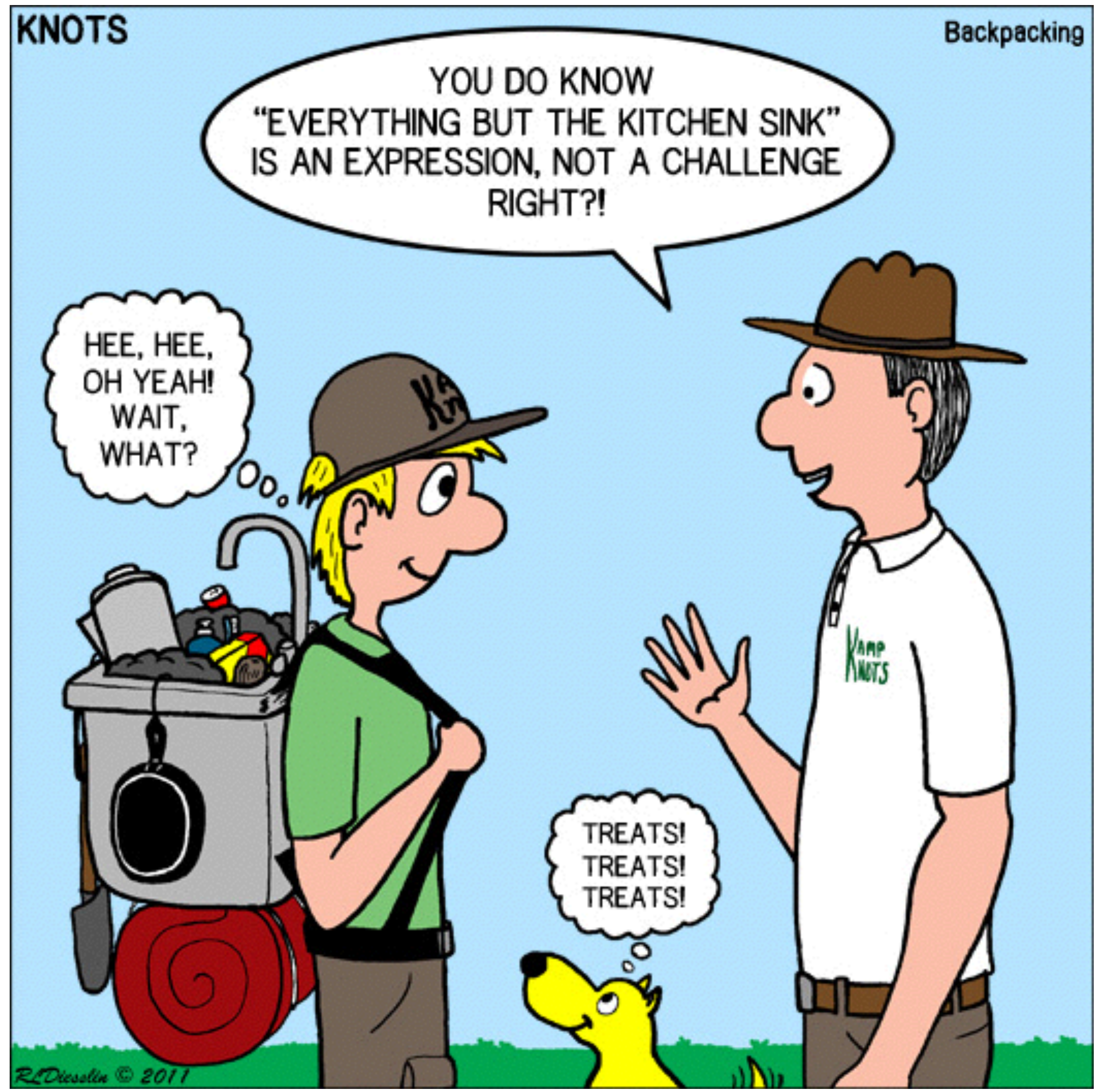
Removing the Kitchen Sink from Software

Jason Landsborough
Stephen Harding
Sunny Fugate

Motivation

- Large feature-rich programs
- Kitchen sink



Used with permission from Rich Diesslin, www.the-cartoonist.com

Why Thin

Pros:

- **Security: Thinned programs do not contain the vulnerabilities present in code which was removed.**
- Size: Thinned programs might be smaller and have performance gains over non-thinned code.
- Validity: Thinned programs are likely to be simpler and easier to validate, requiring fewer test cases.
- Optimality: Thinned programs can be better optimized for the specific machine, removing code for cross- compatibility.

Cons:

- **Security: Thinned programs might not contain various security checks not normally raised in execution.**
- Size: Thinned programs will fail or require additional time to load removed code segments.
- Validity: Thinned programs might require the same number of tests to validate that unwanted features are indeed removed.
- Optimality: Thinned programs may not run at all on other machines.

What to Remove

Two basic classes:

1. Undesirable features
    - Ex: Heartbeat in OpenSSL
2. Unused features
    - Cold code

Image source: http://heartbleed.com/heartbleed.png

Manually Remove Features

- Identify feature

```
080484c4 <fun3>:
 80484c4:       55                      push   %ebp
 80484c5:       89 e5                   mov    %esp,%ebp
 80484c7:       83 ec 18                sub    $0x18,%esp
 80484ca:       c7 04 24 06 86 04 08    movl   $0x8048606,(%esp)
 80484d1:       e8 9a fe ff ff          call   8048370 <puts@plt>
 80484d6:       c9                      leave
 80484d7:       c3                      ret
```

  1. Overwrite with NOPs

  2. Redirect function call

```
00004a0: ec18 c704 24e0 8504 08e8 c2fe ffff c9c3  ....$...........
00004b0: 5589 e583 ec18 c704 24f3 8504 08e8 aefe  U.......$.......
00004c0: ffff c9c3 5589 e583 ec18 c704 2406 8604  ....U.......$...
00004d0: 08e8 9afe ffff c9c3 5589 e583 ec18 c704  ........U.......
                                            77,19          15%
```

- Issue: Very time consuming

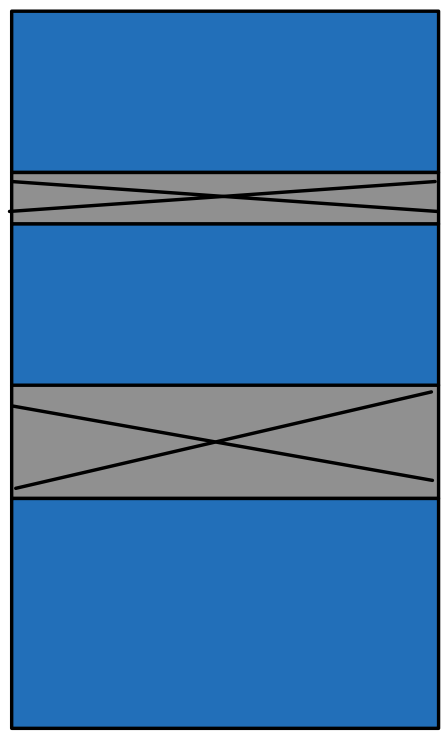
```
00004a0: ec18 c704 24e0 8504 08e8 c2fe ffff c9c3  ....$...........
00004b0: 5589 e583 ec18 c704 24f3 8504 08e8 aefe  U.......$.......
00004c0: ffff c9c3 9090 9090 9090 9090 9090 9090  ................
00004d0: 9090 9090 9090 90c3 5589 e583 ec18 c704  ........U.......
00004e0: 2419 8604 08e8 86fe ffff c9c3 5589 e583  $...........U...
```

Trace-based Thinning

- Dynamic trace of program execution
- diStorm - Trace tool
  - DIFT by Jeff Knockel and Antonio Espinoza at University of New Mexico

Prototype Overview

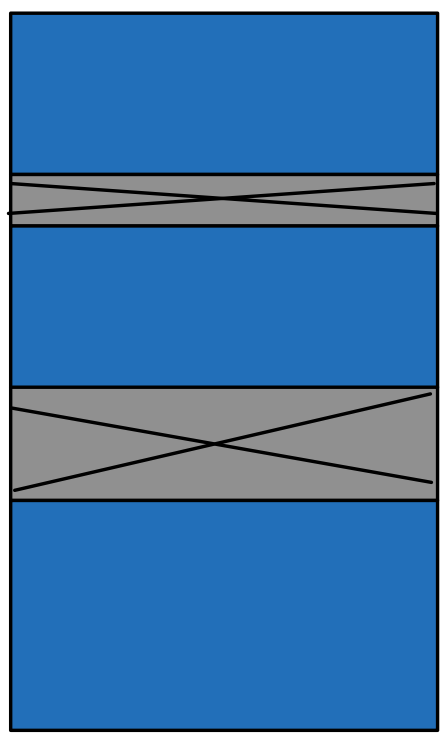
Trace and remove
unused



- Only keeps used instructions
- May remove too much - Riskier

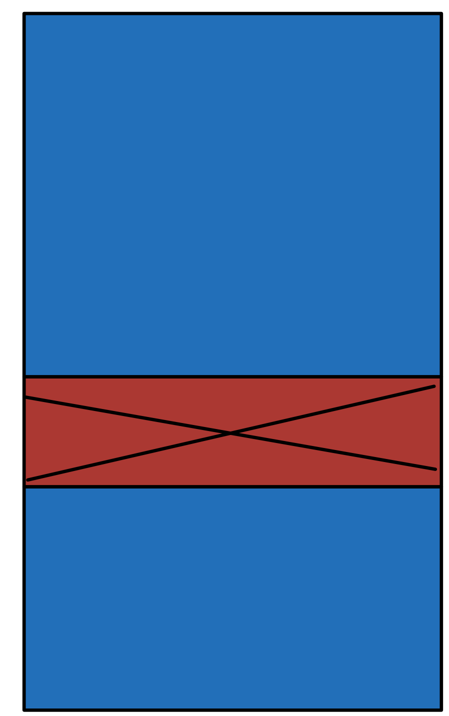
Prototype Overview

Trace and remove unused



- Only keeps used instructions
- May remove too much - Riskier

Trace and remove unwanted



- Only removes unwanted instructions
- May not remove enough - “Safer”

Unused


80484cf: ff d0 call *%eax
80484d1: c9 leave
80484d2: e9 79 ff ff ff jmp 80484
80484d7: e9 74 ff ff ff jmp 80484
080484dc <fun1>:
80484dc: 55 push %ebp
80484dd: 89 e5 mov %esp,
80484df: 83 ec 18 sub $0x18
80484e2: c7 04 24 50 86 04 08 movl $0x80
80484e9: e8 b2 fe ff ff call 80483
80484ee: c9 leave
80484ef: c3 ret
080484f0 <fun2>:
80484f0: 55 push %ebp
80484f1: 89 e5 mov %esp,
80484f3: 83 ec 18 sub $0x18
80484f6: c7 04 24 63 86 04 08 movl $0x80
80484fd: e8 9e fe ff ff call 80483
8048502: c9 leave
8048503: c3 ret
08048504 <fun3>:
8048504: 55 push %ebp
8048505: 89 e5 mov %esp,
8048507: 83 ec 18 sub $0x18
804850a: c7 04 24 76 86 04 08 movl $0x80
8048511: e8 8a fe ff ff call 80483
8048516: c9 leave
8048517: c3 ret
08048518 <fundef>:
8048518: 55 push %ebp
8048519: 89 e5 mov %esp,
804851b: 83 ec 18 sub $0x18
804851e: c7 04 24 89 86 04 08 movl $0x80
8048525: e8 76 fe ff ff call 80483
80484ce: 90 nop
80484cf: 90 nop
80484d0: 90 nop
80484d1: 90 nop
80484d2: 90 nop
80484d3: 90 nop
80484d4: 90 nop
80484d5: 90 nop
80484d6: 90 nop
80484d7: e9 74 ff ff ff jmp 80484
080484dc <fun1>:
80484dc: 55 push %ebp
80484dd: 89 e5 mov %esp,
80484df: 83 ec 18 sub $0x18
80484e2: c7 04 24 50 86 04 08 movl $0x80
80484e9: e8 b2 fe ff ff call 80483
80484ee: c9 leave
80484ef: c3 ret
080484f0 <fun2>:
80484f0: 55 push %ebp
80484f1: 89 e5 mov %esp,
80484f3: 83 ec 18 sub $0x18
80484f6: c7 04 24 63 86 04 08 movl $0x80
80484fd: e8 9e fe ff ff call 80483
8048502: c9 leave
8048503: c3 ret
08048504 <fun3>:
8048504: 90 nop
8048505: 90 nop
8048506: 90 nop
8048507: 90 nop
8048508: 90 nop
8048509: 90 nop
804850a: 90 nop
804850b: 90 nop
804850c: 90 nop
804850d: 90 nop
804850e: 90 nop
804850f: 90 nop
8048510: 90 nop
8048511: 90 nop
8048512: 90 nop
8048513: 90 nop
8048514: 90 nop
8048515: 90 nop
8048516: 90 nop
8048517: 90 nop
08048518 <fundef>:
8048518: 55 push %ebp
8048519: 89 e5 mov %esp,
804851b: 83 ec 18 sub $0x18
804851e: c7 04 24 89 86 04 08 movl $0x80
8048525: e8 76 fe ff ff call 80483
Changes: 15

Unwanted

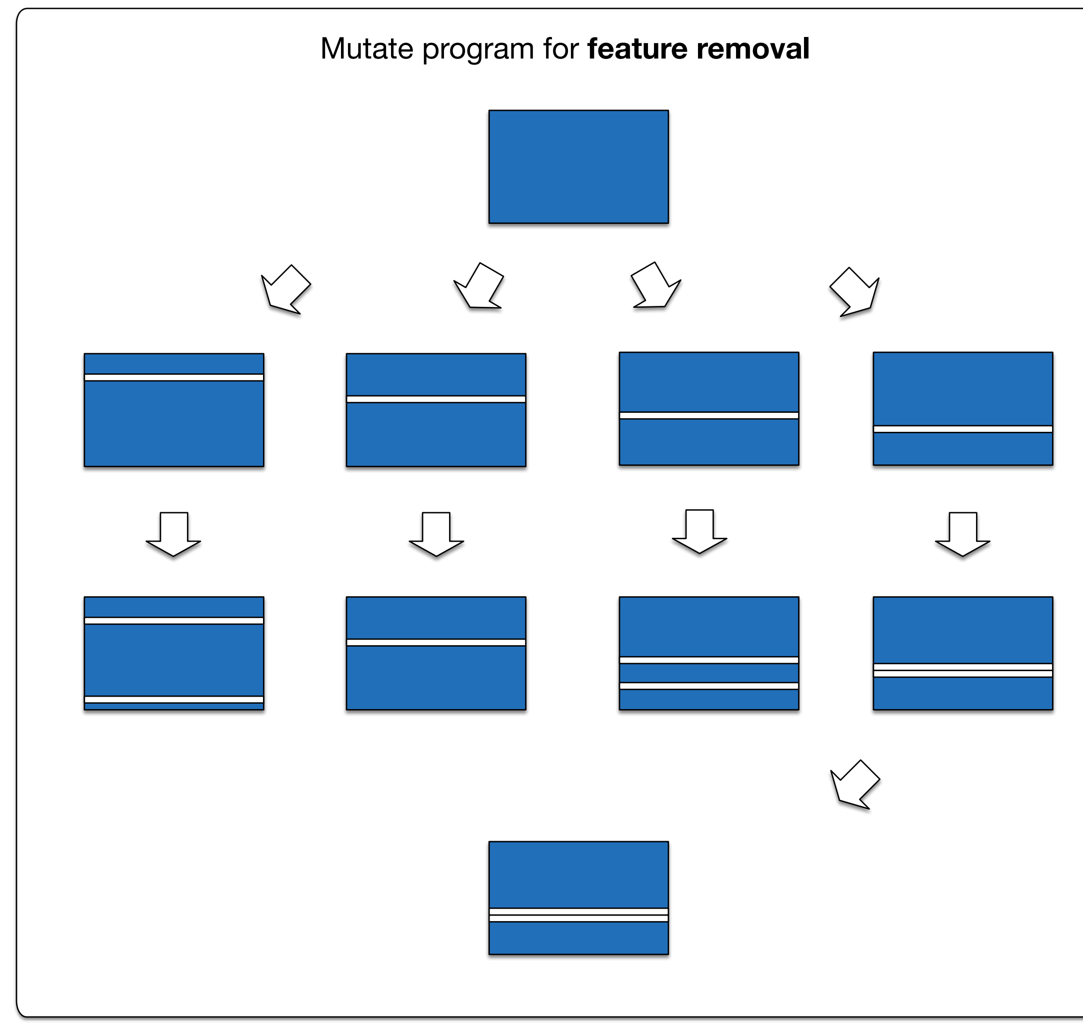
```
 80484be:       85 c0                   test   %eax,
 80484c0:       74 15                   je     80484
 80484c2:       55                      push   %ebp
 80484c3:       89 e5                   mov    %esp,
 80484c5:       83 ec 18                sub    $0x18
 80484c8:       c7 04 24 d8 97 04 08    movl   $0x80
 80484cf:       ff d0                   call   *%eax
 80484d1:       c9                      leave
 80484d2:       e9 79 ff ff ff          jmp    80484
 80484d7:       e9 74 ff ff ff          jmp    80484

080484dc <fun1>:
 80484dc:       55                      push   %ebp
 80484dd:       89 e5                   mov    %esp,
 80484df:       83 ec 18                sub    $0x18
 80484e2:       c7 04 24 50 86 04 08    movl   $0x80
 80484e9:       e8 b2 fe ff ff          call   80483
 80484ee:       c9                      leave
 80484ef:       c3                      ret

080484f0 <fun2>:
 80484f0:       55                      push   %ebp
 80484f1:       89 e5                   mov    %esp,
 80484f3:       83 ec 18                sub    $0x18
 80484f6:       c7 04 24 63 86 04 08    movl   $0x80
 80484fd:       e8 9e fe ff ff          call   80483
 8048502:       c9                      leave
 8048503:       c3                      ret

08048504 <fun3>:
 8048504:       55                      push   %ebp
 8048505:       89 e5                   mov    %esp,
 8048507:       83 ec 18                sub    $0x18
 804850a:       c7 04 24 76 86 04 08    movl   $0x80
 8048511:       e8 8a fe ff ff          call   80483
 8048516:       c9                      leave
 8048517:       c3                      ret

08048518 <fundef>:
 8048518:       55                      push   %ebp
 8048519:       89 e5                   mov    %esp,
 804851b:       83 ec 18                sub    $0x18
 804851e:       c7 04 24 89 86 04 08    movl   $0x80
 8048525:       e8 76 fe ff ff          call   80483
 804852a:       c9                      leave
```

```
 80484be:       85 c0                   test   %eax,
 80484c0:       74 15                   je     80484
 80484c2:       55                      push   %ebp
 80484c3:       89 e5                   mov    %esp,
 80484c5:       83 ec 18                sub    $0x18
 80484c8:       c7 04 24 d8 97 04 08    movl   $0x80
 80484cf:       ff d0                   call   *%eax
 80484d1:       c9                      leave
 80484d2:       e9 79 ff ff ff          jmp    80484
 80484d7:       e9 74 ff ff ff          jmp    80484

080484dc <fun1>:
 80484dc:       55                      push   %ebp
 80484dd:       89 e5                   mov    %esp,
 80484df:       83 ec 18                sub    $0x18
 80484e2:       c7 04 24 50 86 04 08    movl   $0x80
 80484e9:       e8 b2 fe ff ff          call   80483
 80484ee:       c9                      leave
 80484ef:       c3                      ret

080484f0 <fun2>:
 80484f0:       55                      push   %ebp
 80484f1:       89 e5                   mov    %esp,
 80484f3:       83 ec 18                sub    $0x18
 80484f6:       c7 04 24 63 86 04 08    movl   $0x80
 80484fd:       e8 9e fe ff ff          call   80483
 8048502:       c9                      leave
 8048503:       c3                      ret

08048504 <fun3>:
 8048504:       90                      nop
 8048505:       90                      nop
 8048506:       90                      nop
 8048507:       90                      nop
 8048508:       90                      nop
 8048509:       90                      nop
 804850a:       90                      nop
 804850b:       90                      nop
 804850c:       90                      nop
 804850d:       90                      nop
 804850e:       90                      nop
 804850f:       90                      nop
 8048510:       90                      nop
 8048511:       90                      nop
 8048512:       90                      nop
 8048513:       90                      nop
 8048514:       90                      nop
 8048515:       90                      nop
 8048516:       90                      nop
 8048517:       90                      nop

08048518 <fundef>:
 8048518:       55                      push   %ebp
 8048519:       89 e5                   mov    %esp,
 804851b:       83 ec 18                sub    $0x18
 804851e:       c7 04 24 89 86 04 08    movl   $0x80
 8048525:       e8 76 fe ff ff          call   80483
 804852a:       c9                      leave
```

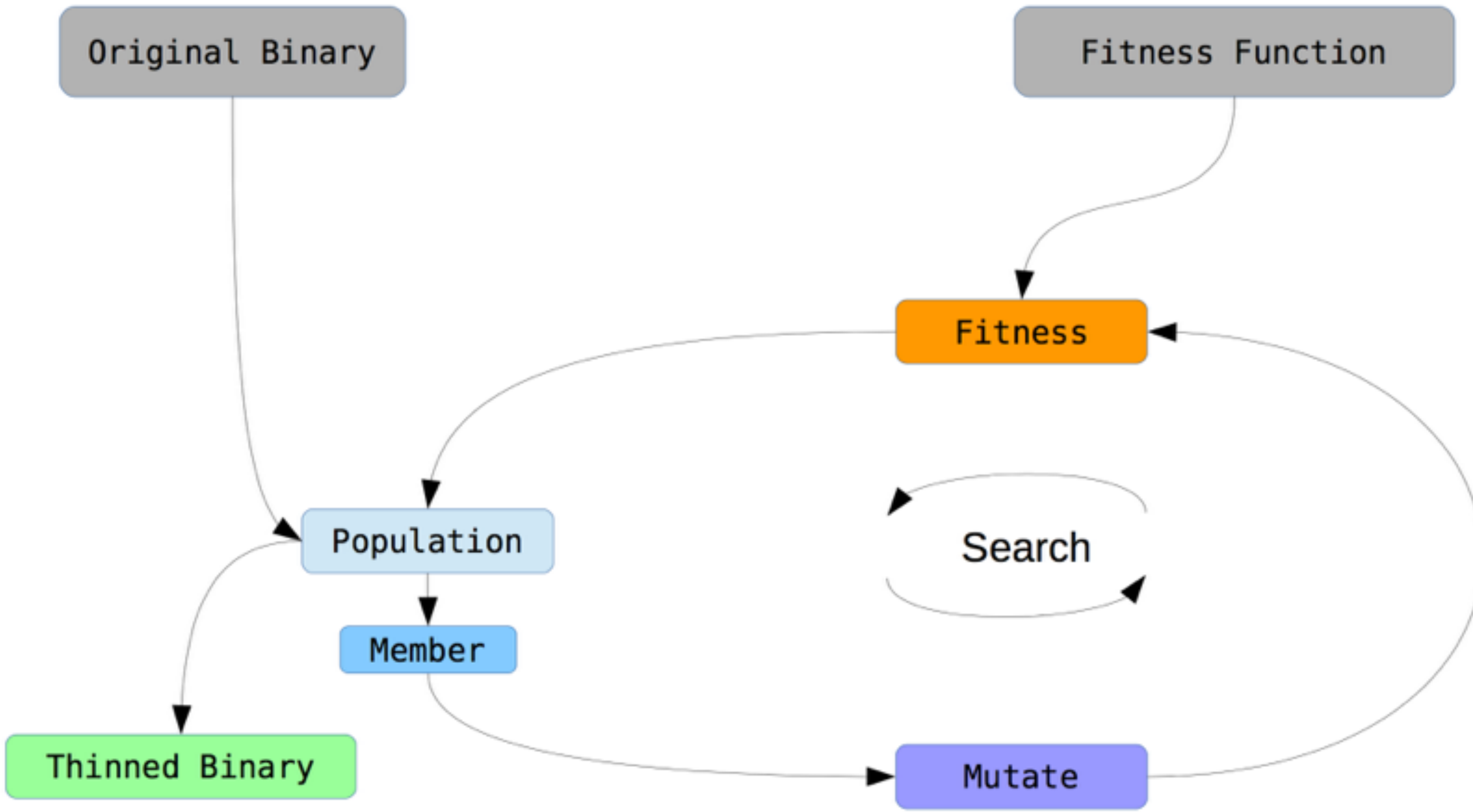
Changes: 3

GA-based Thinning

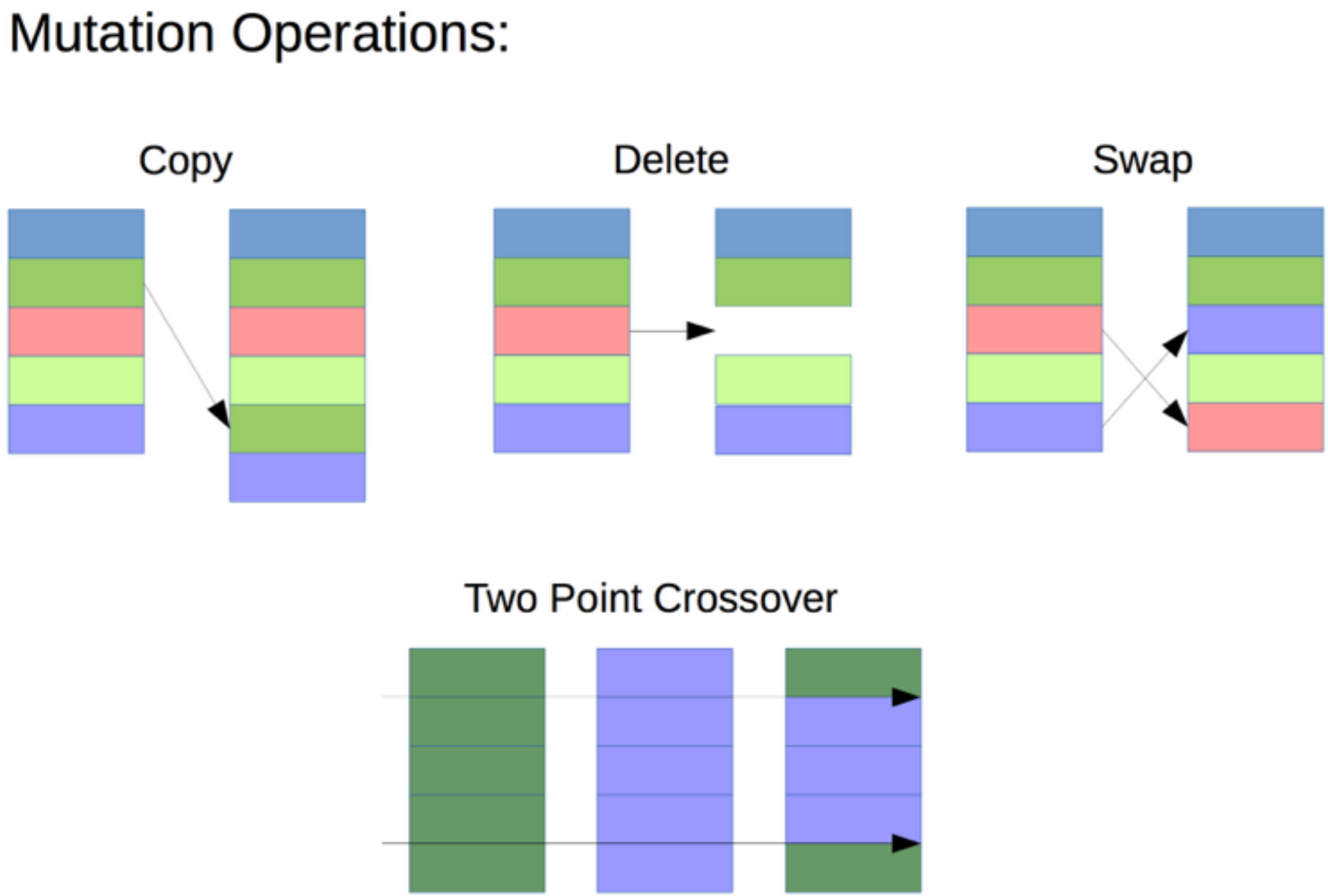
- Genetic Algorithm feature removal
- Software Evolution
    - Developed by Eric Schulte from University of New Mexico



Genetic Algorithm Overview



Binary Mutation



Source: modified from *Post-compiler software optimization for reducing energy* by Schulte, Dorn, Harding, Forrest, and Weimer

Fitness Script

- Script with test cases run by GA
- Basic overview:

1. Variant passed in as script argument
2. Generate many test cases for feature(s) to keep
3. Run each test case on original and variant binaries
4. If output differs, exit with bad fitness, otherwise continue until test cases finish
5. Calculate fitness value

Fitness Calculation

Program size - number of nops, adjusted with largest return value

- PROG_SIZE = length(program_variant)
- PROG_NOPS = count_nops(program_variant)
- FN = PROG_SIZE - PROG_NOPS
- FITNESS = FN + RETVAL * 5

Fitness Script For echo

- Choose 0-5 options from: {-e, -E}
- Choose 0-1000 random characters for string
- Remove special characters
- Test variant and compare with original
  - Same -> continue
  - Diff -> fail, print bad fitness and terminate
- Repeat 1000 times
- Calculate fitness and terminate

Results For echo

Goal: remove -n feature

- Before:

```
[jason]$ echo -n "test"
test[jason]$
```

  - Non-NOP instructions: 2,609

- After:

```
[jason]$ ./thinned_300000 -n "test"
test
[jason]$
```

  - Non-NOP instructions: 2,173
  - 436 fewer instructions — ~17%

Fitness Script For sha1sum

Test case 1:

- Choose random number of random characters for string
- Test string hash of variant and compare with original
  - Same -> continue
  - Diff -> fail, print bad fitness and terminate
- Repeat 50 times

Test case 2:

- Make random file filled with random number of random bytes
- Test file hash of variant and compare with original
  - Same -> continue
  - Diff -> fail, print bad fitness and terminate
- Repeat 50 times

- Calculate fitness and terminate

Results For sha1sum

Goal: Remove unused features — keep string/file hashing

- Before:

```
[jason]$ sha1sum test_file
eba19174a7bf59dfe65a805563d3cb27d9ed673d  test_file
```

  - Non-NOP instructions: 5,049

- After:

```
[jason]$ ./thinned_300000 test_file
eba19174a7bf59dfe65a805563d3cb27d9ed673d  test_file
```

  - Non-NOP instructions: 4,764

  - 285 fewer instructions — ~6%

Minimized Changes

- Terminate as soon as feature is removed
  - fitness script gives perfect fitness (zero)
  - -n removed from echo — 706 fitness evaluations

- Delta debugging
  - Undo changes that do not contribute to fitness

Delta Debugging

- Feature removed: Fitness == 0
- Echo result: 1 instruction

```
[Jason]$ diff echo.s thinnedDDB.s
2c2
< echo:     file format elf64-x86-64
---
> runthinned_DDB:     file format elf64-x86-64
650c650,653
<   401700:     40 80 fe 6e              cmp    $0x6e,%sil
---
>   401700:     90                       nop
>   401701:     90                       nop
>   401702:     90                       nop
>   401703:     90                       nop
[Jason]$ 
```

Current Issues and Limitations

- Machine-dependent
- No GUI programs
- Large programs
- Remove error handling
  - Unintentional program failure

Future Work

I hope you are excited about the future - and ready to transform!

Larger Open Source Software

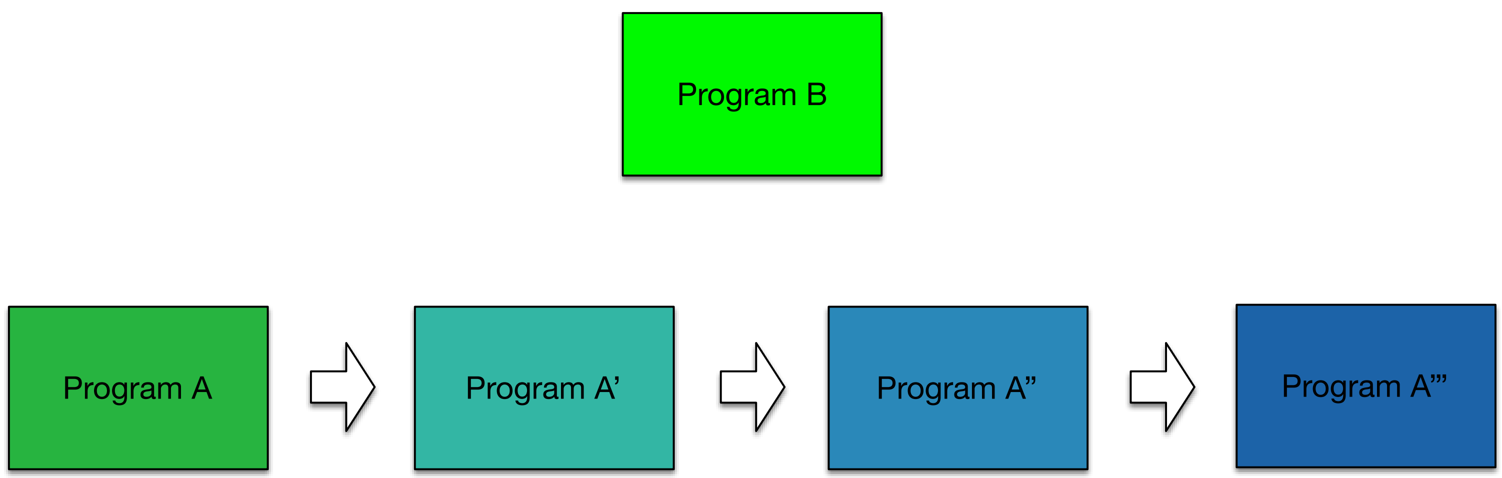
- Try thinning approaches on [slightly] larger **open source** software
- Will need to develop better ways of randomly generating test cases for much larger programs
  - Currently requires knowledge of program usage and arguments (manpage learnin')

GA-Based Diversification

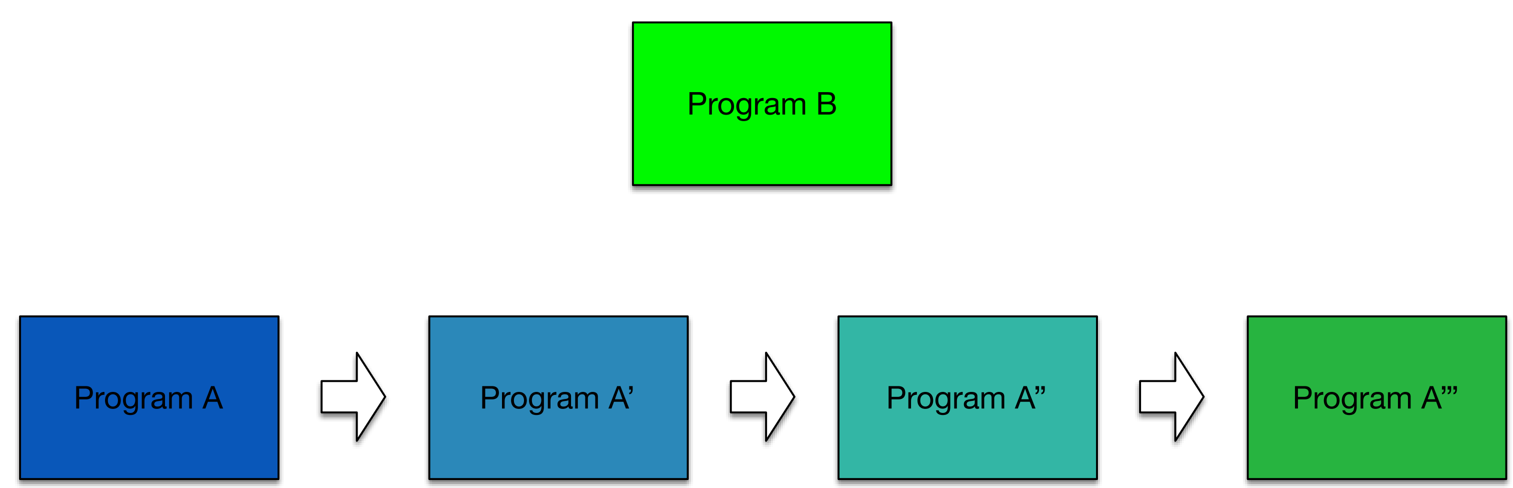
- Why diversification?
  - Change program structure
  - Code-reuse exploits rely on specific program structure

GA-Based Diversification

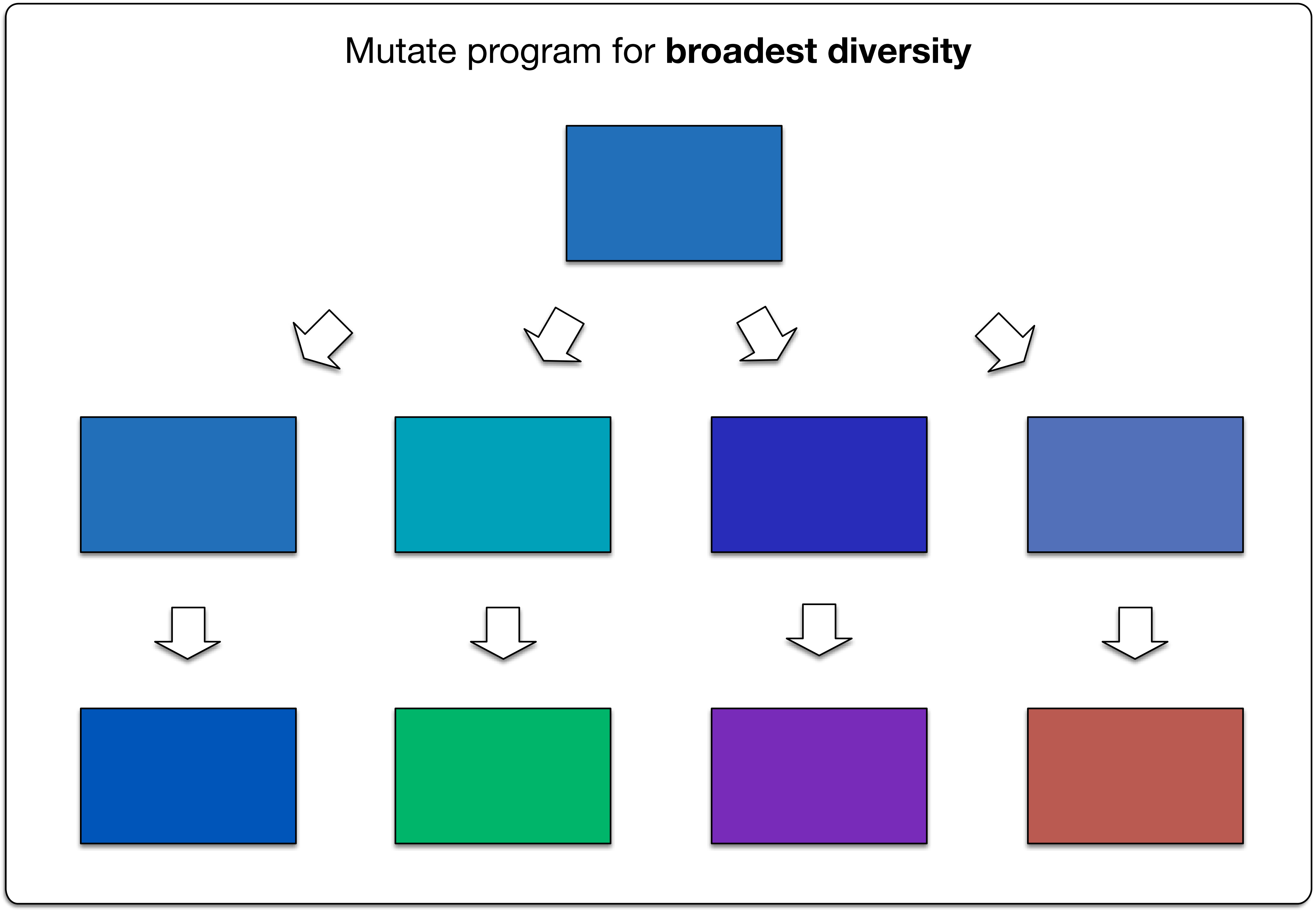
Mutate **Program A** to “look” different from **Program B**



Mutate **Program A** to “look” like **Program B**



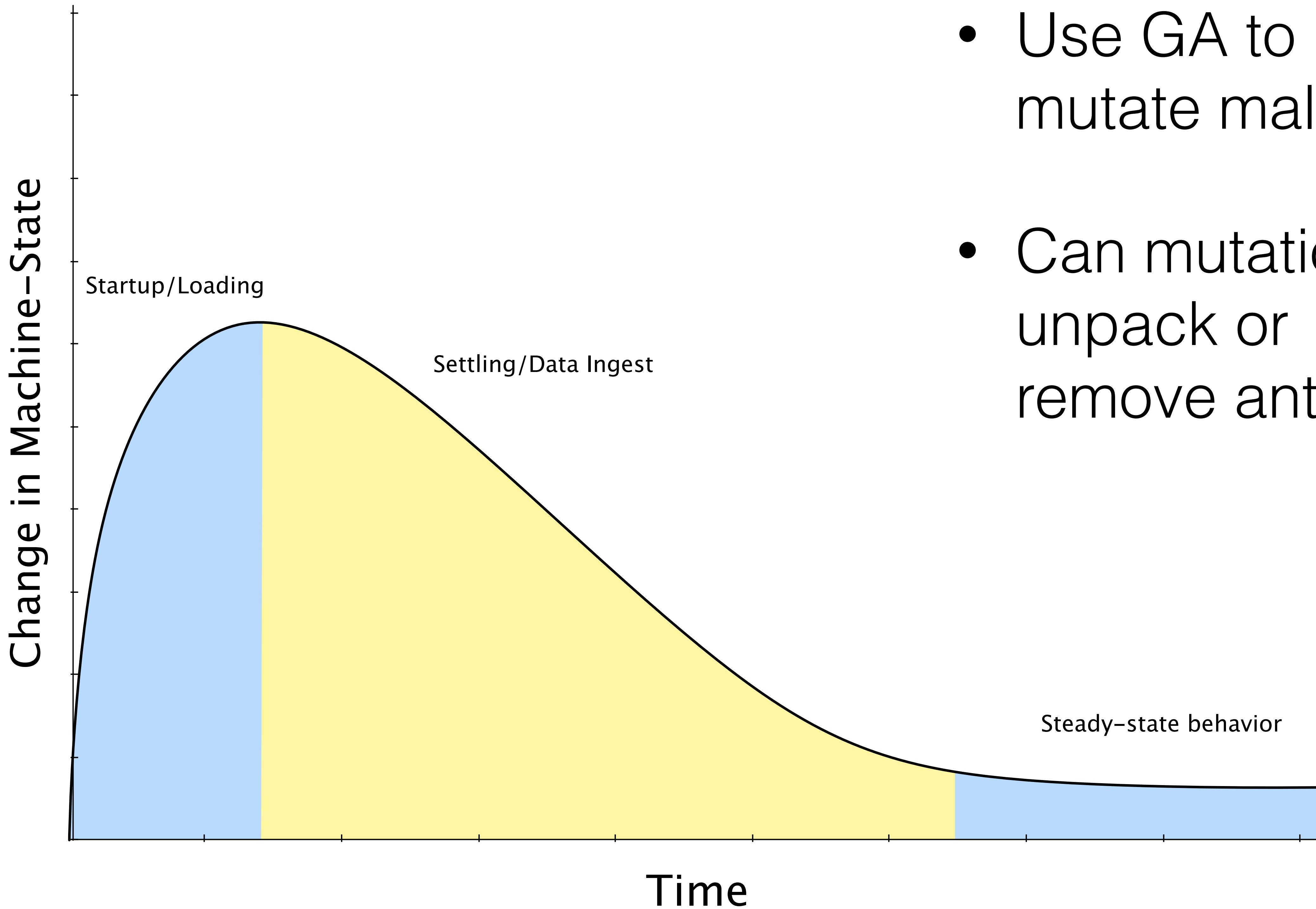
GA-Based Diversification



Semantic Preserving Binary Transformations

- Learn transformations based on GA results
  - Transformations to minimize instructions
  - Transformations to increase diversity

GA-Assisted Malware Reversing



- Use GA to mutate malware
- Can mutations unpack or remove anti-RE?

Other Ideas

- Use tracing to influence GA
- Performance
- Failing gracefully

Questions?

Backup slides

Manual Binary Modification

Showing removal of function from simple program

```
void fun1()
{
   printf("This is function 1\n");
}

void fun2()
{
   printf("This is function 2\n");
}

void fun3()
{
   printf("This is function 3\n");
}

void fundef()
{
   printf("Unsupported function\n");

}
```

```
int main()
{
   int option;
   printf("Enter option: ");
   scanf("%d", &option);
   switch(option)
   {
      case 1:
         fun1();
         break;
      case 2:
         fun2();
         break;
      case 3:
         fun3();
         break;
      default:
         fundef();
         break;
   }
   return 0;
}
```

sjf@schsim1: ~

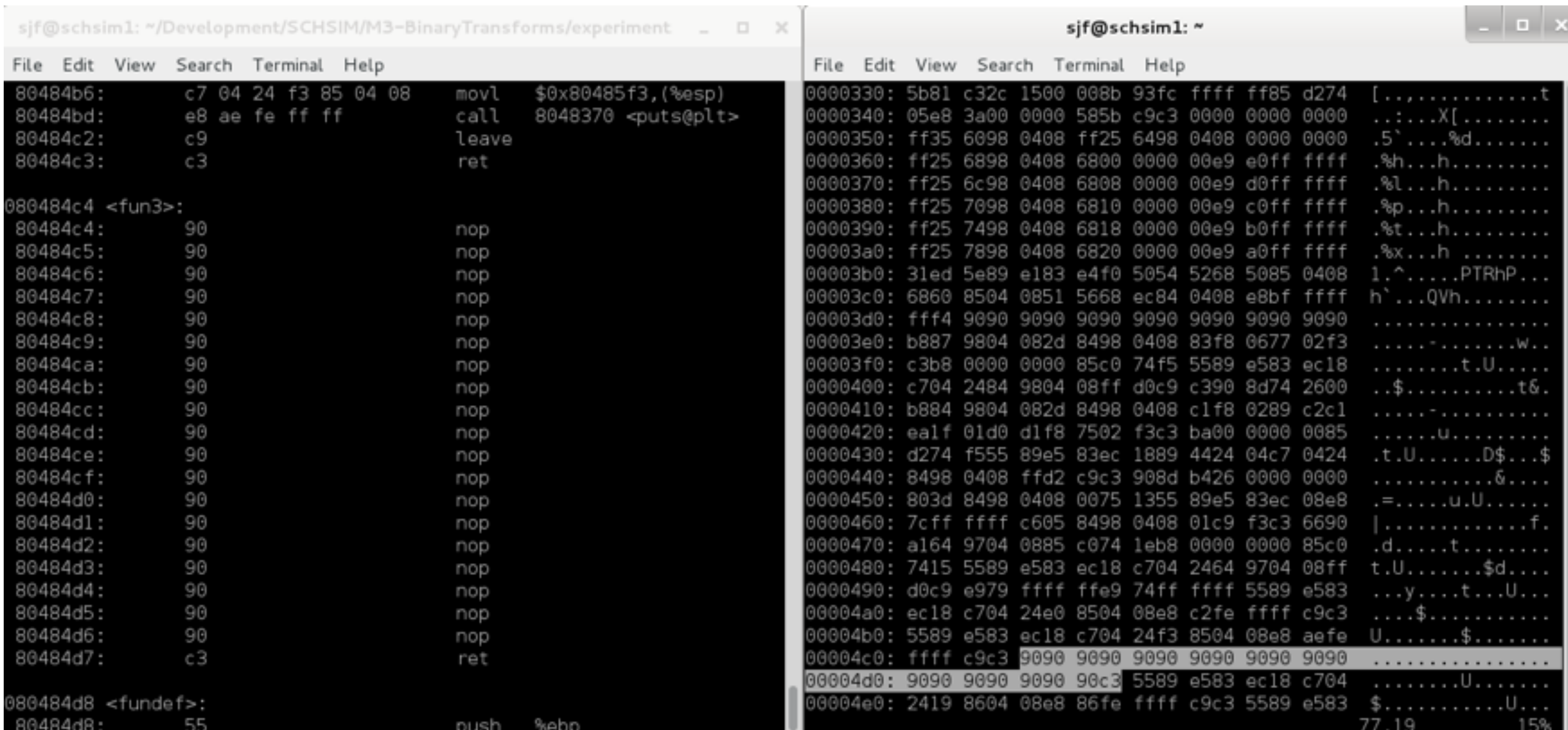
File Edit View Search Terminal Help

```
080484c4 <fun3>:
 80484c4:       55                      push   %ebp
 80484c5:       89 e5                   mov    %esp,%ebp
 80484c7:       83 ec 18                sub    $0x18,%esp
 80484ca:       c7 04 24 06 86 04 08    movl   $0x8048606,(%esp)
 80484d1:       e8 9a fe ff ff          call   8048370 <puts@plt>
 80484d6:       c9                      leave
 80484d7:       c3                      ret

080484d8 <fundef>:
 80484d8:       55                      push   %ebp
 80484d9:       89 e5                   mov    %esp,%ebp
 80484db:       83 ec 18                sub    $0x18,%esp
 80484de:       c7 04 24 19 86 04 08    movl   $0x8048619,(%esp)
 80484e5:       e8 86 fe ff ff          call   8048370 <puts@plt>
 80484ea:       c9                      leave
 80484eb:       c3                      ret

080484ec <main>:
 80484ec:       55                      push   %ebp
 80484ed:       89 e5                   mov    %esp,%ebp
 80484ef:       83 e4 f0                and    $0xfffffff0,%esp
 80484f2:       83 ec 20                sub    $0x20,%esp
 80484f5:       c7 04 24 2e 86 04 08    movl   $0x804862e,(%esp)
 80484fc:       e8 5f fe ff ff          call   8048360 <printf@plt>
 8048501:       8d 44 24 1c             lea    0x1c(%esp),%eax
 8048505:       89 44 24 04             mov    %eax,0x4(%esp)
 8048509:       c7 04 24 3d 86 04 08    movl   $0x804863d,(%esp)
:
```

sjf@schsim1: ~

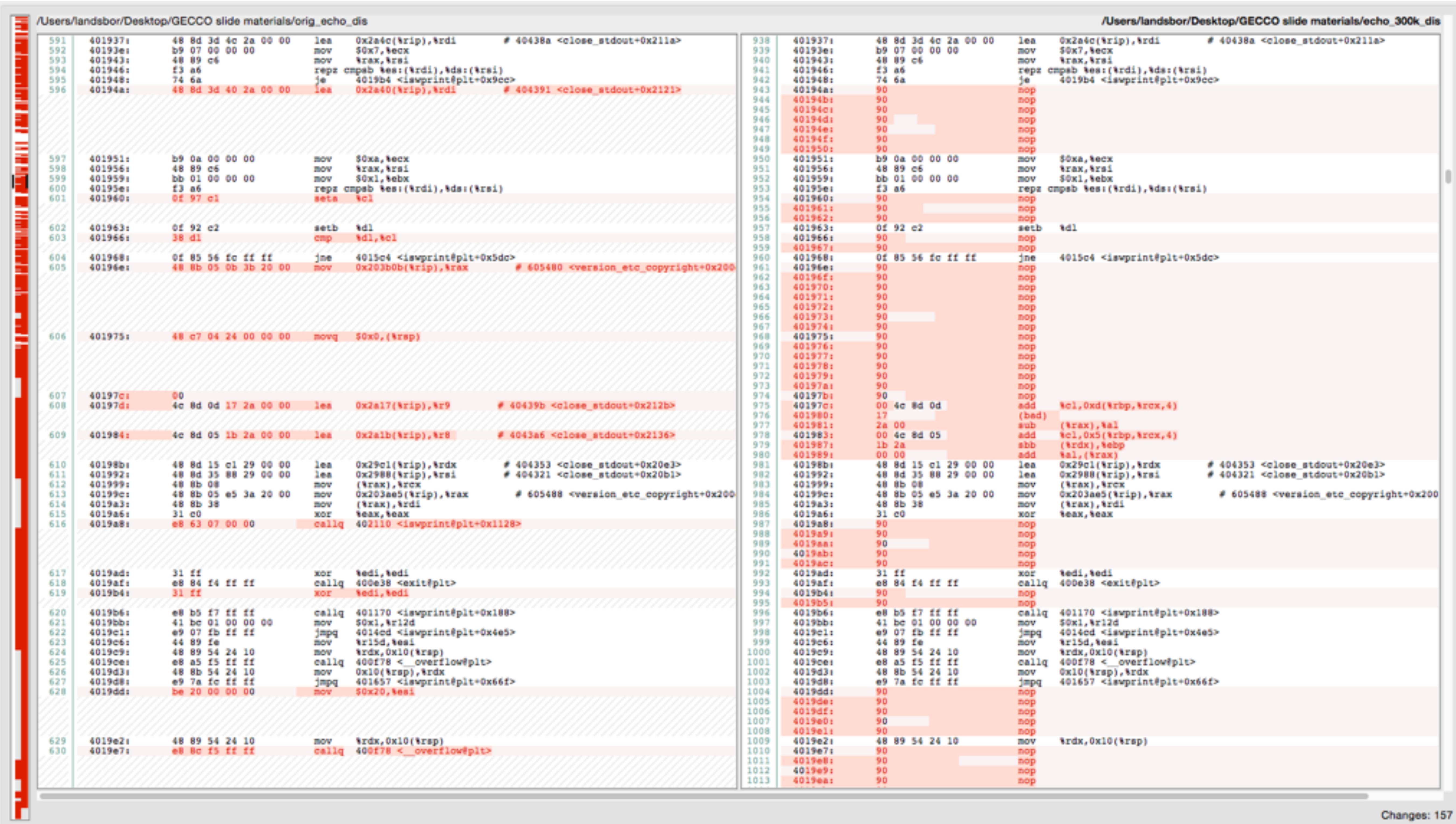
File Edit View Search Terminal Help

```
0000320: 0705 0000 5589 e553 83ec 04e8 0000 0000  ....U..S........
0000330: 5b81 c32c 1500 008b 93fc ffff ff85 d274  [..,...........t
0000340: 05e8 3a00 0000 585b c9c3 0000 0000 0000  ..:...X[........
0000350: ff35 6098 0408 ff25 6498 0408 0000 0000  .5`....%d.......
0000360: ff25 6898 0408 6800 0000 00e9 e0ff ffff  .%h...h.........
0000370: ff25 6c98 0408 6808 0000 00e9 d0ff ffff  .%l...h.........
0000380: ff25 7098 0408 6810 0000 00e9 c0ff ffff  .%p...h.........
0000390: ff25 7498 0408 6818 0000 00e9 b0ff ffff  .%t...h.........
00003a0: ff25 7898 0408 6820 0000 00e9 a0ff ffff  .%x...h ........
00003b0: 31ed 5e89 e183 e4f0 5054 5268 5085 0408  1.^.....PTRhP...
00003c0: 6860 8504 0851 5668 ec84 0408 e8bf ffff  h`...QVh........
00003d0: fff4 9090 9090 9090 9090 9090 9090 9090  ................
00003e0: b887 9804 082d 8498 0408 83f8 0677 02f3  .....-.......w..
00003f0: c3b8 0000 0000 85c0 74f5 5589 e583 ec18  ........t.U.....
0000400: c704 2484 9804 08ff d0c9 c390 8d74 2600  ..$..........t&.
0000410: b884 9804 082d 8498 0408 c1f8 0289 c2c1  .....-..........
0000420: ea1f 01d0 d1f8 7502 f3c3 ba00 0000 0085  ......u.........
0000430: d274 f555 89e5 83ec 1889 4424 04c7 0424  .t.U......D$...$
0000440: 8498 0408 ffd2 c9c3 908d b426 0000 0000  ...........&....
0000450: 803d 8498 0408 0075 1355 89e5 83ec 08e8  .=.....u.U......
0000460: 7cff ffff c605 8498 0408 01c9 f3c3 6690  |.............f.
0000470: a164 9704 0885 c074 1eb8 0000 0000 85c0  .d.....t........
0000480: 7415 5589 e583 ec18 c704 2464 9704 08ff  t.U.......$d....
0000490: d0c9 e979 ffff ffe9 74ff ffff 5589 e583  ...y....t...U...
00004a0: ec18 c704 24e0 8504 08e8 c2fe ffff c9c3  ....$...........
00004b0: 5589 e583 ec18 c704 24f3 8504 08e8 aefe  U.......$.......
00004c0: ffff c9c3 5589 e583 ec18 c704 2406 8604  ....U.......$...
00004d0: 08e8 9afe ffff c9c3 5589 e583 ec18 c704  ........U.......
                                                77,19         15%
```

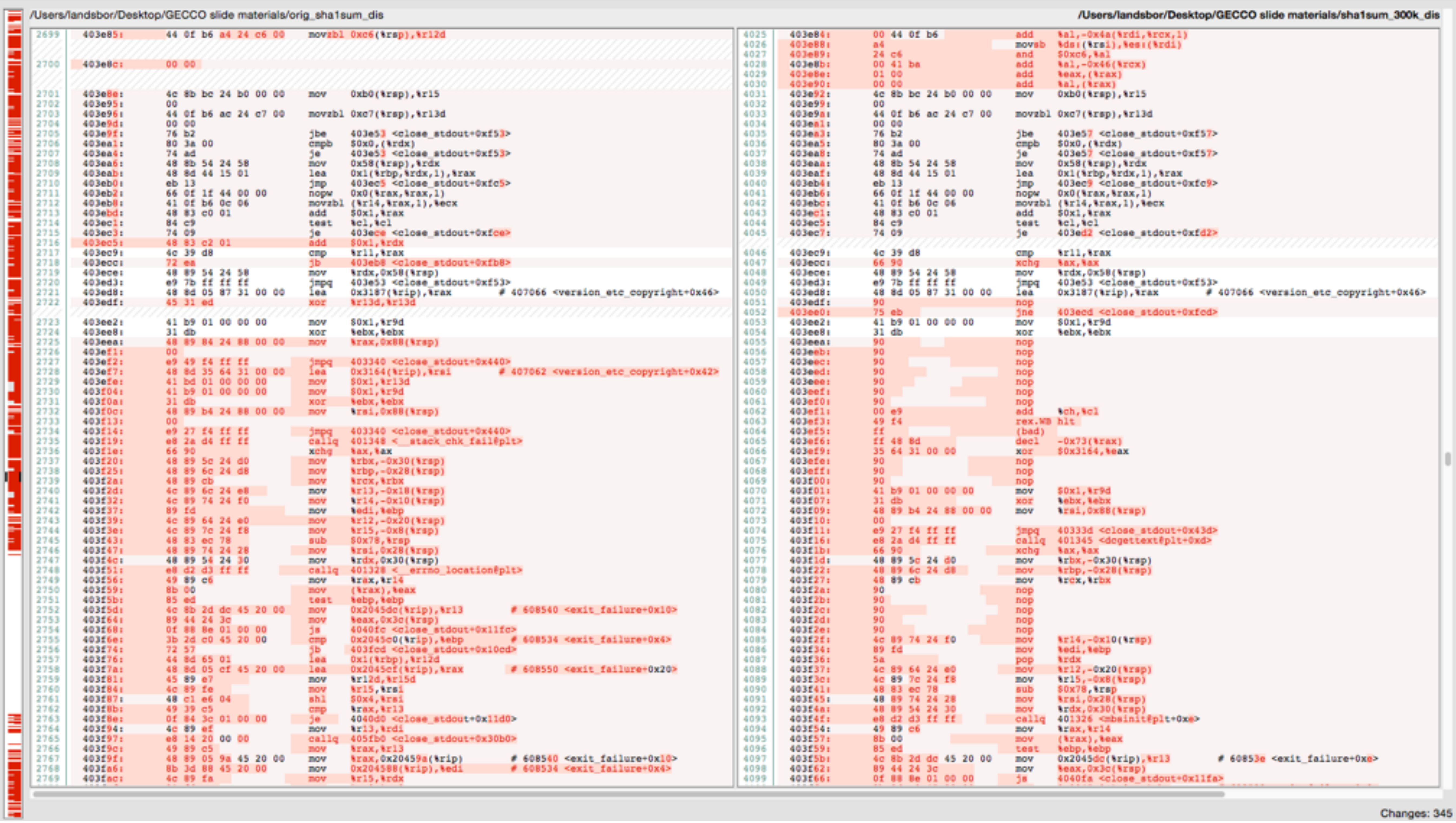

sjf@schsim1: ~/Development/SCHSIM/M3-BinaryTransforms/experiment
File Edit View Search Terminal Help
80484b6: c7 04 24 f3 85 04 08 movl $0x80485f3,(%esp)
80484bd: e8 ae fe ff ff call 8048370 <puts@plt>
80484c2: c9 leave
80484c3: c3 ret
080484c4 <fun3>:
80484c4: 90 nop
80484c5: 90 nop
80484c6: 90 nop
80484c7: 90 nop
80484c8: 90 nop
80484c9: 90 nop
80484ca: 90 nop
80484cb: 90 nop
80484cc: 90 nop
80484cd: 90 nop
80484ce: 90 nop
80484cf: 90 nop
80484d0: 90 nop
80484d1: 90 nop
80484d2: 90 nop
80484d3: 90 nop
80484d4: 90 nop
80484d5: 90 nop
80484d6: 90 nop
80484d7: c3 ret
080484d8 <fundef>:
80484d8: 55 push %ebp
sjf@schsim1: ~
File Edit View Search Terminal Help
0000330: 5b81 c32c 1500 008b 93fc ffff ff85 d274
0000340: 05e8 3a00 0000 585b c9c3 0000 0000 0000
0000350: ff35 6098 0408 ff25 6498 0408 0000 0000
0000360: ff25 6898 0408 6800 0000 00e9 e0ff ffff
0000370: ff25 6c98 0408 6808 0000 00e9 d0ff ffff
0000380: ff25 7098 0408 6810 0000 00e9 c0ff ffff
0000390: ff25 7498 0408 6818 0000 00e9 b0ff ffff
00003a0: ff25 7898 0408 6820 0000 00e9 a0ff ffff
00003b0: 31ed 5e89 e183 e4f0 5054 5268 5085 0408
00003c0: 6860 8504 0851 5668 ec84 0408 e8bf ffff
00003d0: fff4 9090 9090 9090 9090 9090 9090 9090
00003e0: b887 9804 082d 8498 0408 83f8 0677 02f3
00003f0: c3b8 0000 0000 85c0 74f5 5589 e583 ec18
0000400: c704 2484 9804 08ff d0c9 c390 8d74 2600
0000410: b884 9804 082d 8498 0408 c1f8 0289 c2c1
0000420: ea1f 01d0 d1f8 7502 f3c3 ba00 0000 0085
0000430: d274 f555 89e5 83ec 1889 4424 04c7 0424
0000440: 8498 0408 ffd2 c9c3 908d b426 0000 0000
0000450: 803d 8498 0408 0075 1355 89e5 83ec 08e8
0000460: 7cff ffff c605 8498 0408 01c9 f3c3 6690
0000470: a164 9704 0885 c074 1eb8 0000 0000 85c0
0000480: 7415 5589 e583 ec18 c704 2464 9704 08ff
0000490: d0c9 e979 ffff ffe9 74ff ffff 5589 e583
00004a0: ec18 c704 24e0 8504 08e8 c2fe ffff c9c3
00004b0: 5589 e583 ec18 c704 24f3 8504 08e8 aefe
00004c0: ffff c9c3 9090 9090 9090 9090 9090 9090
00004d0: 9090 9090 9090 90c3 5589 e583 ec18 c704
00004e0: 2419 8604 08e8 86fe ffff c9c3 5589 e583
77,19 15%

GA-Thinning

Echo thinned



Sha1sum thinned



Simple program and echo plots