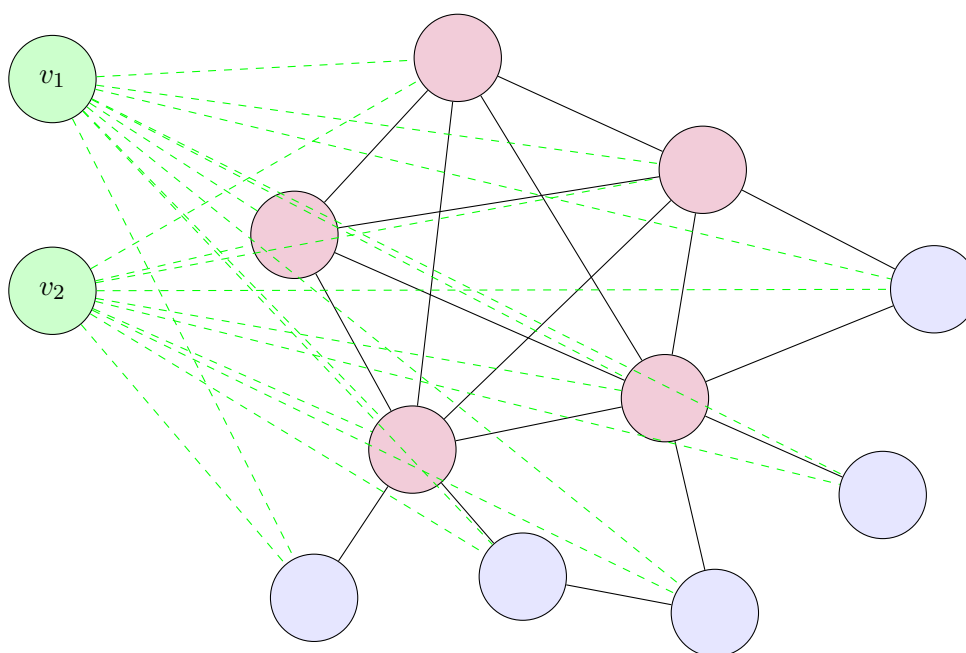# LaTeX Snippet Compilation

April 29, 2014
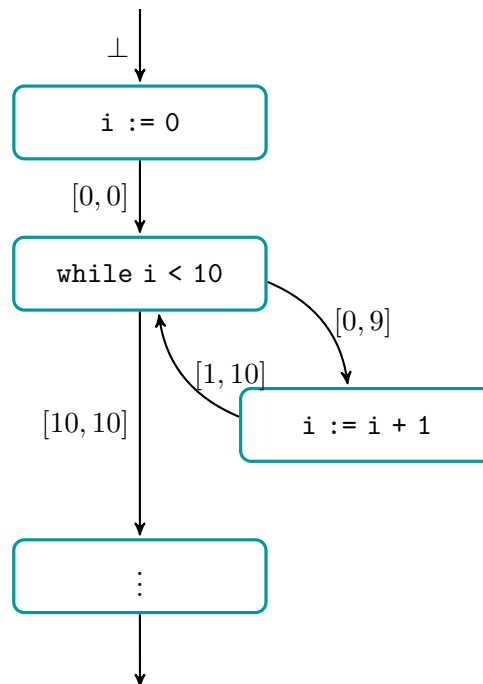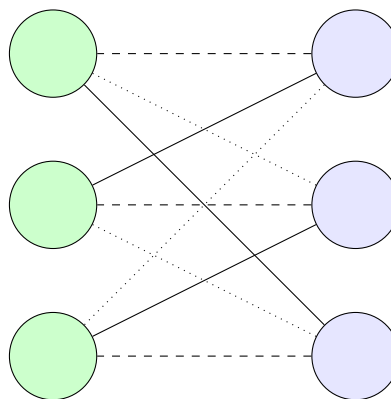
1 TikZ Graphs

Clique

Graph G with clique k_5 highlighted in red and v_1, v_2 connected to every vertex in G. A near clique of size $k + 2$ forms between the red and green colored nodes.

Flow chart style graph

$\perp$

```
i := 0
```

$[0, 0]$

```
while i < 10
```

$[0, 9]$

$[1, 10]$

```
i := i + 1
```

$[10, 10]$

$\vdots$

Bipartite Graphs

Vertical bipartite, Vertical flow

A_1 A_2 A_3 A_4 A_5 A_6

B_1 B_2 B_3 C
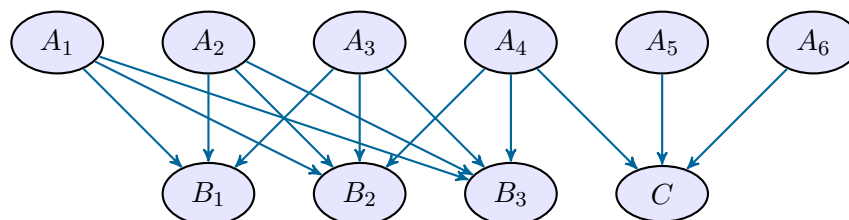
Doubly labeled nodes, Flow Graph

$\Phi(s_i', s_{-i})$

ϕ_i

$\Phi(s)$

ϕ_0

$\Phi(s_{ij}', s_{-i-j})$

ϕ_{ij}

$\Phi(s_j', s_{-j})$

ϕ_j
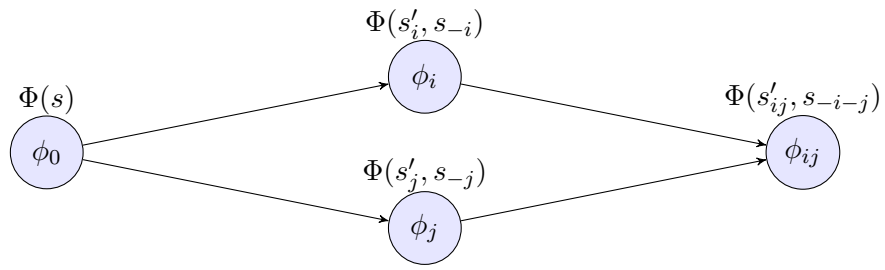
Side by side graphs, bent labeled edges

$c_e(x)$

$c_e'(x)$

Self Loops

40

X

10
10

Y

20

Markets

S_1

0

+1

T_1

B_1

1

B_3

S_2

0

+1

T_2

1

B_2
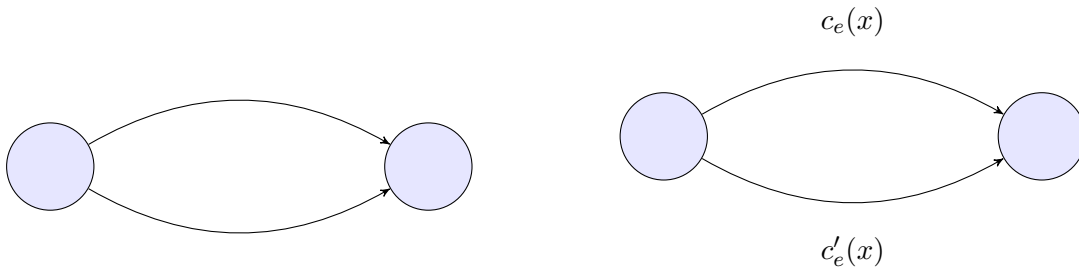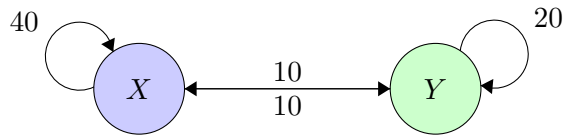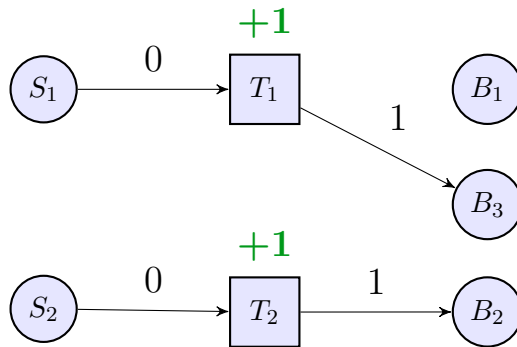
Network

Annotated Graph

$$c_e(s) = \begin{cases} -u_i(s), & x_e = n \\ 0, & \text{otherwise} \end{cases}$$

$$c_e'(s) = \begin{cases} -c_e(s), & x_e = n-1 \\ 0, & \text{otherwise} \end{cases}$$

Player A's Set   Player B's Set
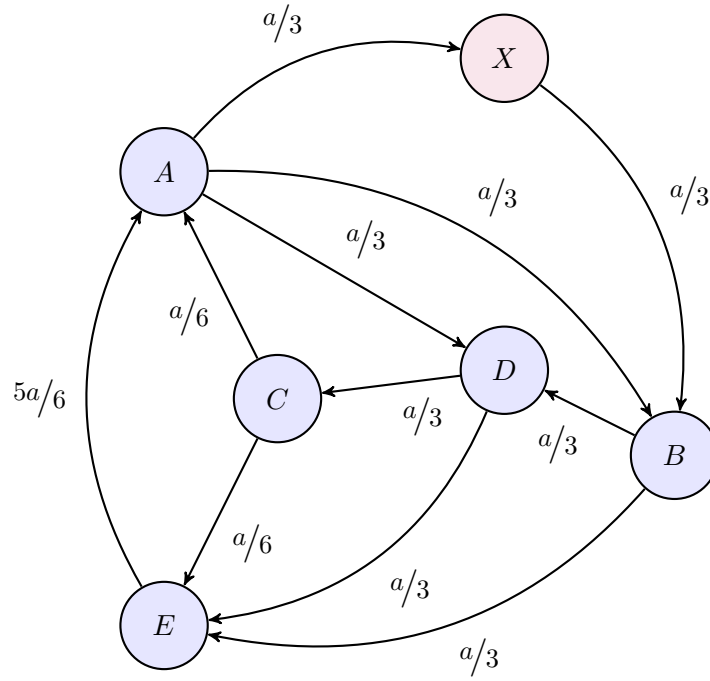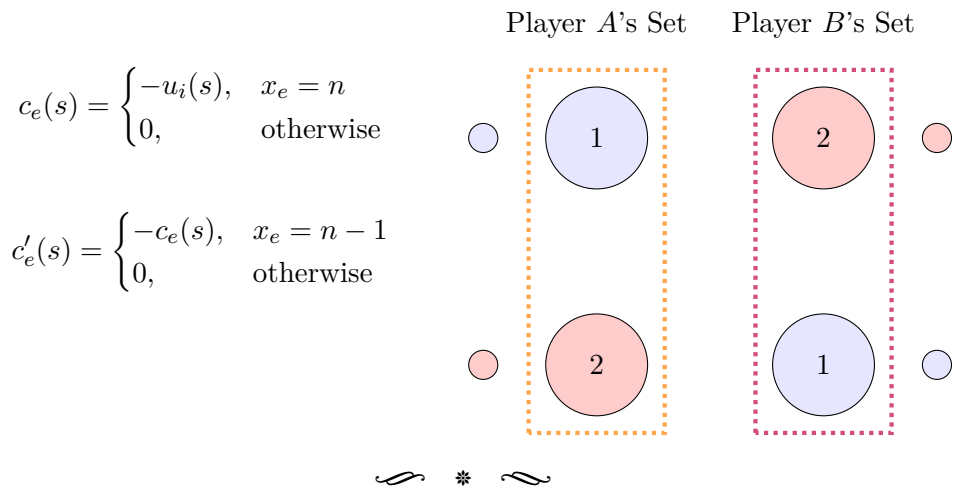
Subgraph
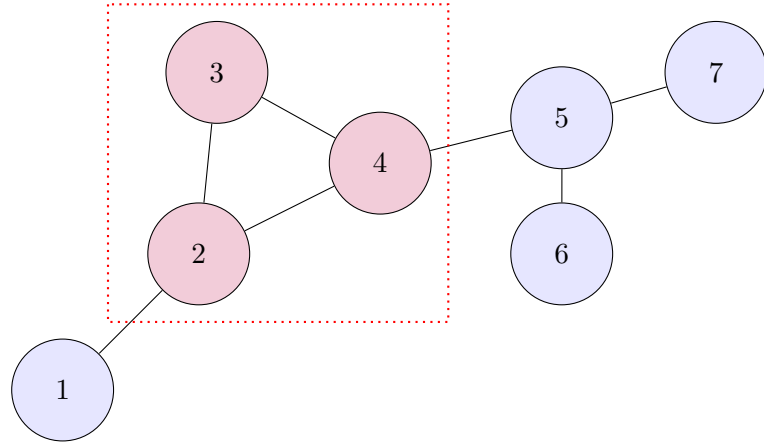
Graph G with clique k_3 highlighted.

Runtime of σ

Add nodes v_1, v_2:	$O(1)$
Connect v_1, v_2 to $v \,\forall v \in V$:	$O(V)$
Total Reduction Runtime:	$O(V)$

2   Matrices

$$M_k = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \qquad M_G = \begin{bmatrix} 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 2 \end{bmatrix}$$

Runtime of σ

Compute M_k:	$O(k^2)$
Compute M_G:	$O(V^2)$
Total Reduction Runtime:	$O(V^2)$

τ **Reduction:** Construct a function τ that takes the output of σ and converts it to a valid solution of the clique problem:

This is a decision problem with only a boolean output. `True` and `False` map to the same values and the reduction is trivial.

Runtime of τ

        Output boolean is equivalent to solution of vertex cover: $O(1)$
        Total Reduction Runtime: $O(1)$

2.1 Efficient Verifier:

Given a solution set S to the submatrix domination problem, test all values of A into $r(\cdot), c(\cdot)$. If every value of $r(\cdot), c(\cdot)$ matches, then the algorithm should have returned `True`, and if not, then `False`.

Runtime of Verifier

        Iterate through m_1 rows and n_1 columns of A: $O(n_1 m_1)$
        Total Runtime: $O(n_1 m_1)$

3 Runtime table

Runtime of σ

        Calculate number of edges: $O(E)$
        Create List L': $O(E)$
        Find M: $O(1)$
        Total Reduction Runtime: $O(E)$

4 Algorithms

1: Initialize $t := 0$
2: Create $g(p, v, time)$ ▷ Returns the current location of Sub
3: **for** $s_i = (p_i, v_i) \in S$ **do**   ▷ Exhaustively try all possible solutions
4:    $time := time + 1$
5:    $location := g(p_i, v_i, time)$
6:    $hit := f(location)$
7:    **if** $hit = 1$ **then** $first\_hit := \{hit, time\}$
8:    **end if**
9: **end for**
10: **for** $s_i = (p_i, v_i) \in S$ **do**   ▷ Find Sub location a second time for linear interpolation
11:    $time := time + 1$
12:    $location := g(p_i, v_i, time)$
13:    $hit := f(location)$

14:     **if** $hit = 1$ **then** $second\_hit := \{hit, time\}$
15:     **end if**
16: **end for**
17: $(p, v) := Interpolate(first\_hit, second\_hit)$ ▷ Linearly interpolate between known points
     return (p, v)

5 Language Semantics

$$\frac{\Psi \,|\, \Theta \,|\, \Delta \,|\, \Gamma \,|\, \vdash s : \hat{\Gamma} \quad \Psi \,|\, \Theta \,|\, \Delta \,|\, \Gamma, \hat{\Gamma} \vdash e : \texttt{Boolean}\langle\rangle}{\Psi \,|\, \Theta \,|\, \Delta \,|\, \Gamma \,|\, \hat{\Gamma} \vdash \mathbf{do}(s : \hat{\Gamma}) \,\mathbf{until}(e)}$$

6 Code and Syntax Highlighting

```java
// Hello.java
import javax.swing.JApplet;
import java.awt.Graphics;

public class Hello extends JApplet {
    public void paintComponent(Graphics g) {
        g.drawString("Hello, world!", 65, 95);
    }
}
```

7 Tables

	<	>	/	=	word
TAG	OPENTAG C CLOSETAG				
C	TAG C, ϵ				word C
EQ					word = word
S		ϵ			EQ S
OPENTAG	<word S>				
CLOSETAG	</ word>				

Diagonal Box

CD \ AB	00	01	11	10
00	$1/8$	0	$1/8$	0
01	0	$1/8$	0	$1/8$
11	0	$1/8$	0	$1/8$
10	$1/8$	0	$1/8$	0

Utility Matrices

A \ B	s_1	s_2	s_3
s_1	$0 \quad \epsilon$	$0 \quad \epsilon$	$0 \quad \epsilon$
s_2	$0 \quad 0$	$0 \quad 0$	$0 \quad 0$
s_3	$0 \quad 0$	$0 \quad 0$	$0 \quad 0$

A \ B	s_1	s_2
s_1	$0 \quad \epsilon$	$0 \quad \epsilon$
s_2	$0 \quad 0$	$0 \quad 0$

E	2	2ˆ3 * 4 + 5
$\rightarrow T_1 E'$	2	2ˆ3 * 4 + 5
$\rightarrow T_2\, T_1' E'$	2	2ˆ3 * 4 + 5
$\rightarrow N\, T_2'T_1' E'$	2	2ˆ3 * 4 + 5
$\rightarrow 2T_2'T_1' E'$	ˆ	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}T_2 T_1' E'$	3	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}N T_2'T_1' E'$	3	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3T_2'T_1' E'$	*	2ˆ3*4 + 5
$\rightarrow 2\text{ˆ}3T_1' E'$	*	2ˆ3*4 + 5
$\rightarrow 2\text{ˆ}3 * T_1 E'$	4	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * T_2 T_1' E'$	4	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * N T_2'T_1' E'$	4	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 T_2'T_1' E'$	+	2ˆ3 * 4+5
$\rightarrow 2\text{ˆ}3 * 4 T_1' E'$	+	2ˆ3 * 4+5
$\rightarrow 2\text{ˆ}3 * 4 E'$	+	2ˆ3 * 4+5
$\rightarrow 2\text{ˆ}3 * 4 + E$	5	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 + T_1 E'$	5	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 + T_2 T_1' E'$	5	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 + N T_2'T_1' E'$	5	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 + 5 T_2'T_1' E'$	\$	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 + 5 T_1' E'$	\$	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 + 5 E'$	\$	2ˆ3 * 4 + 5
$\rightarrow 2\text{ˆ}3 * 4 + 5$	\$	2ˆ3 * 4 + 5

8 Trees

$((3 + 5) * 2)$

```
        *
       / \
      +   2
     / \
    3   5
```

$(3 + (5 * 2))$

```
      +
     / \
    3   *
       / \
      5   2
```

9 Piecewise Functions

$$f(n) = \begin{cases} n/2, & \text{if } n \text{ is even} \\ 3n + 1, & \text{if } n \text{ is odd} \end{cases}$$

10 Chinese and Foreign Characters

方启明

11   Other Decorations