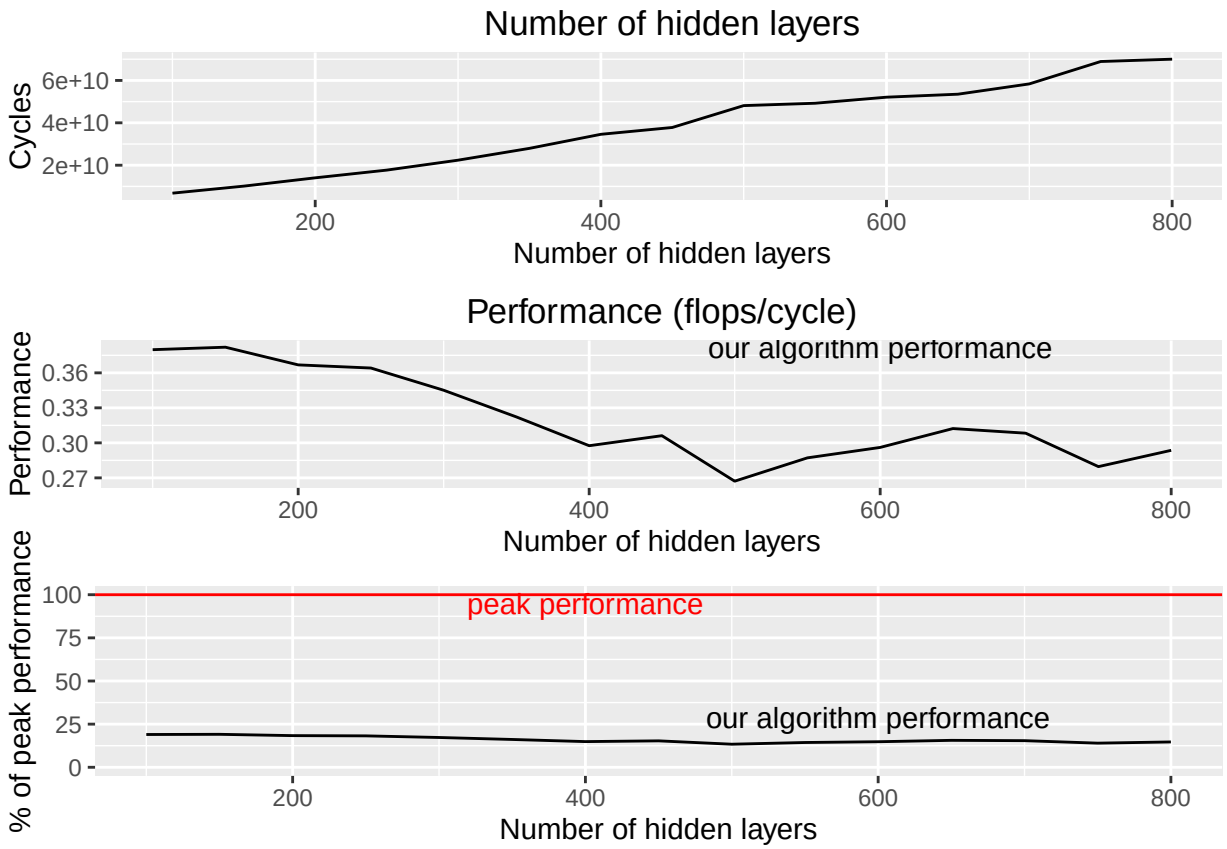
Initial Performance

Prabhakaran, Sergio, Ales

28 April 2016

Performance

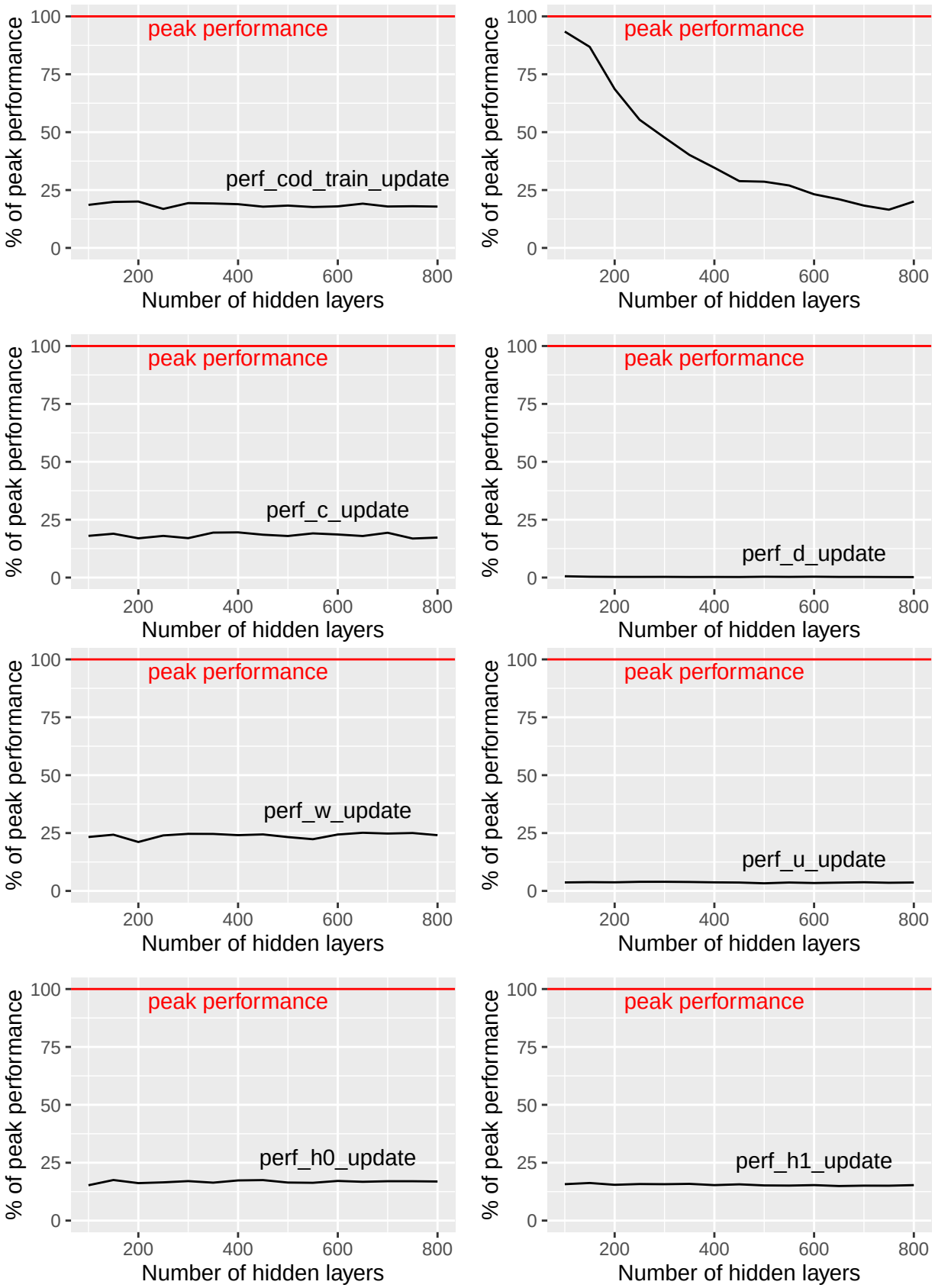


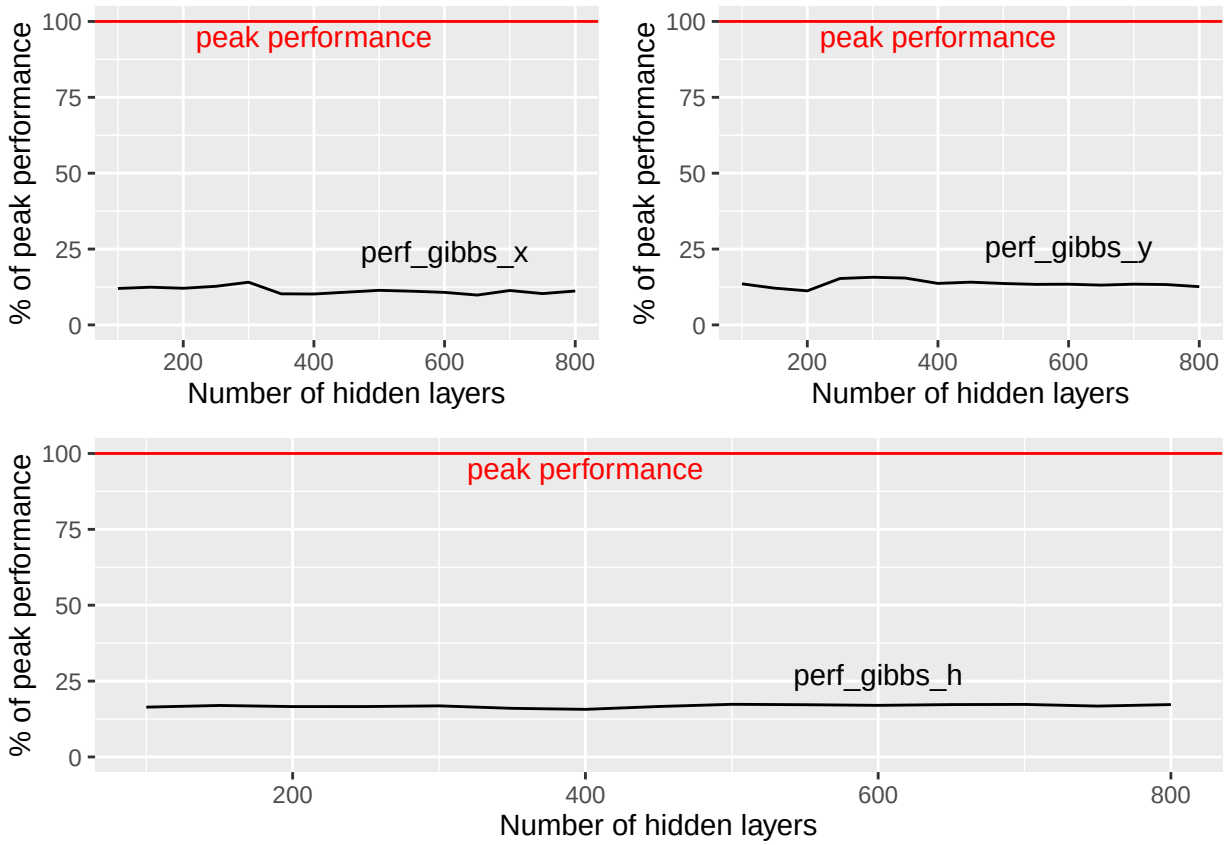
Note: The peak performance is just a little bit over 2 flops/cycle but that is not significant.

Functions Performance

Note: Compiltion with O0 and no vectorization

```
## Warning: Removed 1 rows containing missing values (geom_text).
```


peak performance
perf_cod_train_update
perf_cod_update
perf_c_update
perf_d_update
perf_w_update
perf_u_update
perf_h0_update
perf_h1_update
% of peak performance
Number of hidden layers
0
25
50
75
100
200
400
600
800



Operations Counts and Optimization Ideas

```
H_update:
    adds:     n 785
    mults:    n 784
    sigmoid:  n 4
    reads:    n 1570
    writes:   n
    if:       n 10  (should we count them???)

    All: 1573 n

    Comment: 1st inner loop: loop unrolling + scalar replacement
             2nd inner loop: full unrolling (K = 10)
             outer loop: unrolling + scalar replacement
             vector SIMD
             consider cache misses (blocking?, ...)
             if vs mask???

sigmoid:
    adds:     2
    mults:
    div:      1
    exp:      1
```

```
    All: 4

W_update:
    adds:   n 2*786
    mults:  n 3*786
    reads:  n 5*786
    writes: n 786

    All: 3930 n

    Comment: loop unrolling + scalar replacement to rearrange adds and mults
             SIMD instructions


b_update:
    adds:   2*786
    mults:  786
    read:   3*786
    writes: 786

    All: 2358

    Comment: loop unrolling
             SIMD instructions

c_update:
    adds:   2n
    mults:  n
    reads:  3n
    writes: n

    All: 3n

    Comment: loop unrolling
             SIMD instructions

d_update:
    adds:   bound at most 2
    mults:  0
    reads:  20
    writes: 20

    All: 2

    Comment: is it possible to do sth with IFs???

U_update:
    adds:   2 n
    mults:  2 n
    reads:  3 n
    writes: 2 n

    All of them upper bound
```

```
    All: 4 n  (40before)

    Comment: IFs???

gibbs_H:
    adds:     n 787
    mults:    n 786
    sigmoid:  4n
    uniform:  2n
    reads:    n(2 + 2*786)
    writes:   n
    ifs:      n

    All: 1579 n

    Comment: inner loop - loop unrolling + scallar replacement
             outer loop - loop unrolling
             SIMD instructions

gibbs_Y:
    adds:     10n + 11
    mults:    10n
    divs:     10
    uniform:  2
    reads:    20n + 40
    writes:   20
    ifs:      10
    allocation: 10 doubles

    All: 20 n + 23

    Comment: loop unrolling
             full unrolling of K-loops
             SIMD instructions

gibbs_X:
    adds:     n 786
    mults:    n 786
    sigmoid   4*786
    uniform   2*786
    reads:    786 + 2*n*786
    writes:   786
    ifs:      786

    All: 1572 n + 4716

    Comment: loop unrolling + sc. replacement
             SIMD instructions


uniform:
    rand: 1
    div:  1
```

```
    All: 2

COD_training_update:
    h_update
    allocations: n ints, 786 ints, n doubles
    gibbs_H
    ...

    All flops: 10263 n + 7099

COD_train:
    All: 2500*(10263 n + 7099)

Predict:
    adds:   n 786 + n + 3Kn + K
    mults:  n 786
    log:    n 10
    exp:    n 10
    reads:  2n786 + n + 20n + 20
    writes: n + 10
    ifs:    10

    allocations: n doubles, 10 doubles,
```

Questions

Q: add:1, mult:1, div:39, exp:?, random:?

Q: Should we count IFs or not? I think that they don't impact floationg operations

Q: Measuring - I think we should measure with exactly the same input and average it

Q: Measuring over unittests with smaller matrices and other input thigs? Yes/No?

Q: Validation - are unittests enough? To test it in details is quite hard.

Q: Input size: Can we take subset of data for faster computation?