Identifying *Linchpin Vertices* that Cause Large Dependence Clusters

Dave Binkley
Loyola University Maryland

Mark Harman
Crest Centre, Kings College London

What’s Coming

- Dependence Defined
- Dependence Clusters
- Financial Motivation
- The MSG
- Finding Dependence Cluster Causes

Dependence

I’ll go if you go

Dependence Cluster

I'll go if you go

I'll go if you go

I'll go if you go

Thus you get
all or nothing!

Dependence

- main()
- {
- a = 42;

Data Dependence (definition – use)

- if a > 10

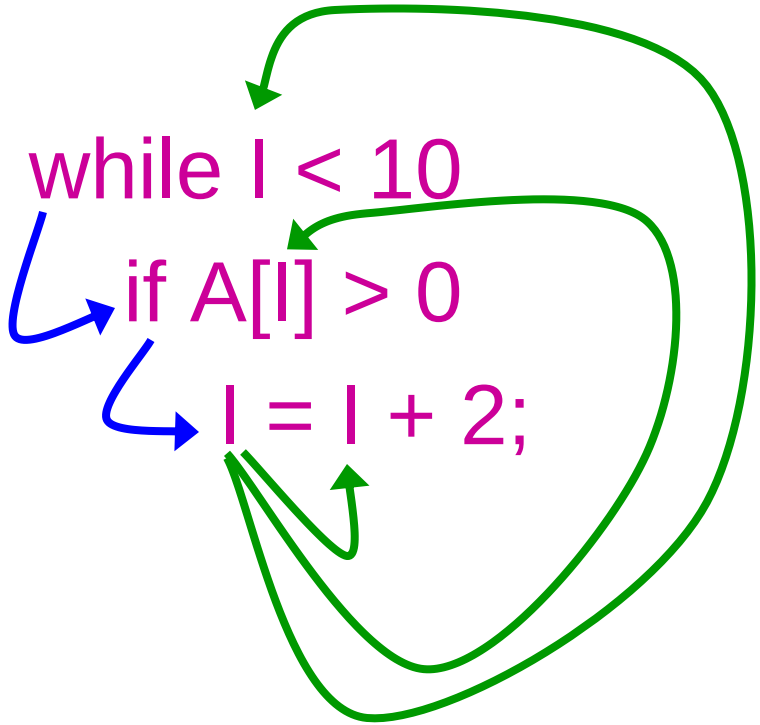
Control Dependence

- b = a / 2;

}

Dependence Cluster

- main()
- {
- while I < 10
- if A[I] > 0
- I = I + 2;
- }



Dependence Cluster

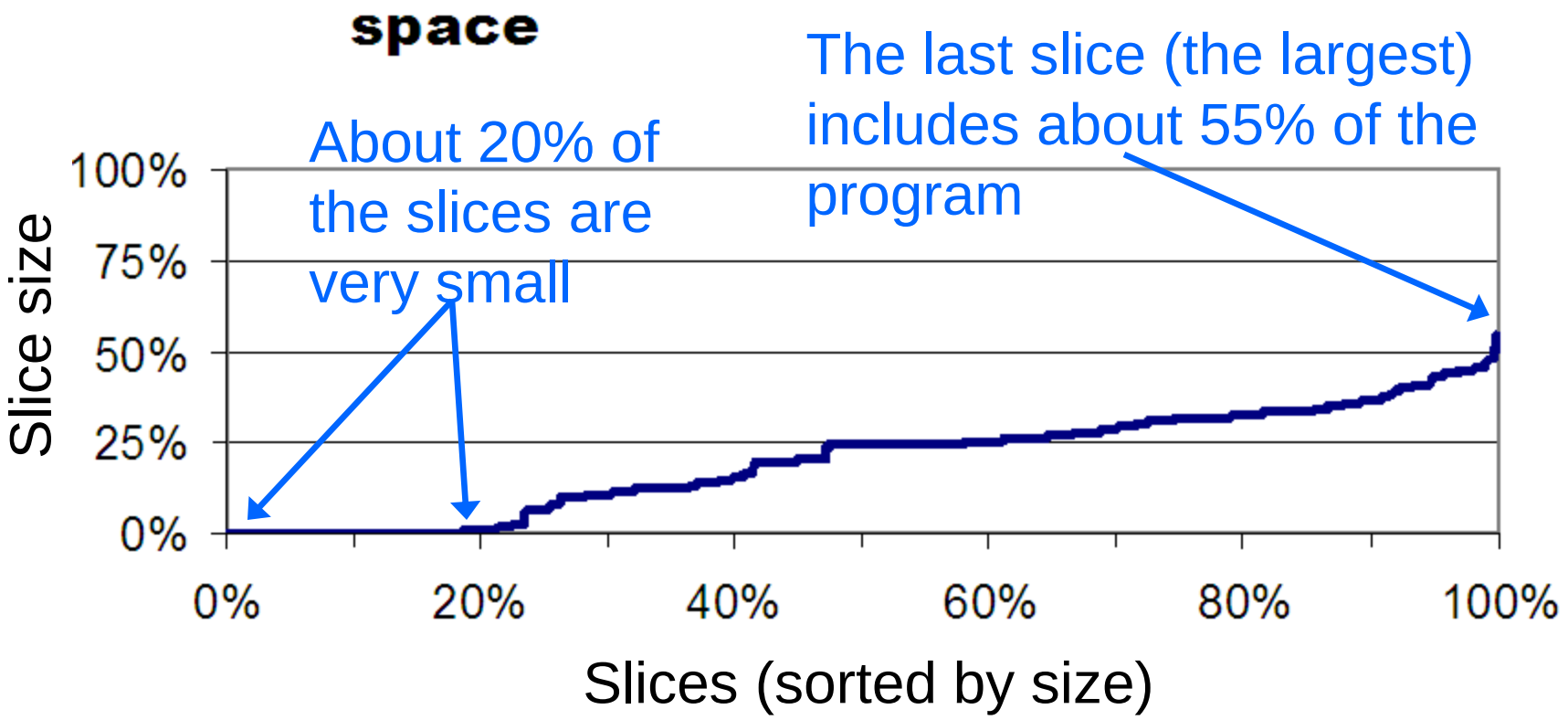
For statement ‘s’ of program ‘P’

Cluster(s) = { t $\in$ P | slice(t) = slice(s) }

slice(t) approximated using sizeof(slice(t))

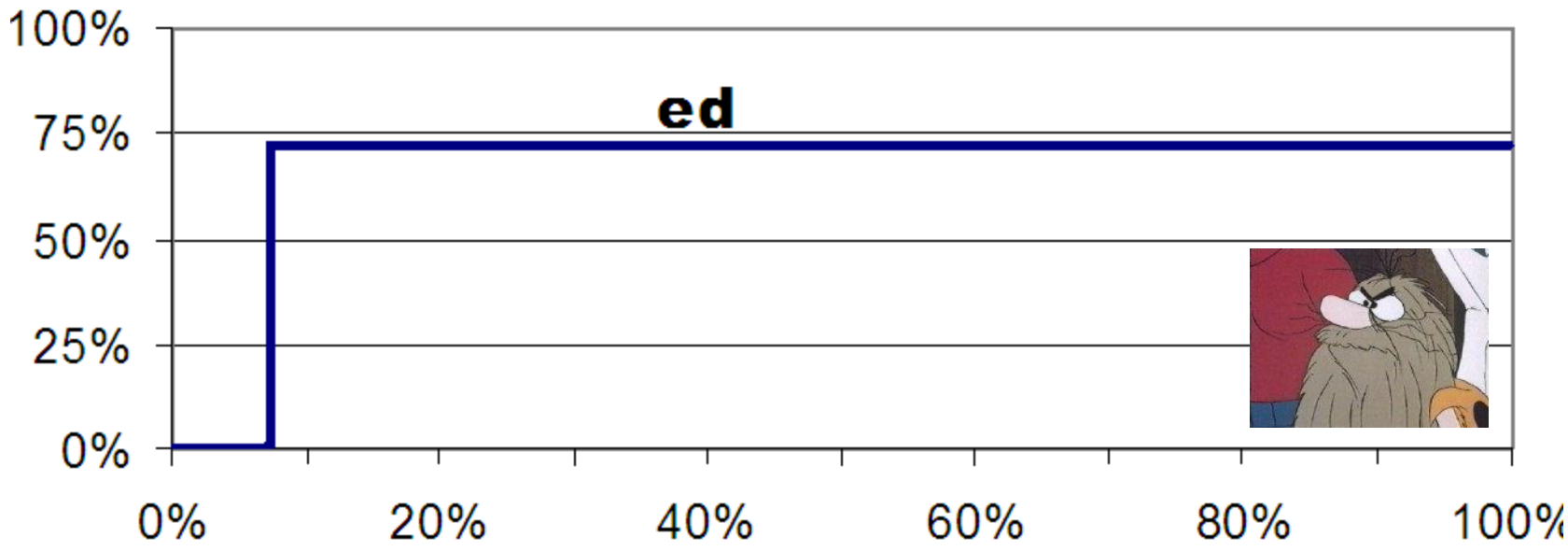
approximation is 99% accurate

The Approximation Yields an Interesting Visualisation ***the MSG*** (the *Monotone Slice-size Graph*)



An MSG

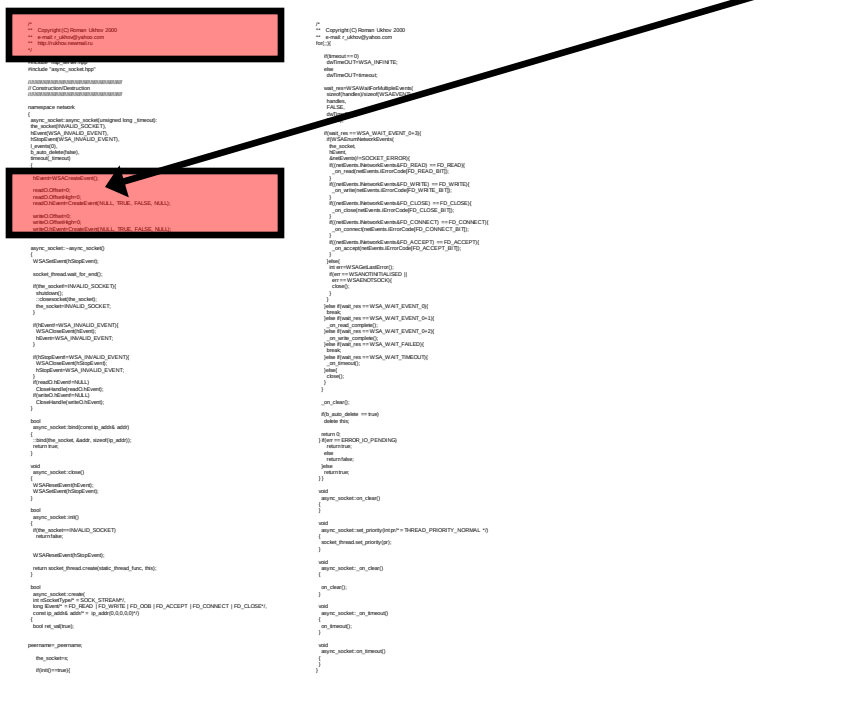
“Programs Resist Transformation” - Mike



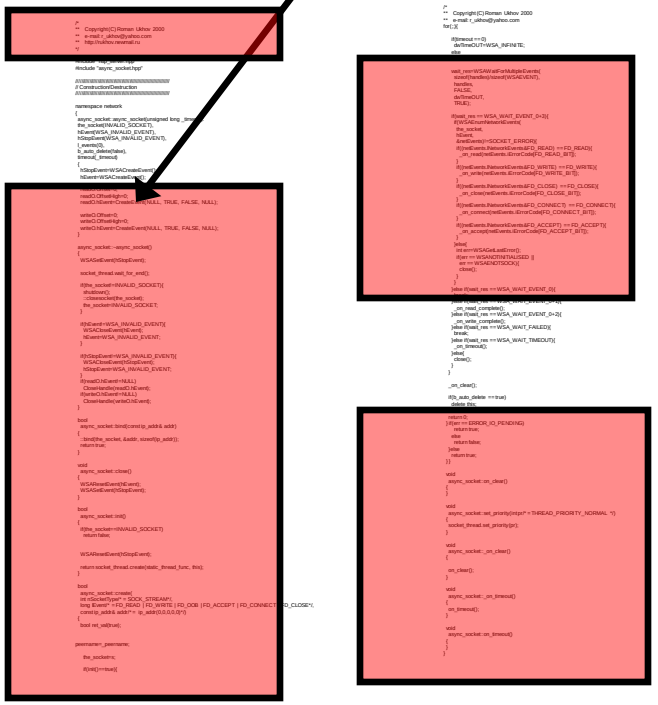
Impact of Dependence Clusters

Consider making a change to Line 42

Program Without Cluster

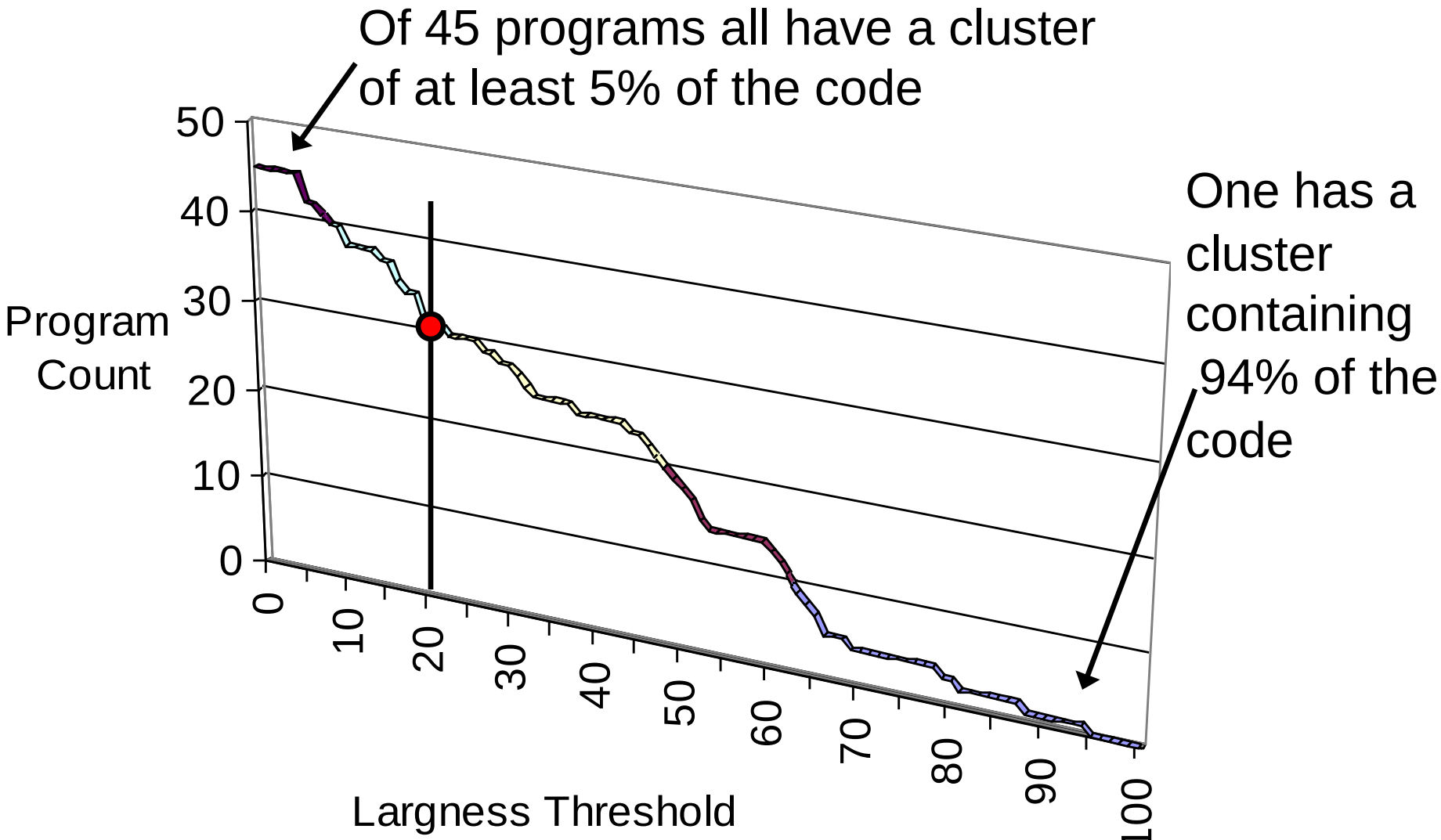


Program With Cluster

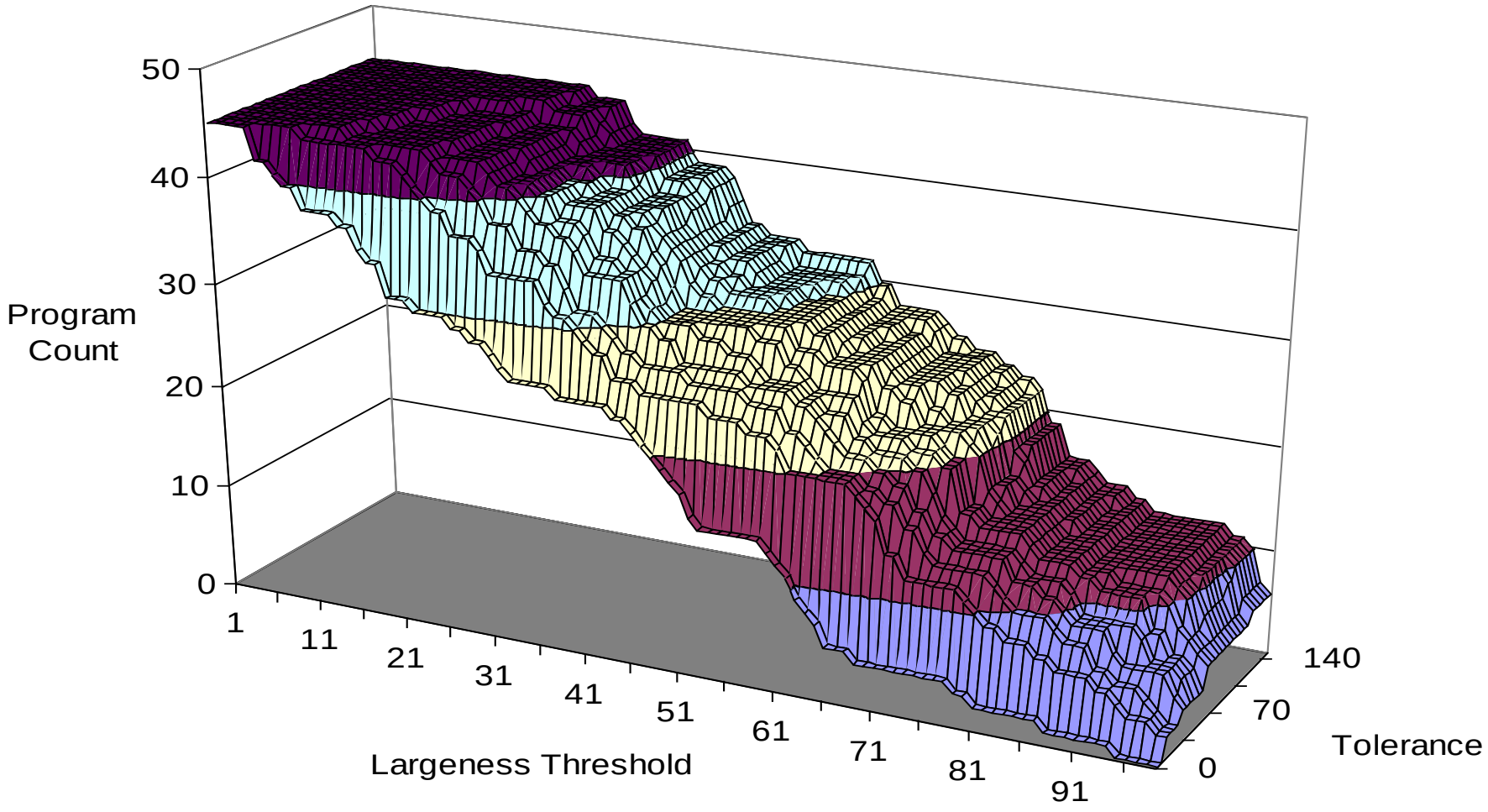


One example does make a problem!

Alas, programs with clusters larger than *Threshold*



Tolerance for Slightly Different Sizes



OK They exist (and They are bad) but, *Can Causes be Identified*?

- Yes!
  - By Hand – bit tedious
  - (Semi) Automatically
    - Vertices and Edges (Statements and Dependences)
    - Global Variables

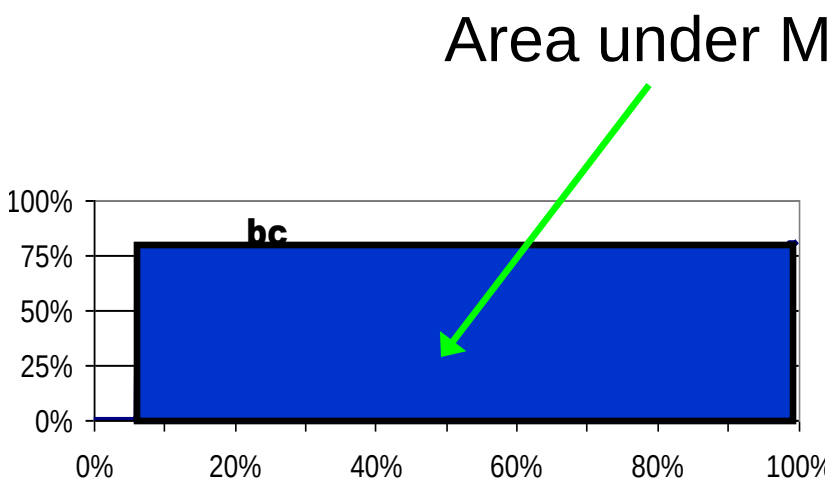
The Automated Vertex Technique

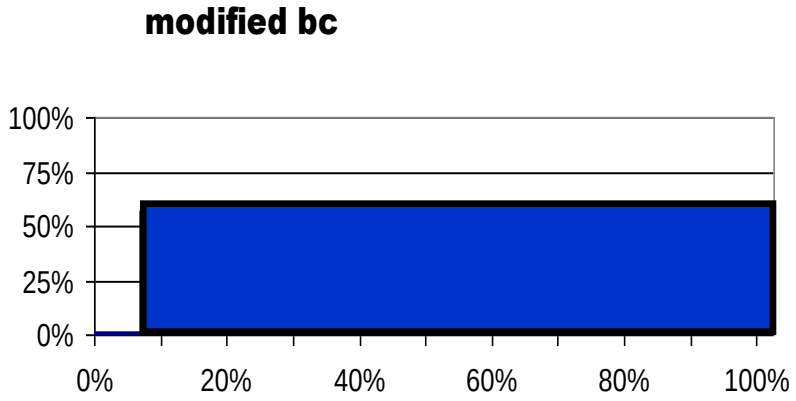
Ignore the dependences associated with each SDG *vertex* then rebuild MSG

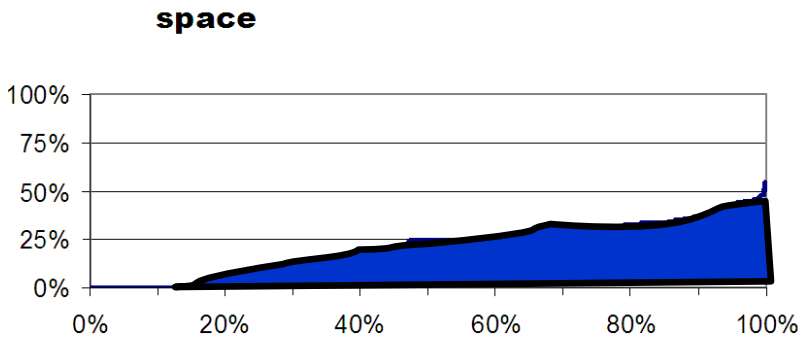
- a very small change to the “program”

Consider MSG area reduction

Measuring Effect of a *Single* Vertex (or edge)



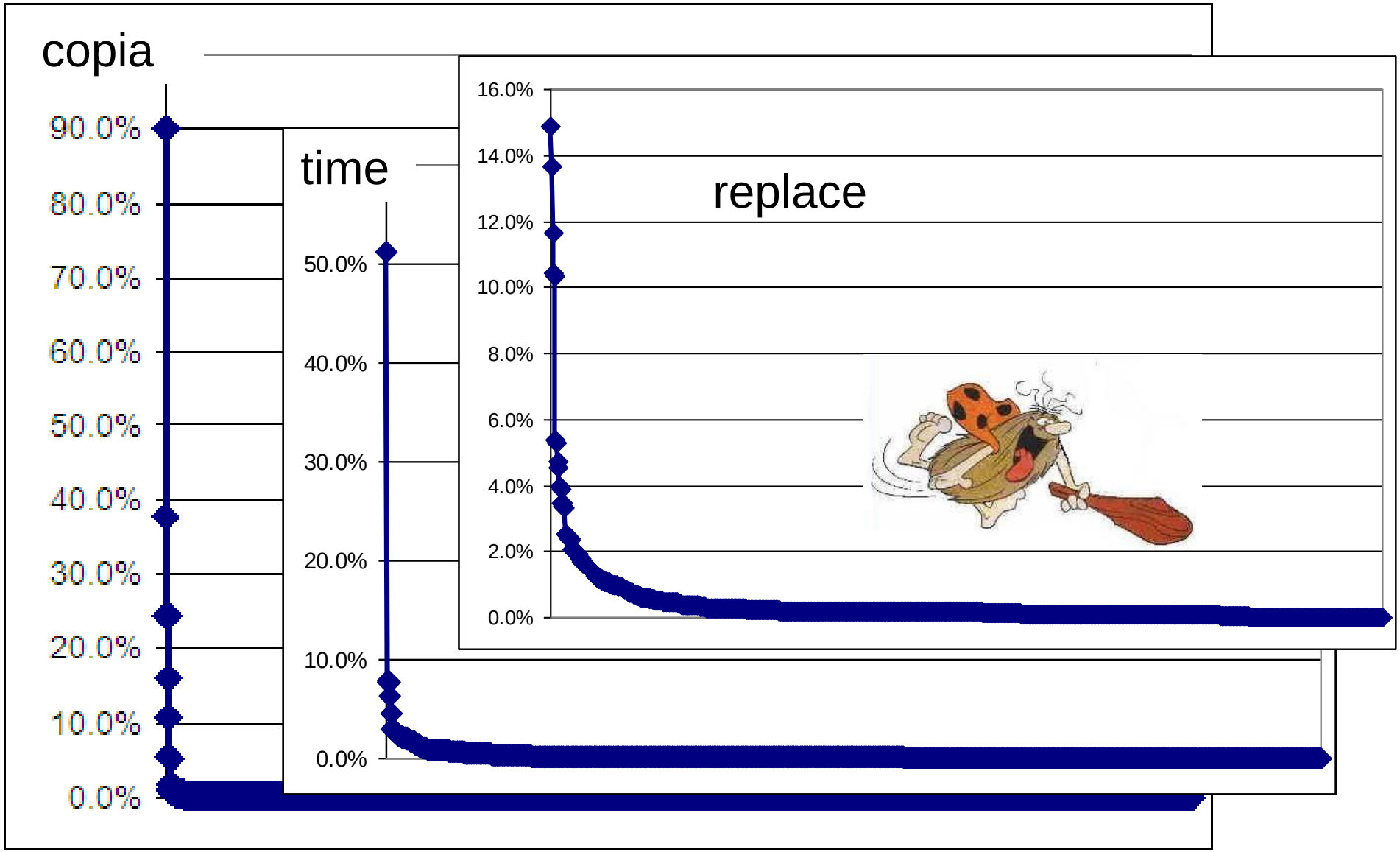




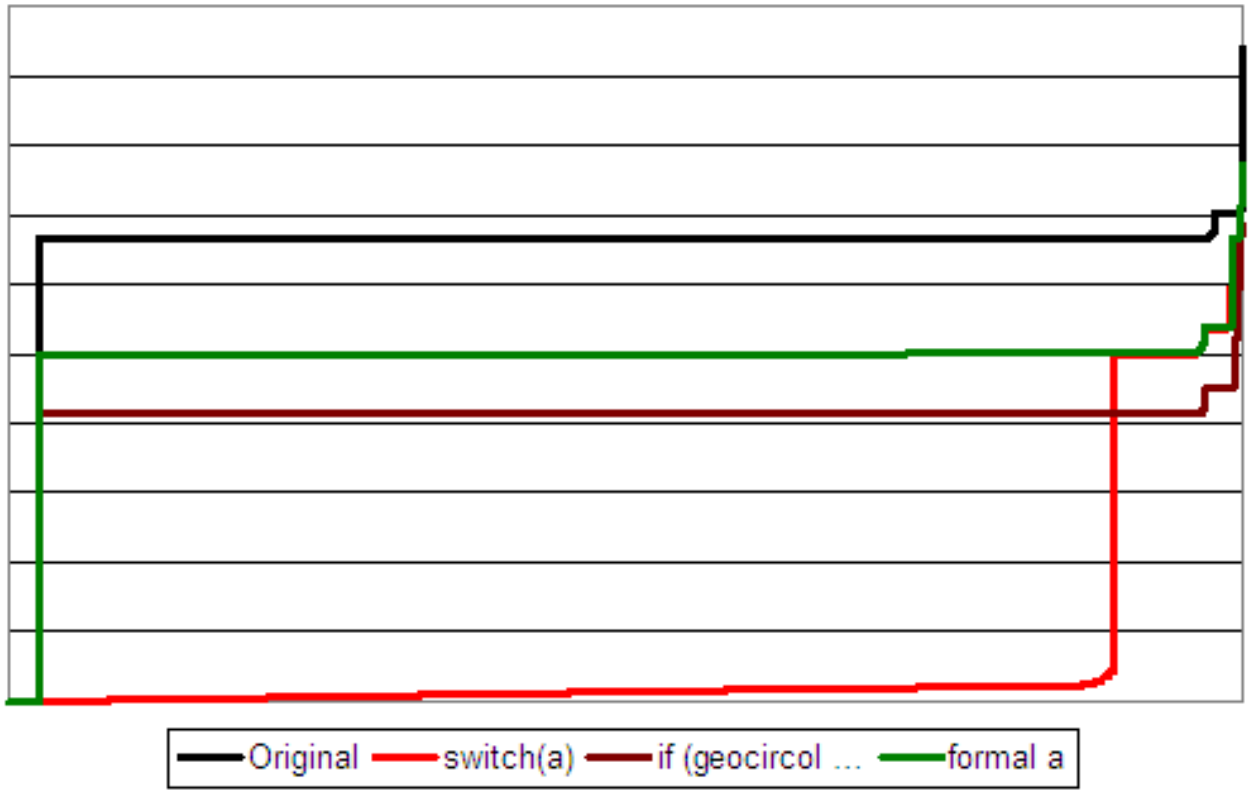
Ignoring Vertex Dependence

slices	program	reduc-tion	vertex type	source
4686	copia	**89.96%**	control-point	switch (a)
1044	time-1.7	**51.23%**	control-point	switch (*++fmt)
1077	conversion	**27.04%**	control-point	switch (pick_op)
747	driver	**26.45%**	control-point	switch(choice)
585	sudoku	**22.87%**	control-point	while(!check_completed())
11277	space	**7.90%**	control-point	if (error != 0)
9556	gnubg-0.0	**7.32%**	indirect-call	pc->pf()
3909	barcode	**5.95%**	indirect-call	cptr->encode()
12492	EPWIC-1	**5.79%**	control-point	while(state != DONE)
10151	byacc	**1.54%**	expression	k = keyword()

Key Vertices?



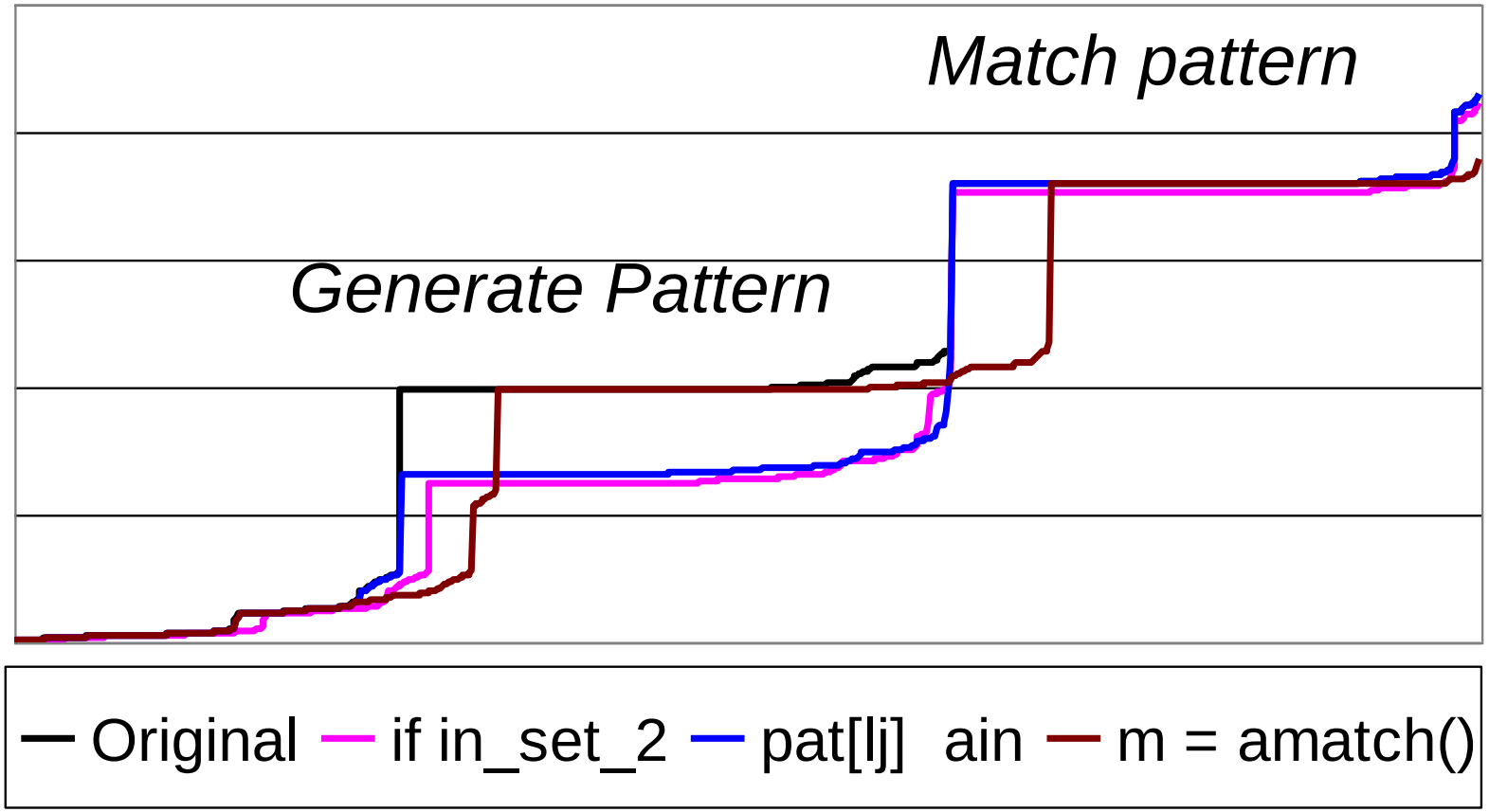
MSGs for Copia's Top 3



Function “Next-state” from Copia

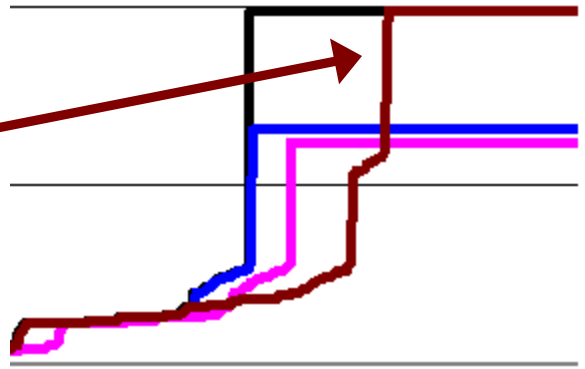
```
 void  seleziona(int a)
{
   switch (a) {
      case 0: grid(); break;
      case 1: hex(); break;
     …
```

Sample MSGs for Replace

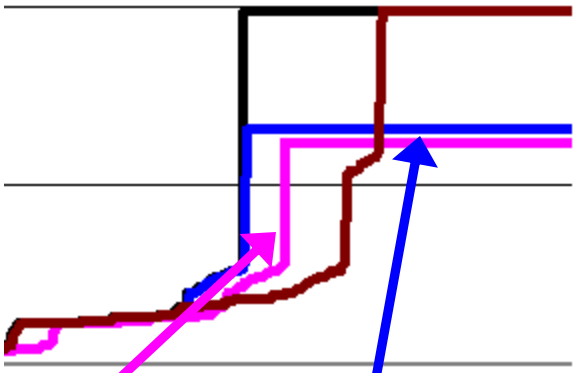


Ignoring of “m = amatch(…)”

```
while ((lin[i] != ENDSTR))
{
    m = amatch(lin, i, pat, 0);
    if ((m >= 0) && (lastm != m))
        putsub(lin, i, m, sub);
        lastm = m;
    }
    if ((m == -1) || (m == i)) {
        fputc(lin[i],stdout);
        i = i + 1;
    } else
        i = m;
}
```

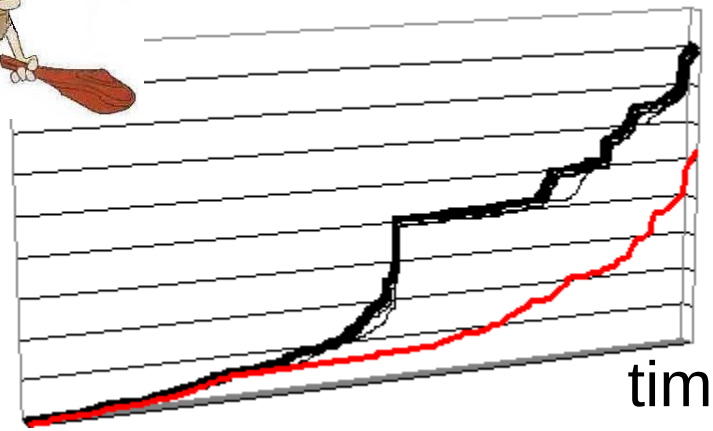


Ignoring of “if” and actual “pat[lj]”



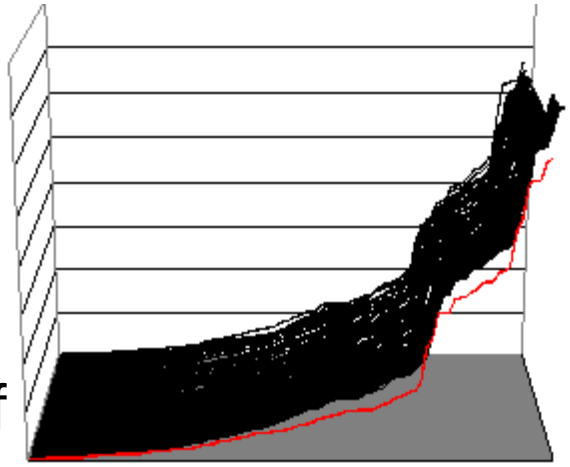
```
else if ((arg[i] == CLOSURE) && (i > start))
{
     lj = lastj;
     if (in_set_2(pat[lj]))
          done = true;
     else
          stclose(pat, &j, lastj);
}
else
```

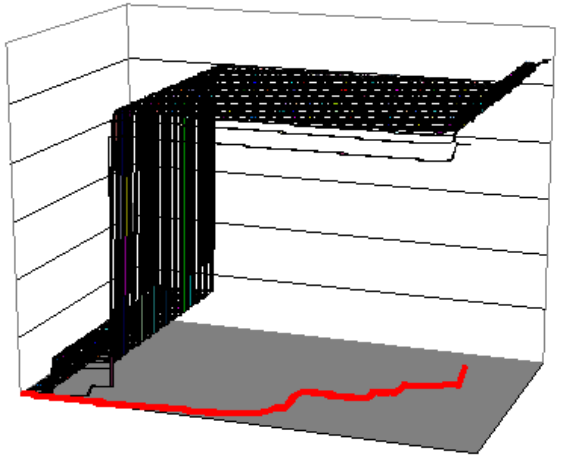
“Future work” – Globals



time

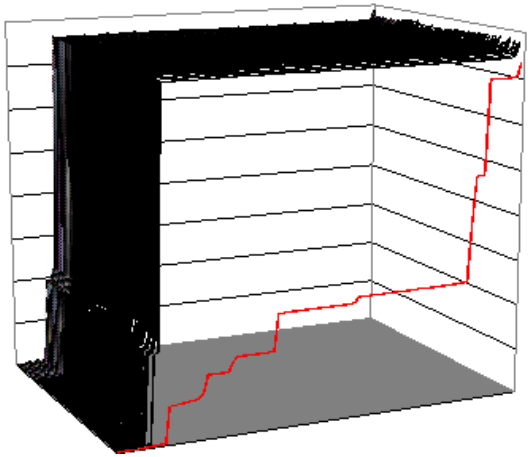
wdiff





ctags

barcode



Summary

- Dependence Clusters Exist
- They impact all (dependence based) static analysis
- Features as small as an individual vertex (even an edge) can play a key role in holding a cluster together

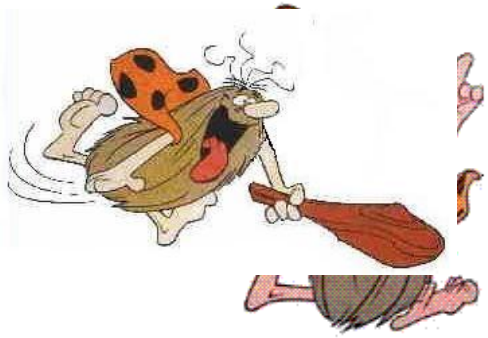
Thanks!

Questions ?

ACluB Web Page

www.dcs.kcl.ac.uk/staff/mark/aclub

Controversy



You talk way to fast

You need to consider the impact of dependence clusters in your work.