Engs 104, W2010

Optimization

George Cybenko

Today

Examples of NP-Complete Problems

Travelling Salesman Problem

Hamiltonian Circuit

Partition

Knapsack

Travelling Salesman Problem (TSP)

n nodes

$d_{ij} = d_{ji}$ distances between nodes i and j (symmetric), integers

Integer L

Question: Is there a "tour" $1 \to i_2 \to i_3 \to \dots \to i_n \to 1 = i_1$ so that

$$\sum_{j=1}^{n} d_{i_j i_{j+1}} \le L \qquad (i_{n+1} = i_1)$$

<u>Note</u> We really want the minimal distance tour i.e.

$$\min \sum_{j=1}^{n} d_{i_j i_{j+1}}$$

As a decision problem, check if a tour costing $\leq L$ exists.

If "yes", try $L/2 = L$

If "no", try $2L = L$

Always have L, U so that

$\min \leq U$ is <u>no</u>, $\min \leq L$ is yes

Start with $U = 0$, $L = \sum_{ij} d_{ij}$

Note: L requires $\leq$ # bits to store TSP

If $U \leq \min \leq L$,

Try $\frac{U+L}{2}$. If "yes", $L = \frac{U+L}{2}$

If "no", $U = \frac{U+L}{2}$

Each step like this identifies one bit of the actual min and it requires no more than # bits to store TSP $\Rightarrow$ polynomial # steps

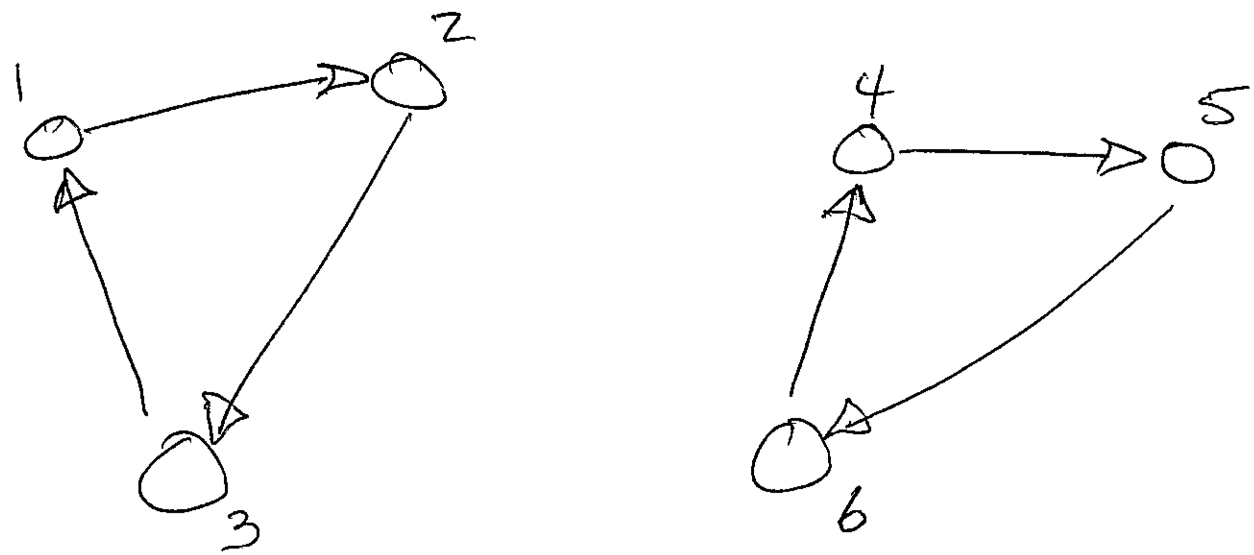
Consider this formulation of TSP as an Integer Linear Program:

$$\min \sum_{i,j} x_{ij} d_{ij}$$

$x_{ij} = 1$ if $i \to j$ on a "tour"

$$\left\{ \begin{array}{ll} \sum_i x_{ij} = 1, & \text{for all } i \\ \sum_j x_{ij} = 1, & \text{for all } j \\ x_{ij} = 0 \text{ or } 1 & \end{array} \right.$$

What's wrong with this?



These constraints do not prohibit disconnected tours such as above:

$$\sum_i x_{ij} = 1 = \sum_j x_{ij} \quad \text{satisfied!}$$

In fact, this is just the assignment problem!

<u>Trick</u>: Add subtour elimination constraints:

n+1 cities, not u_0 though

- New variables $u_1, \ldots, u_n$
- Constraints

$$u_i - u_j + n x_{ij} \leq n-1$$

for $1 \leq i \neq j \leq n$

If $i_1, \ldots, i_k$ is a circuit/tour without city 0,

$$u_{i_j} - u_{i_{j+1}} + n \leq n-1$$

so $Kn \leq K(n-1)$ contradiction!

Conversely, if we let $u_i = t$ if i is the t^{th} city on the tour, then

$$\underbrace{u_i - u_j + 0n}_{\leq n-1} \leq n-1 \quad \Big\} \text{ if } i \to j \text{ not on tour}$$

$$t - (t+1) + n = n-1 \leq n-1 \quad \Big\} \text{ if } i \to j \text{ is on the tour}$$

So adding $\approx n^2$ constraints makes TSP an ILP.

Hamiltonian Circuit Problem

Given undirected graph $G = (V, E)$ is there a circuit/tour using only edges in E? ie visit nodes/cities only once and start and end at same node/city?

Reduce to TSP as follows:

Embed $G = (V, E)$ into

$H = (V, E')$ as follows:

E' includes all edges in E

Cost of an edge in E is $\underline{1}$

Cost of an edge in $E' - E$ is 2

Is there a solution to TSP

with cost $\leq$ # nodes ?

So if we can solve TSP we can solve HC (efficiently)!

If we can show that HC is NP-Complete, then TSP is also NP-Complete.

To show HC is NP-Complete reduce 3-SAT to HC.

Construct a graph with an HC $\Leftrightarrow$ 3-SAT is satisfiable

not be feasible
corresponding
rresponding to
and the second
are executed in
rresponding to
can include all
tices. Hence the
ue $C = \{v: S(v)$

sible schedule S
all i; $S(v) = 1$
and only if v, u
in Fig. 15–7(c).
□

TSP and related

e shall now show
TON CIRCUIT.
C_m and involving

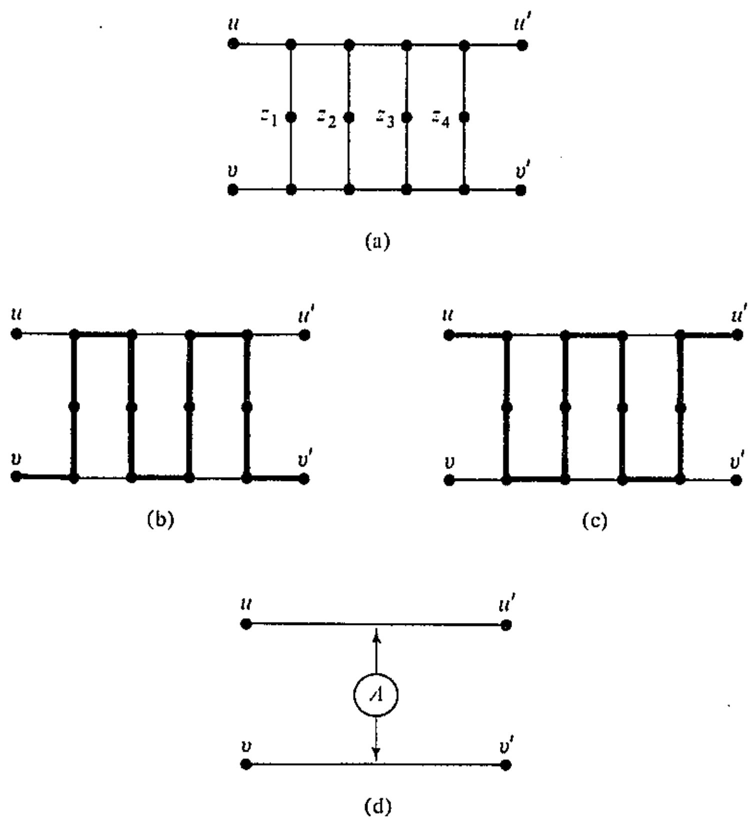


Figure 15–8

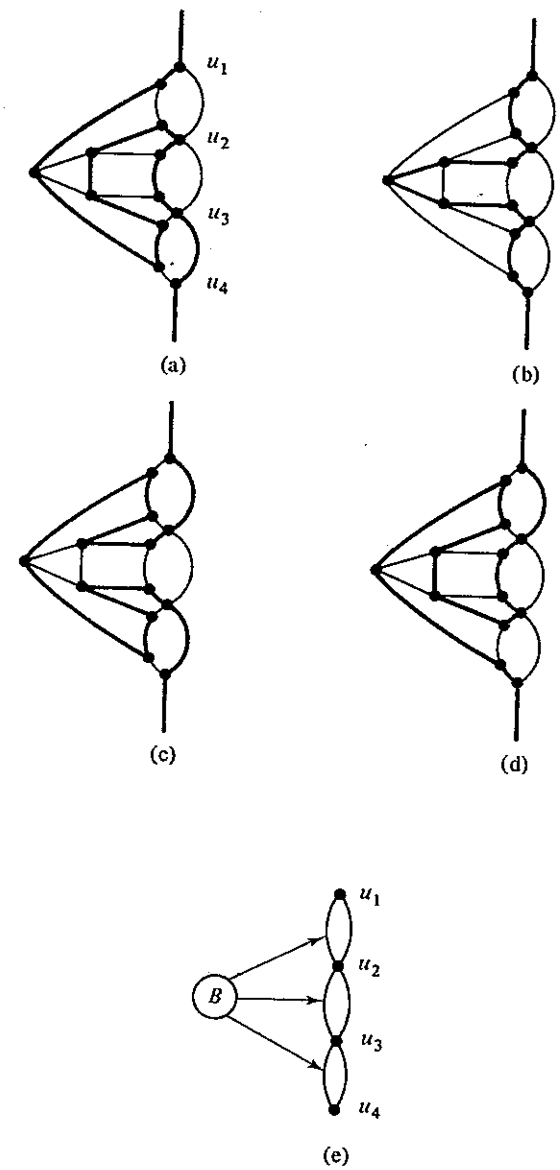
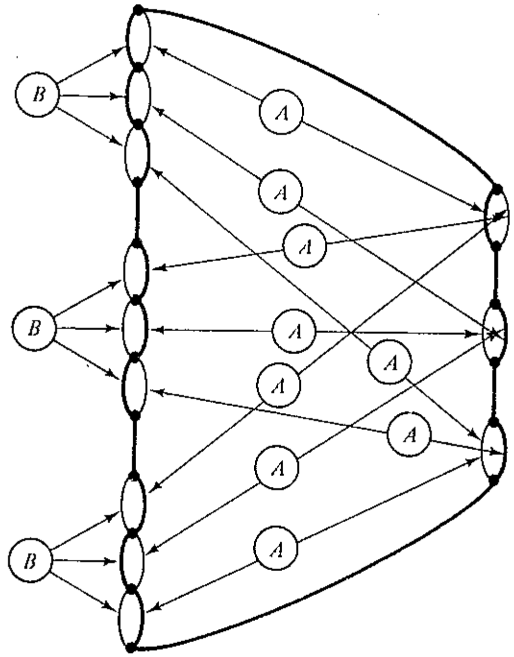


Figure 15–9

Notice that, so far, only the parameters m and n of our formula have (into the construction of G. Since G is supposed to capture the intricacy satisfiability question for F, we must now take into account the exact na the clauses of F. Thus we connect (via the A-connector) the edge $[u_{ij}, u_{i,j+}$ the left copy of $[v_k, w_k]$ in the case that the jth literal of C_i is x_k, and w



$F = (x_1 + \bar{x}_2 + x_3)(\bar{x}_1 + x_2 + \bar{x}_3)(\bar{x}_1 + \bar{x}_2 + x_3)$

The Hamilton circuit shown corresponds to:

$t(x_1) =$ *true*
$t(x_2) =$ *false*
$t(x_3) =$ *false*

Figure 15–10

There are <u>many</u> NP-Complete problems.

<u>Examples</u>: 3-Dimensional Matching

Three sets, U, V, W with

$n = |U| = |V| = |W|$.

$T \subseteq U \times V \times W$

Question: is there an $M \subseteq T$ with

$|M| = n$ and if $(u_1, v_1, w_1), (u_2, v_2, w_2) \in T$

then $u_1 \neq u_2$, $v_1 \neq v_2$, $w_1 \neq w_2$

Note: 2-Dim Matching is "easy"

Show that 2-D Matching is a special case of the Assignment Problem.

Applications of 3-D Matching:

Multi-Target Tracking

Engs 104, L15

Knapsack Problems

Given: Integers c_i, $i = 1, \ldots, n$ & K

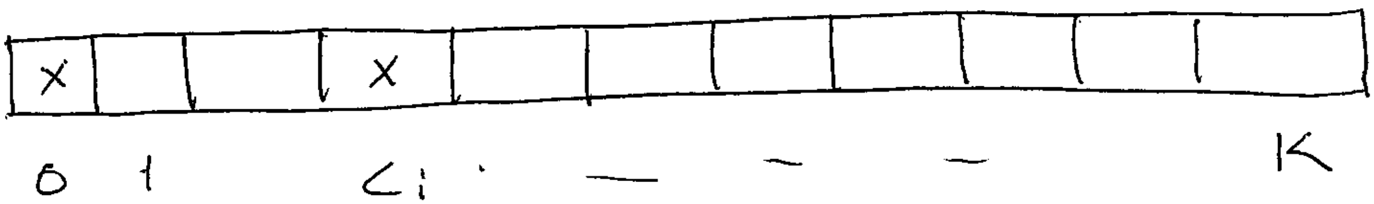
Question: Are there integers $x_i \geq 0$ so that $\sum_i c_i x_i = K$?

0-1 Knapsack Integers x_i are 0, 1

Knapsack has volume K, c_i are volumes of items. Can you fill the knapsack?

Clever Algorithm for 0-1 Knapsack

array



Start with array of length K.

All entries are initially "unmarked" (=0)

for $i = 1, \ldots, n$

Mark entry c_i

If an entry is marked, mark entry + c_i

end

If entry K is marked → done!

What is "wrong" with this approach?

Problem size is # bits needed to specify the problem but this solutions uses $2^{\# \text{bits}}$ memory & steps.

Partition Problem

Given $c_1, \ldots, c_n$ is there a subset $S \subset \{1, \ldots, n\}$ so that

$$\sum_{i \in S} c_i = \sum_{i \notin S} c_i = \frac{1}{2} \sum_{i \in \{1, \ldots, n\}} c_i$$

i.e. Knapsack Problem with $K = \frac{1}{2}$ total sum.

Clique Problem

Given graph $G = (V, E)$ and integer $K > 0$ ~~of~~ is there a clique of size K?

Clique = set of completely connected nodes, size = # of nodes.

Maximum clique is also (therefore) NP-Complete.

Note if K is fixed, this is a polynomial problem because you check

$$\binom{|V|}{K} = \frac{|V|!}{K!\,(|V|-K)!}$$

$\leq |V|^K$ possible subsets!

Both K & $(|V|, |E|)$ are variables.

Take away: Many problem are NP complete ⟹ nobody Knows of efficient algorithm to solve them (do not know if efficient algs are impossible or not!)

Other problems:

Minimal Spanning Tree

Shortest Path

Max flow

have efficient solutions!