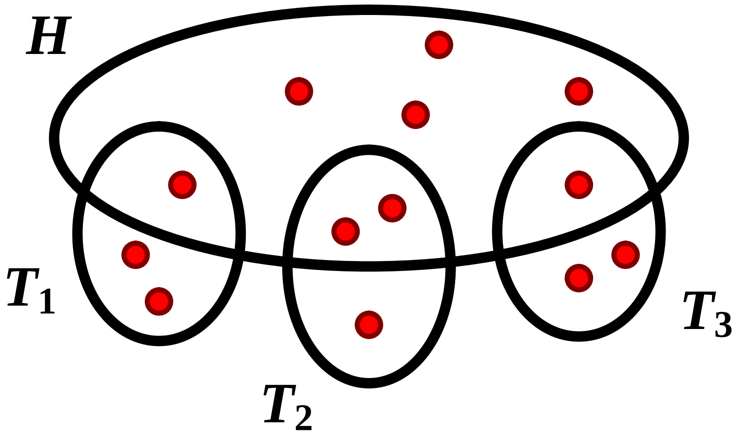
The comb inequality is:

$$x(E(H)) + \sum_{j=1}^{p} x(E(T_j)) \leq |H| + \sum_{j=1}^{p} |T_j| - \lceil 3p/2 \rceil .$$

A *Chvátal* comb inequality also satisfies:
$|H \cap T_j| = 1$ for all j (Chvátal, 1973).



Letchford & Lodi (2002) define *simple* comb inequalities, in which, for all j, either $|H \cap T_j| = 1$ or $|T_j \setminus H| = 1$.



So from most general to least: comb, simple comb, Chvátal comb, 2-matching.

2.3. Other 'handle-tooth' inequalities

Many other known inequalities can be expressed in the form:

$$\sum_{j=1}^{p} x(E(H_j)) + \sum_{j=1}^{q} x(E(T_j)) \le \text{RHS},$$

or, equivalently, in the form

$$\sum_{j=1}^{p} x(\delta(H_j)) + \sum_{j=1}^{q} x(\delta(T_j)) \ge \text{RHS}.$$

These include:

Clique-tree inequalities
(Grötschel & Pulleyblank, 1986)

Path inequalities
(Cornuéjols, Fonlupt & Naddef, 1985)

Star inequalities (Fleischmann, 1988)

Hyperstar inequalities
(Fleischmann, unpublished)

Bipartition inequalities
(Boyd & Cunningham, 1991)

Binested inequalities (Naddef, 1992)

2.4. Miscellaneous inequalities

Not all inequalities are of “handle-tooth” type:

Hypohamiltonian (Grötschel, 1980),
Chain (Padberg & Hong, 1980),
Crown (Naddef & Rinaldi, 1992),
Ladder (Boyd et al., 1995)

Indeed, Christof, Jünger & Reinelt (1991, 1995, 1996) have studied STSP(n) for $n \leq 10$ and the majority of the facets are not of handle-tooth type.

Various other facets have been discovered, e.g., by Naddef & Rinaldi.

3. Exact separation based on templates

Following Applegate, Bixby, Chvátal & Cook, we say that a specified class of facet-inducing inequalities (SECs, combs, etc.) is a *template*.

To use inequalities from a template in a cutting plane algorithm, we need to solve the following *separation problem* (Grötschel, Lovász & Schrijver 1988):

> *Given a vector $x^* \in \Re^{|E|}$ as input, either find an inequality in the template which is violated by x^*, or prove that none exists.*

Exact polynomial time separation algorithms are known for:

i) SECs (Crowder & Padberg, 1980)
ii) 2-matching ineq.s (Padberg & Rao, 1982)
iii) clique tree inequalities with a fixed number of handles and teeth (Carr, 1997)
iv) certain inequalities defined by lifting (Carr, 1996, 1997).

3.1 Subtour elimination constraints

If we write the SECs in the form $x(\delta(S)) \geq 2$, we see that the separation problem reduces to a minimum cut problem.

The *support graph* G^* is the graph with vertex set V and edge set $E^* = \{e \in E: x^*_e > 0\}$.

Each $e \in E^*$ is given the weight x^*_e.

Then, a violated SEC exists if and only if there is a cut in G^* whose x^*-weight is less than 2.

We can therefore use any min-cut algorithm (e.g., Gomory – Hu, Padberg – Rinaldi, Hao – Orlin…).

Current fastest is $O(nm + n^2 \log n)$ (Nagamochi, Ono & Ibaraki, 1994).

3.2 2-matching inequalities

Padberg & Rao (1982) noted that the inequality can be re-written as:

$$x(\delta(H) \setminus F) + \sum_{e \in F} (1 - x_e) \geq 1,$$

where $F \subset \delta(H)$ is the set of teeth.

This enabled them to reduce the separation problem to that of finding a minimum weight odd cut in the so-called 'split graph'.

Their algorithm uses O(m) max-flows in a graph with $m + n$ vertices and $2m$ edges.

Grötschel & Holland (1987) reduced this to O(m) max-flows in graphs with $n + 2$ vertices and $m + 2$ edges.

Letchford, Reinelt & Theis (2003) showed how to reduce this to n - 1 max-flows in G^* itself. This leads to $O(n^2 m \log (n^2/m))$ time.

3.3 Carr's separation algorithms

Carr's original separation algorithm was based on enumeration of possible configurations ("backbones"), plus the solution of a sequence of max-flows.

E.g., for a comb with p teeth, there are $O(n^{2p})$ possible configurations, thus $O(n^{2p})$ max-flow problems need to be solved

The same idea works for any handle-tooth template, but the order of the polynomial grows rapidly with the number of handles and teeth.

Later, he showed how to separate inequalities from other templates, not necessarily handle-tooth based, by solving an LP for each backbone.

Impractical but theoretically elegant.

4. Heuristic separation based on templates

A *heuristic* separation algorithm (for a given template) outputs either one or more inequalities in the template violated by x^*, or a failure message.

SECs: connected components (folklore)
shrinking (Crowder & Padberg, 1980)
spanning trees (Fischetti et al.)
segments (ABCC).

2-matching / Chvátal comb: usually based on *blocks* in the *fractional graph* (Grötschel & Holland, 1991; Padberg & Rinaldi, 1990; Naddef & Clochard, 1994; Naddef & Thienel, 2002…)

General comb: tend to be based on shrinking, followed by heuristics for Chvátal comb separation.

Other inequalities: Clochard & Naddef, 1993; Naddef & Thienel, 2002; ABCC…

The small instance approach (Heidelberg):

List all facets of STSP(n) up to n = 10.

Put them into equivalence classes

For each class:

Shrink G^* to a small graph.
Heuristically solve a QAP.

Not competitive at present (explained later...)
But can be easily parellelised.

5. Non-template-based separation

Instead of concentrating on a specific class of inequalities, it seems to be better to look at *the way in which the inequalities are derived*.

E.g., the comb inequalities can be derived as so-called *{0, ½}-cuts* (Caprara, Fischetti & Letchford, 2000).

CFL (2000): $O(n^2m)$ exact algorithm for detecting *maximally violated* {0, ½}-cuts.

Letchford (2000): $O(n^3)$ exact separation algorithm for a class of {0, ½}-cuts containing all combs, when the support graph is *planar*.

Letchford & Lodi (2002): $O(n^3m^3 \log n)$ exact separation algorithm for a class of {0, ½}-cuts containing all simple combs.

Fleischer, Letchford & Lodi (2003) reduce running time to $O(n^2m^2 \log (n^2/m))$.

Caprara & Letchford (2003) show that in the case of the STSP, the {0, ½}-cuts can be derived by a special disjunctive technique based on handles.

This led to an exact separation algorithm when the handle is *fixed*.

Finally, Applegate, Bixby, Chvátal & Cook introduced the idea of *local cuts*: as in the Heidelberg approach, the support graph is shrunk to a smaller one.

However, instead of resorting to a long list of templates, they use column generation.

The templates are dealt with “implicitly”.

6. The Asymmetric TSP

Every inequality for the STSP has a counterpart for the ATSP.

So any separation algorithm for the STSP can also be used for the ATSP.

There are also many asymmetric inequalities known:

D_k, C3 (Grötschel & Padberg)
Odd CAT (Balas)
Source-Destination (Balas & Fischetti)
A-path (Chopra & Rinaldi)
Lifted cycle (Balas & Fischetti)

Fischetti & Toth found an exact separation algorithm for D_k inequalities…

… which was later proven polynomial.

They also gave a good separation heuristic for the Odd CAT inequalities, again based on the idea of {0, ½}-cuts.

7. Conclusions and Open Problems

Not just of theoretical interest. On average TSPLIB instance:

SECs bring within 98% of optimal
Combs within 99.5%
Local cuts/DPIs within 99.8%.

Only with these last inequalities is it possible to solve very large STSP instances.

Main lesson: seems better to look at *methods for deriving inequalities*, rather than templates.

Key Open Problems:

- complexity of comb separation (or a superclass such as DPIs.

- separation for ATSP, especially Odd CAT, SD and lifted cycle inequalities.

- worst-case ratios (4/3 for STSP using SECs? 6/5 using combs?)