An Evaluation of Double-Double Precision Operation for Iterative Solver Library using AVX

Toshiaki Hishinuma[†], Akihiro Fujii[†], Teruo Tanaka[†] and Hidehiko Hasegawa[††]

[†]Department of Computer Science, Faculty of Informatics, Kogakuin University, Tokyo, Japan

[††]Faculty of Library, Information and Media Science, University of Tsukuba, Tsukuba, Japan

Engineer our Future 2012

Introduction

In order to improve accuracy, scientific computing needs more than single/double precisions. We evaluated behavior of high precision operation routines using new Intel architecture AVX.

Double-Double precision operation

- Double-Double precision composed of two Double precision data
- It has 104bit significant part
- It takes 10-30 times longer than Double precision operation

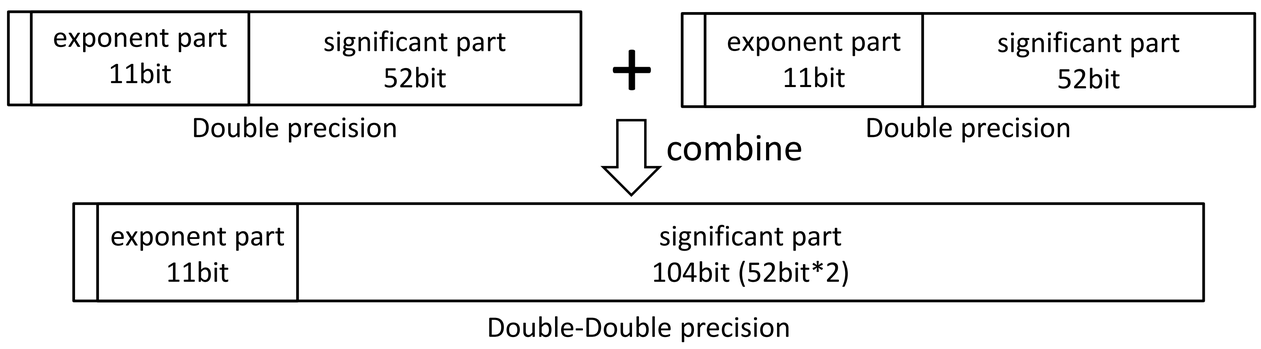

Fig.1 Double-Double precision data

Acceleration using AVX

SSE2 and AVX supported by Intel

- SSE2 : 2 Double operations (in 2000)
- AVX : 4 Double operations (New, in 2011)
- Performance of AVX is twice faster than SSE2 theoretically

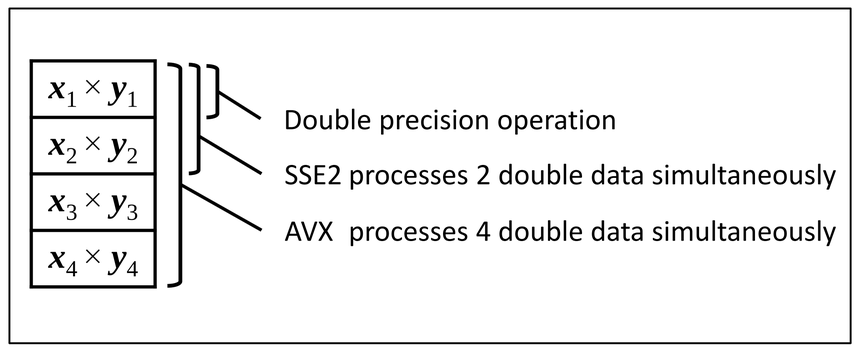

Fig.2 Architecture of AVX and SSE2

Vector operations

Table 1 Double-Double precision vector operations

Name	axpy	axpyz	xpay	scale	dot	nrm2
Operations	$y = ax + y$	$z = ax + y$	$y = x + ay$	$x = ax$	$val = x \cdot y$	$val = \|\|x\|\|_2$

CPU : Intel core i 7 2600K 3.4GHz

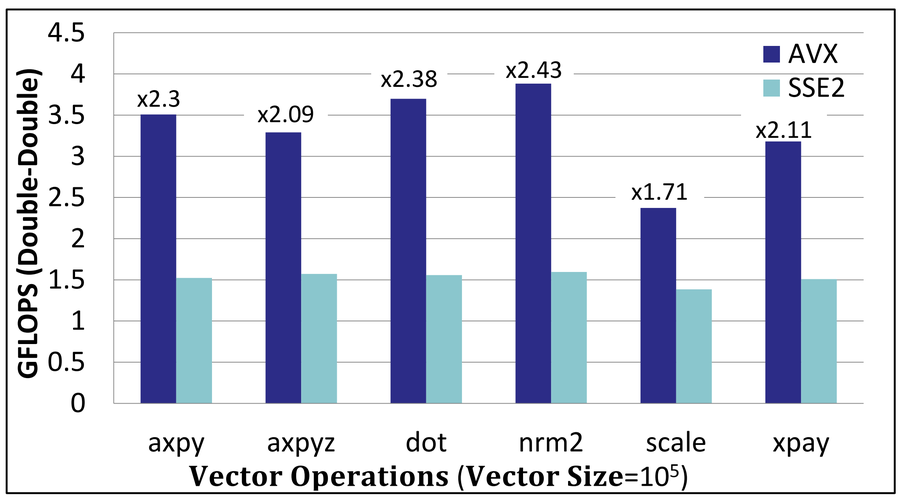

Fig.3 Performance of vector operations (4Threads)

- The speedup ratio varied from 1.7 to 2.4
- Some performance of AVX are more than twice of SSE2
  - AVX uses three operand instruction

Three operand instruction (AVX)

w = x * y + z

```
mul(x , y , temp)     //temp = x * y
add(temp , z , w)     //w = temp + z
```

Two operand instruction (SSE2)

w = x * y + z

```
mul(y , x)      //x = x * y
add(x, z)       //z = z + x
mov(z , w)      //w = z
```

Fig.4 Difference between three and two operand instruction (example)

Sparse matrix and vector multiplication

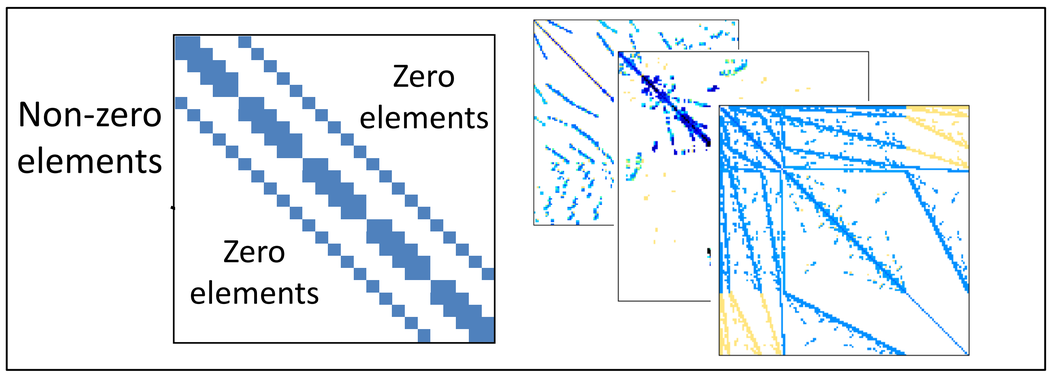

Fig.5 Various structure of non-zero elements of Sparse matrices

- Performance of sparse matrix and vector multiplication depends on number of non-zero element/row, regardless of structure of non-zero elements

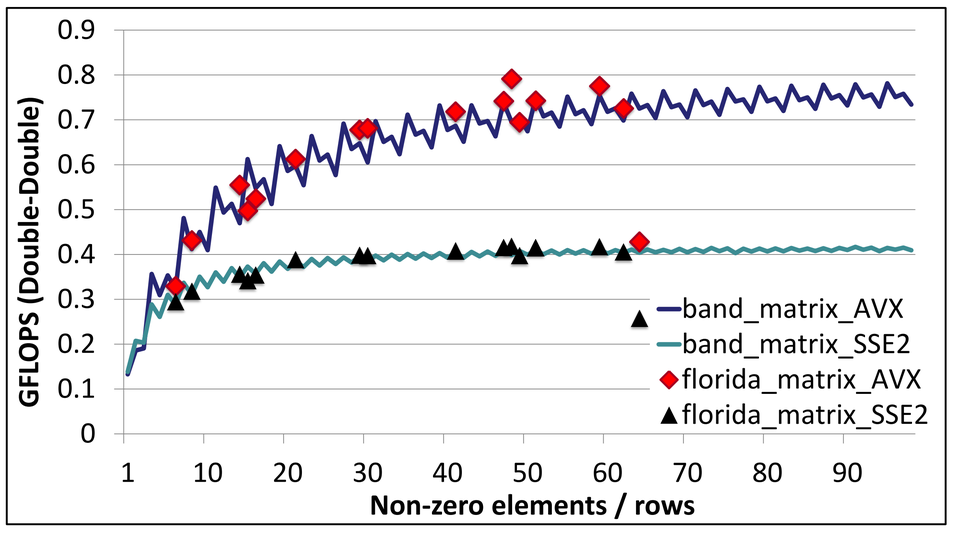

Fig.6 Performance of sparse matrix and vector multiplication (1Thread)

References

- Bailey, D, H.: *High-Precision Floating-Point Arithmetic in Scientific Computation, computing in Science and Engineering, pp. 54–61 (2005).*
- T. Hishinuma, et al.: *Acceleration of Double-Double Precision Operation for Iterative Solver Library using AVX , IPSJ, Vol.2012-HPC-135 No.16, pp.1-6, SWoPP (2012.8)(in Japanese)*

Conclusion

- Performance of AVX was more than twice of SSE2, because AVX uses three operand instruction.
- Performance of sparse matrix and vector multiplication strongly depends on nonzero elements/rows.