Energy-Efficient Transprecision Techniques for Iterative Refinement

Lee, J., Vandierendonck, H., & Nikolopoulos, D. (Accepted/In press). Energy-Efficient Transprecision Techniques for Iterative Refinement. Poster session presented at Supercomputing'17 (SC17): International Conference on High Performance Computing, Networking, Storage and Analysis, Denver, United States.

Energy-Efficient Transprecision Techniques for Iterative Refinement

JunKyu Lee, Hans Vandierendonck, Dimitrios S. Nikolopoulos
Centre for Data Science and Scalable Computing, Queens University of Belfast, Northern Ireland, UK
junkyu.lee@qub.ac.uk,h.vandierendonck@qub.ac.uk,d.nikolopoulos@qub.ac.uk

ABSTRACT

This paper presents transprecision techniques for iterative refinement, which utilize various precision arithmetic dynamically according to numeric properties of the algorithm and computational latencies depending on precisions. The transprecision techniques were plugged into a mixed precision iterative refinement on an Intel Xeon E5-2650 2GHz core with MKL 2017 and XBLAS 1.0. The transprecision techniques brought further 2.0-3.4X speedups and 3.0-4.1X energy reductions to a mixed precision iterative refinement when double precision solution accuracy was required for forward error and a matrix size was ranged from 4K to 32K.

CCS CONCEPTS

• **Computing methodologies → Linear algebra algorithms**;

KEYWORDS

transprecision; mixed precision; iterative refinement

ACM Reference Format:
JunKyu Lee, Hans Vandierendonck, Dimitrios S. Nikolopoulos. 2017. Energy-Efficient Transprecision Techniques for Iterative Refinement. In *Proceedings of IEEE/ACM Supercomputing 2017, Denver, Colorado USA, November 2017 (SC17),* 2 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Parallel computing generally obtains speedup, but it requires additional power according to the increased number of cores, keeping from energy saving. Mixed precision method is promising to save energy when solving A**x**=**b** using Iterative Refinement (IR), since it obtains speedup without increasing the number of cores [1]. The idea of the mixed precision method is to utilize a low precision for O(N^3) LU solver, while attaining a solution accuracy through O(N^2) refinement, where N is a matrix size. In this paper, we discuss IRs producing double precision accuracy for forward error, therefore a matrix consists of double precision data and double-double precision (dbl-dbl) arithmetic is required for refinement [4]. We name an IR Uni-precision IR (Uni-IR) if it employs double precision arithmetic (i.e., the same precision to the data) for LU solver and dbl-dbl for refinement and Mixed precision IR (Mixed IR) if it employs single precision arithmetic for LU solver and dbl-dbl for refinement.

In order to bring further speedup and energy reduction to Mixed IR, we develop novel Transprecision Techniques (TTs) which utilize various precision arithmetic dynamically according to numeric properties of Mixed IR and computational latencies depending on precisions.

2 TRANSPRECISION TECHNIQUES FOR ITERATIVE REFINEMENT

In earlier work [2], we proposed the Numeric Property 1 (NP 1) of IR in Fig. 1. In this work, we propose NP 2 and 3 in Fig. 1 and develop TTs according to the three NPs. Algorithm I in Fig. 1 describes Uni-IR, Mixed IR, and TTs plugged into Mixed IR. IRs consist of approximation (Step 1) and refinement (Step 2 to 4). Approximation is executed only once and refinement is executed recursively until a computed solution is accurate enough. The p and q are empirical parameters and we currently set p, q = 4 respectively. The precision for refinement denotes the precision for Step 2 in this paper.

3 RESULTS

TTs were implemented into Mixed IR on an Intel Xeon E5-2650 2GHz core with MKL 2017 and XBLAS 1.0 [3]. Figure 2 shows the impact of TTs on speedup, accuracy, and energy reduction, compared to Mixed IR and Uni-IR. In Fig 2., Trans-IR represents Mixed IR equipped with TTs. We employed uniform random dense matrices for tests and took averages of 10 test cases for runtime and 2 cases for energy measurements. In the top left figure, additional iterations were required to refine **z** by TT 2 (e.g., Trans-IR(Inner Loop)). However, total runtimes were significantly reduced using TTs in the top center figure. Mixed IR runtime becomes shorter than Uni-IR when N=32K thanks to the reduced runtime portion of refinement (refer to the top right figure). TTs brought significant energy reduction to Mixed IR in the bottom-center figure. For measurements, we used the ALEA tool employing constant power model which profiles with Intel Running Average Power Limit (RAPL) and then estimates total energy consumption [5, 6]. The left bottom figure shows the accuracy and runtime trade-off for a matrix of N=32K. The horizontal axis represents runtimes (secs) and the vertical axis represents the log_{10} based relative errors in the solutions. The "vertical" accuracy variation in Trans indicates shorter runtimes for refinement by TT 1. TT 2

SC17, November 2017, Denver, Colorado USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Numerical Properties of IR – refer to Algorithm I for notations

(NP 1) Residual accuracy is almost kept if the mantissa bits for Step 2 are attached as many as cancellation bits per iteration.
(TT 1) Start with double precision arithmetic for Step 2 and switch it to dbl-dbl when the convergence is saturated.
(NP 2) Exact arithmetic yields $\mathbf{z} = A^{-1}(\mathbf{r}+\delta\mathbf{r})$ in Step 3, where $\delta\mathbf{r}$ is a rounding error generated in Step 2. Therefore, $\|A^{-1}\delta\mathbf{r}\|_\infty$ is an irreducible error quantity through refinement of **z**.
(TT 2) Refine **z** using a precision $\epsilon_{ref\text{-}z}$ (e.g., for our case, $\epsilon_{ref\text{-}z} = \epsilon_2^{init}$) when ϵ_2 is a considerably higher precision than ϵ_3 (e.g, smaller $\delta\mathbf{r}$) and a computational latency of ϵ_2 is considerably longer than $\epsilon_{ref\text{-}z}$.
(NP 3) If ϵ_1 = single precision and the refinement of **z** achieves single precision accuracy for **z**, double precision solution accuracy for **x** is guaranteed. We omitted the proof due to the page limitation.
(TT 3) Skip the accuracy check if the conditions of (NP 3) are met. This can save one dbl-dbl matrix-vector multiplication.

Algorithm I. Transprecision Techniques for Mixed IR

```
Step 1: LUPP(A);  LU x = P b; $\epsilon_1$ = double ($2^{-53}$) for Uni-IR, single ($2^{-24}$) for Mixed IR
Step 2: r = b − Ax          $\epsilon_2$ = dbl-dbl ($2^{-106}$)      (TT 1) $\epsilon_2^{init}$ = double
Step 3: LU z = P r          $\epsilon_3 = \epsilon_1$                 (TT 1) if ($\|z\|_\infty/\|z^{prev}\|_\infty > ½$) $\epsilon_2$ = dbl-dbl
  Accuracy Check: If ($\|z\|_\infty/\|x\|_\infty < 2^{-53}$)  exit(Success);   (TT 2) if ($l(\epsilon_2) > p\cdot l(\epsilon_{ref\text{-}z})$ && $\epsilon_2<\epsilon_3^q$ && $\epsilon_2$=dbl-dbl) refine z using double
Step 4:  x = x + z          $\epsilon_4$ = dbl-dbl
  Go to Step 2                                          (TT 3) if (NP3)  exit(Success);
NOTE. $z^{prev}$: z of previous iteration, $\epsilon_i$: a precision for Step i arithmetic, $\epsilon_2^{init}$: a precision applied initially for Step 2 by TT 1
```

Figure 1: Transprecision Techniques for Mixed IR

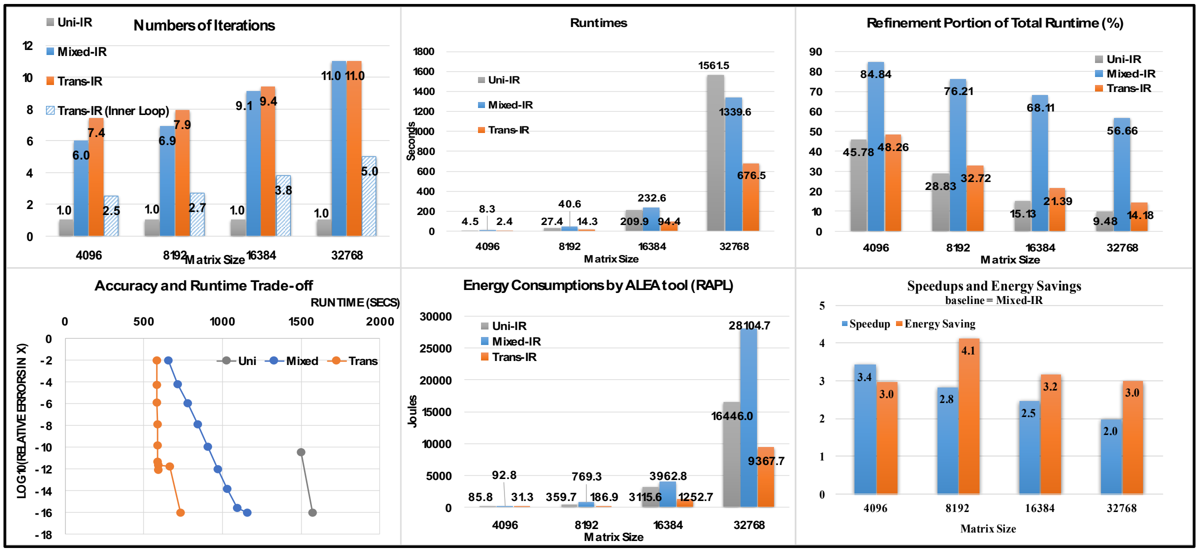

Figure 2: Impact of Transprecision Techniques on Speedups and Energy Savings

enables the accuracy to leap from 10^{-12} to 10^{-16}. Although two dbl-dbl refinements (e.g., ~60 secs for each) appear for Trans due to the accuracy check, TT 3 will remove the second dbl-dbl refinement. A convergence rate of Uni-IR is superior, since it mainly depends on ϵ_1 [4]. TTs brought further 2.0-3.4X speedups and 3.0-4.1X energy reductions to Mixed IR in the bottom right figure. In this paper TTs gave rise to significant speedups and energy savings by minimizing software-emulated precision arithmetic operations.

ACKNOWLEDGMENTS

This project has received funding by the European Commission Horizon 2020 research and innovation programme under grant agreement No. 732631(OPRECOMP).

REFERENCES

[1] J. Langou et al. 2006. Exploiting the Performance of 32 bit Floating Point Arithmetic in Obtaining 64 bit Accuracy (Revisiting Iterative Refinement for Linear Systems). In *SC 2006 Conference, Proceedings of the ACM/IEEE.*
[2] J. Lee. 2012. *AIR: Adaptive Dynamic Precision Iterative Refinement.* Ph.D. Dissertation. University of Tennessee, TN, USA.
[3] X.S. Li et al. 2002. Design, Implementation and Testing of Extended and Mixed Precision BLAS. *ACM Trans. Math. Softw.* 28, 2 (2002), 152–205.
[4] C.B. Moler. 1967. Iterative Refinement in Floating Point. *J. ACM* 14, 2 (1967), 316–321.
[5] L. Mukhanov et al. 2015. ALEA: Fine-Grain Energy Profiling with Basic Block Sampling. In *2015 International Conference on Parallel Architecture and Compilation (PACT).* 87–98.
[6] L. Mukhanov et al. 2017. ALEA: A Fine-Grained Energy Profiling Tool. *ACM Trans. Archit. Code Optim.* 14, 1, Article 1 (2017), 1:1-1:25 pages.