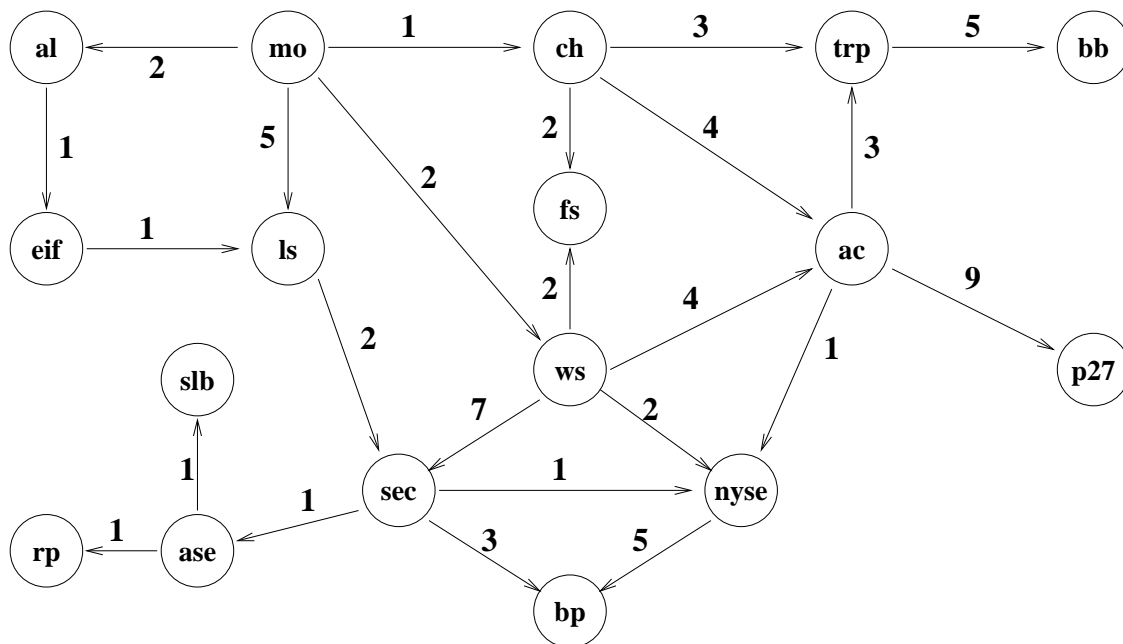


The Bike Courier's Search Problem in Lower Manhattan

CSC384 Supplemental Handout
Courtesy Craig Boutilier

January 29, 2003

The following graph represents *one version* of a map of Lower Manhattan and the Financial District. Our general problem is to find the best route for our bicycle courier from our main office to a specified location. But often she phones in and requests the best route from her current location to another, or sometimes wants to know how to get back to the office. This graph represents the costs of various moves during rush hour. (The key is given below.)



mo: Main Office	ch: City Hall	trp: T.Rowe Price
al: Alley	fs: Fulton Street	bb: Brooklyn Bridge
eif: Ellis Island Ferry	ws: Wall Street	ac: Anderson Consulting
ls: Loeb Securities	sec: Securities Exch. Comm.	p27: Pier 27
slb: Shearson-Lehmann Bros.	bp: Battery Park	nyse: NY Stock Exch.
rp: Rector Park	ase: Amer. Stock Exch.	

Each arc represents a path from one location to another, and the label on each arc represents the cost of traversing that arc. We can represent this graph using a “neighbors” relation `nb` for nodes. If we ignore costs for now, we can represent the graph using the following facts:

```

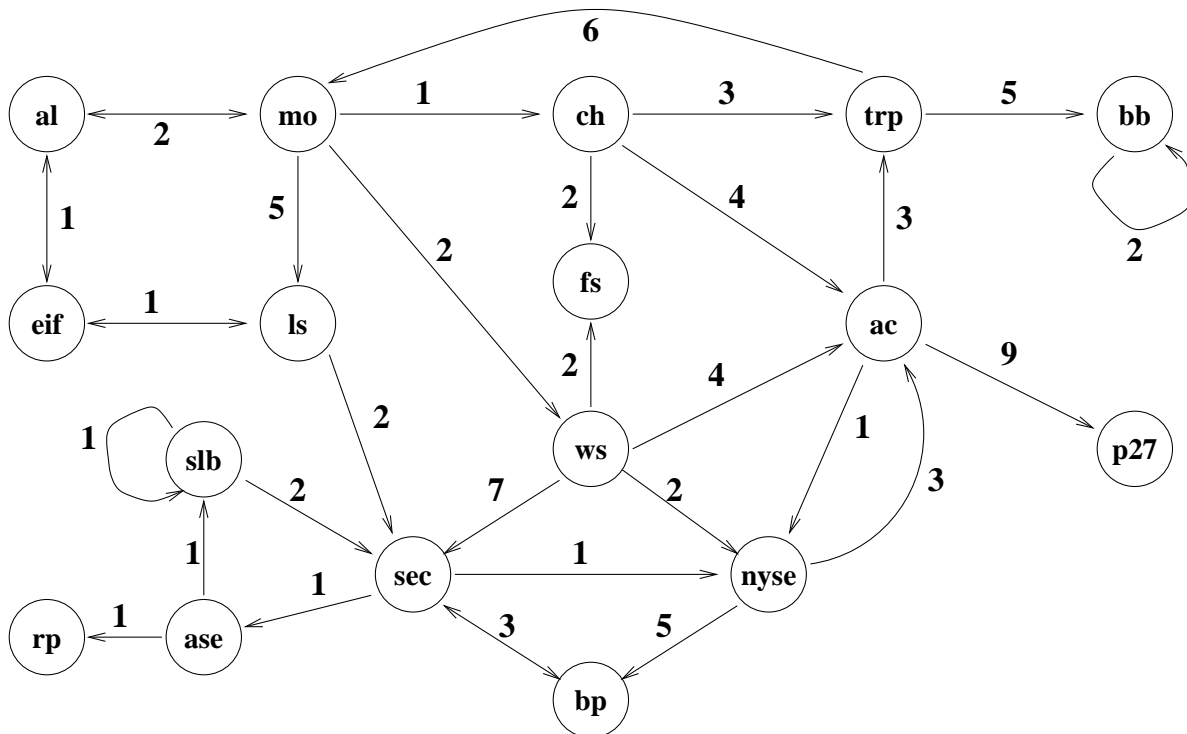
nb(mo, [al, ls, ws, ch]).
nb(sec, [ase, bp, nyse]).
nb(ws, [fs, sec, nyse, ac]).
nb(ac, [trp, nyse, p27]).
nb(ase, [slb, rp]).
nb(ch, [fs, ac, trp]).
nb(sl, [ ]).
nb(fs, [ ]).
nb(eif, [ls]).
nb(al, [eif]).
nb(ls, [sec]).
nb(rp, [ ]).
nb(trp, [bb]).
nb(bb, [ ]).
nb(nyse, [bp]).
nb(p27, [ ]).
nb(bp, [ ]).

```

If we encode arc costs by making each “neighbor” in the neighbor list be a list itself, consisting of the node and the cost of reaching it, we obtain the following:

```
nb(mo, [[al,2], [ls,5], [ws,2], [ch,1]]).      nb(fs, []).
nb(ch, [[fs,2], [ac,4], [trp,3]]).            nb(rp, []).
nb(al, [[eif,2]]).                            nb(trp, [[bb,5]]).
nb(ws, [[fs,2], [sec,7], [nyse,2], [ac,4]]).  nb(p27, []).
nb(ls, [[sec,2]]).                            nb(eif, [[ls,1]]).
nb(ac, [[trp,3], [nyse,1], [p27,9]]).         nb(slb, []).
nb(ase, [[slb,1], [rp,1]]).                  nb(bb, []).
nb(nyse, [[bp,5]]).                          nb(bp, []).
nb(sec, [[ase,1], [bp,3], [nyse,1]]).
```

Here is a revised graph that reflects new routes that open up just after rush hour. Unlike the earlier graph this has cycles. (An arc with a double arrow indicates identical cost in both directions.) It's (cost including) neighbor list is given below (you can easily figure out the cost-less representation).



```
nb(mo, [[al,2], [ls,5], [ws,2], [ch,1]]).      nb(fs, []).
nb(ch, [[fs,2], [ac,4], [trp,3]]).            nb(rp, []).
nb(al, [[eif,2], [mo,2]]).                    nb(trp, [[bb,5], [mo,6]]).
nb(ws, [[fs,2], [sec,7], [nyse,2], [ac,4]]).  nb(p27, []).
nb(ls, [[sec,2], [eif,1], [mo,5]]).           nb(eif, [[ls,1], [al,1]]).
nb(ac, [[trp,3], [nyse,1], [p27,9]]).         nb(slb, [[sec,2], [slb,1]]).
nb(ase, [[slb,1], [rp,1]]).                  nb(bb, [[bb,2]]).
nb(nyse, [[bp,5], [ac,3]]).                  nb(bp, [[sec,3]]).
nb(sec, [[ase,1], [bp,3], [nyse,1]]).
```

Hang on to this handout (or pick it up online). We will use it as a running example for our various search algorithms for the next couple classes.