

32.973.202-018.2
50.07.05,50.33.04
519.681.5:004.272.25

.

AVX-512

.

.

.

.

.

.

.

, AVX-512,

AVX-512,

512-

C 2019
C
C

(, 2019
) , 2019

256-bit AVX Intel x86
[1], Intel, Intel
Xeon Phi KNL [2] Intel Xeon Skylake.

AVX-512F (Foundation) -

AVX-512PF (PreFetch)

AVX-512ER (Exponential and Reciprocal) -

AVX-512CD (Conflict Detection)
, AVX-512BW AVX-512DQ, Skylake.
AVX-512
52-

AES

AVX-512

icc - [3, 4],
AVX-512. -

512-

C.

permutevar swizzle, shuffle, permute -

shuf
AVX-512

1.

[13]

1).

1 [14–16]:

1.

<hr/>	
<hr/>	
, 1959 .	$k_1 = b^{\frac{N}{2}}c; k_i = b^{\frac{k_i-1}{2}}c; k_t = 1$
, 1963 .	$2^i - 1 \quad N; i \geq N$
, 1971 .	$2^i - 3^j \quad \frac{N}{2}; i \geq N; j \geq N$
, 1986 .	$k_i = \begin{pmatrix} 9 - 2^i & 9 - 2^{\frac{i}{2}} + 1; k \text{ even} \\ 8 - 2^i & 6 - 2^{\frac{i+1}{2}} + 1; k \text{ odd} \end{pmatrix}$
<hr/>	

$$k (\quad , 1).$$

1.

2.

AVX-512

16
 $j,$

).

float (AVX-512
(

k

$m[j]$

$m[j \quad k]$

(
)

$i,$

$i_1 \quad i_2$

$$j i_1 \quad i_2 j < k.$$

2

$$\frac{k}{16} \quad 16.$$

$$(\quad) .$$

2.

16

16 (

$$1 < k < 16.$$

16,

$$k = 1.$$

16

shell_sort_k_i_w,

 w k .

[17],

(3).

3.

false,

 w

(
(
).

,

.

,

4:

4.

scatter (4, 25)

(1, 23)

-

scatter
scatter (4,
10, 16, 19).
_mm512_mask_load_ps
_mm512_mask_store_ps,
vmovups.

Intel Xeon Phi KNL Intel Xeon Skylake
vmovaps vmovups
vmovaps [18]. icc

gather
scatter.
3.

$T(k; i)$
 i .
 T):
(

(1)
$$T = \prod_{k=2k_s}^{\times} \prod_{i=k}^{\times 1} l(k; i);$$

$$T_v(k; i)$$

$$k \leq i, \quad w(k) = \min(k; 16).$$
$$b \frac{n-k}{w(k)} c$$

$$(2) \quad T_v = \sum_{k=2}^n \sum_{s=0}^{G(k)-1} \sum_{g=0}^{k+w(k)(g+1)-1} I(k; i) A + \sum_{i=k+w(k)G(k)}^{n-1} I(k; i) A;$$
$$w(k) = \min(k; 16), \quad G(k) = \lfloor \frac{n-k}{w(k)} \rfloor. \quad T = T(n)$$
$$T_v = T_v(n)$$

$$k = 1 \quad (\quad) .$$
$$T^0(n), \quad T_v^0(n) \quad s^0(n)$$

Intel Xeon Phi 7290

KNL.

4.

:

icc -O3

AVX-512 (-xmi c-avx512)

Intel Xeon Phi KNL.

10 , Intel Xeon Phi

7290.

5 6

5. -

6.

float), (16 -
2 (, ,
).

7. 设 G 是一个有 n 个顶点的图, 且 G 中每个顶点的度数都不超过 k . 证明: G 中边数 $m \leq \frac{nk}{2}$.

7. 设 G 是一个有 n 个顶点的图, 且 G 中每个顶点的度数都不超过 k . 证明: G 中边数 $m \leq \frac{nk}{2}$.

8. 设 G 是一个有 n 个顶点的图, 且 G 中每个顶点的度数都不超过 k . 证明: G 中边数 $m \leq \frac{nk}{2}$.

9.

-

$$k = 3$$

10.

-

$$k = 15$$

$$m[l]$$

,

.

,

$$w$$

$$w$$

,

.

,

,

.

$Z = Z \cdot Z + C$), (

(15

2,5-6),

6,5
(

1,6-2,3).

[11]

AVX-512

Intel KNL.

:

[11]

(

),

1,4

C++ STL.

$$k < 16.$$

0065-2019-0016 (reg. no. - 19-119011590098-8).

-10

- [1] Intel 64 and IA-32 Architectures Software Developer's Manual. Combined volumes: 1, 2A, 2B, 2C, 2D, 3A, 3B, 3C, 3D and 4. Intel Corporation. 2019. URL: <https://software.intel.com/en-us/download/intel-64-and-ia-32-architectures-sdm-combined-volumes-1-2a-2b-2c-2d-3a-3b-3c-3d-and-4> (access date: 21.10.2019). "78
- [2] J. Jeers, J. Reinders, A. Sodani. *Intel Xeon Phi processor high performance programming*, Knights Landing Edition, 2 ed., Morgan Kaufmann Publ., 2016, 632 pp. "78
- [3] Intel C++ Compiler 16.0 User and Reference Guide. Intel Corporation. 2015. URL: <https://software.intel.com/en-us/download/intel-c-compiler-160-update-4-user-and-reference-guide> (access date: 21.10.2019). "78
- [4] Intel Intrinsics Guide. URL: <https://software.intel.com/sites/landingpage/IntrinsicsGuide> (access date: 21.10.2019). "78
- [5] W. McDaniel, M. Hohnerbach, R. Canales. et al.. "LAMMPS' PPPM Long-Range Solver for the Second Generation Xeon Phi", *J.M. Kunkel et al. (Eds.): ISC High Performance 2017, LNCS*, 10266 (2017), pp. 61–78. "79
- [6] T. Malas, T. Kurth, J. Deslippe. "Optimization of the Sparse Matrix-Vector Products of an IDR Krylov Iterative Solver in EMGeo for the Intel KNL

: , 2019, 10:4(43), . 77–96.

10.25209/2079-3316-2019-10-4-77-96

http://psta.psi-ras.ru/read/psta2019_4_77-96.pdf

— *Journal of the American Medical Association*, 1997

[illegible]

e-mail: rybakov.aax@gmail.com

[illegible]

e-mail: shumilin@jscs.ru

CSCSTI 50.07.05,50.33.04
UDC 519.681.5:004.272.25

Alexey Rybakov, Sergei Shumilin. *Study of the vectorization efficiency of loop nests with an irregular number of iterations.*

Abstract. Computation vectorization is an important low-level optimization used to create highly efficient parallel code. However, when used in context with an unknown program execution profile, a danger of low effectiveness of the application emerges. This is especially pronounced when vectorizing nests of cycles with an irregular number of iterations of the inner loop. The article discusses a comparison of the theoretical and practical efficiency of vectorization on the example of Shell sorting, since this program code is extremely inconvenient for vectorization.

Key words and phrases: vectorization, AVX-512, loop sockets with an irregular number of iterations, Shell sorting, theoretical acceleration.

2010 *Mathematics Subject Classification:* 68W10; 65P99, 68M07

References



- [1] Intel 64 and IA-32 Architectures Software Developer's Manual. Combined volumes: 1, 2A, 2B, 2C, 2D, 3A, 3B, 3C, 3D and 4. Intel Corporation. 2019. URL: <https://software.intel.com/en-us/download/intel-64-and-ia-32-architectures-sdm-combined-volumes-1-2a-2b-2c-2d-3a-3b-3c-3d-and-4> (data obrashcheniya: 21.10.2019). "78
- [2] J. Jeers, J. Reinders, A. Sodani. *Intel Xeon Phi processor high performance programming*, Knights Landing Edition, 2 ed., Morgan Kaufmann Publ., 2016, 632 pp. "78
- [3] Intel C++ Compiler 16.0 User and Reference Guide. Intel Corporation. 2015. URL: <https://software.intel.com/en-us/download/intel-c-compiler-160-update-4-user-and-reference-guide> (data obrashcheniya: 21.10.2019). "78
- [4] Intel Intrinsics Guide. URL: <https://software.intel.com/sites/landingpage/IntrinsicsGuide> (data obrashcheniya: 21.10.2019). "78
- [5] W. McDoniel, M. Hohnerbach, R. Canales. et al.. "LAMMPS: PPPM Long-Range Solver for the Second Generation Xeon Phi", *J.M. Kunkel et al. (Eds.): ISC High Performance 2017, LNCS*, 10266 (2017), pp. 61—78. "79
- [6] T. Malas, T. Kurth, J. Deslippe. "Optimization of the Sparse Matrix-Vector Products of an IDR Krylov Iterative Solver in EMGeo for the Intel KNL Manycore Processor", *M. Taufer et al. (Eds.): ISC High Performance Workshops 2016, LNCS*, 9945 (2016), pp. 378–389. "79

c A. A. Rybakov, S. S. Shumilin, 2019

c Joint Supercomputer Center of RAS, 2019


c Program Systems: Theory and Applications (design), 2019

10.25209/2079-3316-2019-10-4-77-96

- [7] L. A. Benderskiy, A. A. Rybakov, S. S. Shumilin. "Vektorizatsiya peremnozheniya malorazmernykh matrits spetsial'nogo vida s ispol'zovaniyem instruktsiy AVX-512", *Sovremennyye informatsionnyye tekhnologii i IT-obrazovaniye*, 14:3 (2018), pp. 594–602.  ⁷⁹
- [8] O. Krzikalla, F. Wende, M. Hohnerbach. "Dynamic SIMD vector lane scheduling", *Lect. Notes Comput. Sci.*, 2016, no.9945, pp. 354–365. ^{79, 90}
- [9] B. Cook, P. Maris, M. Shao. "High Performance Optimizations for Nuclear Physics Code MFDn on KNL", *M. Taufer et al. (Eds.): ISC High Performance Workshops 2016, LNCS*, 9945 (2016), pp. 366–377. ⁷⁹
- [10] C. R. Ferreira, K. T. Mandli, M. Bader. "Vectorization of Riemann solvers for the single- and multi-layer shallow water equations", *Proceedings of the 2018 International Conference on High Performance Computing and Simulation, HPCS 2018*, 2018, pp. 415–422. ⁷⁹
- [11] B. Bramas. "Fast Sorting Algorithms using AVX-512 on Intel Knights Landing", *International Journal of Advanced Computer Science and Applications*, 8:10 (2017), pp. 337–344.  ^{79, 91}
- [12] M. S. Gus'kova, L. Yu. Barash, L. N. Shchur.. "Primeneniye AVX512-vektORIZatsii dlya uvelicheniya proizvoditel'nosti generatara psevdosluchaynykh chisel", *Trudy ISP RAN*, 30:1 (2018), pp. 115–126. ⁷⁹
- [13] D. Knut. *Iskusstvo programmirovaniya*. V. 3: *Sortirovka i poisk*, Vil'yams, M., 1994, 832 pp. ⁸⁰
- [14] A168604, posledovatel'nost' Khibbarda. URL: <https://oeis.org/A168604> (data obrashcheniya 21.10.19). ^{80, 86}
- [15] A003586, posledovatel'nost' Pratta. <https://oeis.org/A003586> (data obrashcheniya 21.10.19). ^{80, 86}
- [16] A033622, posledovatel'nost' Sedzhvika. URL: <https://oeis.org/A033622> (data obrashcheniya 21.10.19). ^{80, 86}
- [17] V. Yu. Volkonskiy, S. K. Okunev. "Predikatnoye predstavleniye kak osnova optimizatsii programmy dlya arkhitektury s yavno vyrazhennoy parallel'nost'yu", *Informatsionnyye tekhnologii*, 2003, no.4, pp. 36–45. ⁸³
- [18] Using Intel AVX without Writing AVX. URL: <https://software.intel.com/en-us/articles/using-intel-avx-without-writing-avx> (data obrashcheniya: 21.10.2019). ⁸⁵

Sample citation of this publication:

A. A. Rybakov, S. S. Shumilin. "Study of the vectorization efficiency of loop nests with an irregular number of iterations". *Program Systems: Theory and Applications*, 2019, 10:4(43), pp. 77–96. (In Russian).

 10.25209/2079-3316-2019-10-4-77-96

 http://psta.psi.ras.ru/read/psta2019_4_77-96.pdf