XHYPRE: A high-precision numerical software package for solving large-scale sparse linear equations

Chuanying Li
Hunan University
Changsha, China
lichuanying@hnu.edu.cn

Stef Graillat
Sorbonne Université
Paris, France
stef.graillat@sorbonne-universite.fr

Hao Jiang[1]
National University of Defense Technology
Changsha, China
haojiang@nudt.edu.cn

Zhe Quan
Hunan University
Changsha, China
quanzhe@hnu.edu.cn

ABSTRACT

Due to the inaccurate numerical calculations on high-performance computing platforms, we study how to control the cumulative effect of rounding errors. Specifically, in this paper, we first adopt error-free transformation technology to design high-precision SpMV. Then we design high-precision GMRES, BiCGSTAB, and PCG algorithms. Finally, based on the HYPRE software package, we design and implement a high-precision numerical software package XHYPRE with the high-precision algorithms above for large-scale sparse linear equations. Extensive numerical experiments verify that the final calculation results are more reliable and accurate.

KEYWORDS

rounding errors, error-free transformation techniques, SpMV, GMRES

1 INTRODUCTION

Rounding errors will occur when there are floating-point numbers with a limited number of digits representing real numbers. Rounding error is unavoidable in floating-point operations, and it means the numerical difference between the exact value and the approximate value obtained by floating-point computation.

This paper introduces XHYPRE, a high-precision numerical algorithm library for large-scale sparse linear equations, which is used to solve the rounding error problem of large-scale numerical simulation calculations. We have made these available codes online[2]. First, for the problem that the accuracy of the numerical simulation calculation software decreases, the simulation result is unreliably caused by the inherent error in the floating-point operations. We study the cumulative effect of rounding errors by analyzing the requirements of specific basic module calculation methods for floating-point word-length. Secondly, we design a set of high-precision, efficient and reliable floating-point numerical algorithms by compensating the rounding error in the floating-point calculation process to the original calculation result. Finally, the numerical software library XHYPRE of large-scale sparse linear equations with high-precision is achieved.

[1]Hao Jiang is the author for correspondence.
[2]https://github.com/compilerOpt/-XHYPRE-2.0.0 .

2 HIGH PRECISION XHYPRE

The difference between the exact value and the approximate value obtained by floating-point operations is the rounding error, which causes the instability of the numerical result. On the existing foundation, using the idea of compensation, adding the rounding error back to the original result will significantly improve the accuracy of the calculation. Error-free transformation techniques[1][2][3] become the core tool for the design of compensated algorithms. The TwoSum algorithm is an error-free transformation algorithm for calculating the sum of two floating-point numbers; the TwoProd algorithm is an error-free transformation algorithm for the multiplication of two floating-point numbers. We first used the TwoProd algorithm, but the Fused-Multiply-and-Add (FMA) has higher accuracy and smaller area compared with a separate multiplier and adder. So the combination of FMA and TwoProd algorithm can get a more simplified TwoProdFMA algorithm.

2.1 High Precision SpMV

The execution process of high-precision sparse matrix-vector multiplication is shown in Algorithm 1. Using the high-precision algorithm is the ordinary floating-point result of SpMV that adds the rounding errors.

Algorithm 1 SpMV in XHYPRE(HSpMV)

Input:
$N, A\_value, A\_index, A\_pointer, x$;
Output:
Vector: b;
for $i = 0$ **to** $N-1$ **do**
  Initialize $th = 0, res = 0$;
  for $j = A\_pointer[i]$ **to** $j = A\_pointer[i+1]-1$ **do**
    $[h, l]$ = TwoProdFMA($A\_value[j]$,$x[A\_index[j]]$);
    $[th, tl]$ = TwoSum(th, h);
    res = res + (l + tl);
  end for
  $b_i = res + th$;
end for
return b.

2.2 XHYPRE

Based on HYPRE[4], we analyzed the requirements of specific basic module calculation methods for floating-point word-length, and the realization process of PCG, GMRES, and BiCGSTAB. Then we studied the cumulative effect of rounding errors. Secondly, the algorithm of Dot2 and HSpMV are used to compensate the rounding error to the original key part of the computation. Thus, a set of high-precision and reliable floating-point numerical algorithms is designed. Finally, we realized the high-precision numerical software of large-scale sparse linear equations named XHYPRE.

The following will briefly explain the GMRES algorithm as an example. Algorithm 2 is a high-precision GMRES. The dot product is the Dot2 algorithm in reference[3], and SpMV is the HSpMV algorithm mentioned above. The implementation process of PCG and BiCGSTAB is similar to the GMRES of high-precision, so we do not introduce them in detail.

Algorithm 2 High-precision Generalized Minimal Residual method (HGMRES)

Input:
Matrix A, vector b;
Output:
Approximate solution of linear system $Ax = b$;
1: Compute $r_0 = b - \text{HSpMV}(A, x_0)$, $\beta = \sqrt{\text{Dot2}(r_0, r_0)}$, and $v_1 = r_0/\beta$;
2: **for** $j = 1$ **to** m **do**
3: Compute $w_j = \text{HSpMV}(A, v_j)$;
4: **for** $i = 1$ **to** j **do**
5: $h_{ij} = \text{Dot2}(w_j, v_j)$;
6: $w_j = \text{Dot2}(w_j, h_{ij}v_j)$;
7: **end for**
8: $h_{j+1,j} = \sqrt{\text{Dot2}(w_j, w_j)}$. If $h_{j+1,j} = 0$ set m = j and go to 11
9: $v_{j+1} = w_j/h_{j+1,j}$;
10: **end for**
11: Define the $(m+1) \times$ m Hessenberg matrix $\overline{H}_m = \{h_{ij}\}_{1 \le i \le m+1, 1 \le j \le m}$.
12: Compute y_m the minimizer of $\left\|\beta e_1 - \overline{H}_m y\right\|_2$ and $x_m = x_0 + V_m y_m$
13: **return** x.

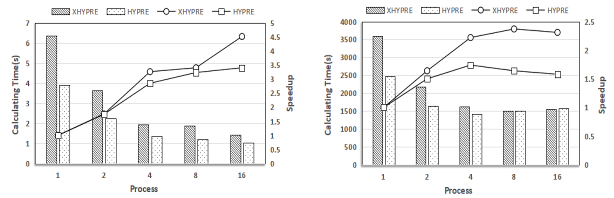
3 EXPERIMENTAL EVALUATION

We conducted tests on the AMD platform, and the results illustrate that XHYPRE is effective. AMD platform includes AMD Ryzen 7 2700X Eight-Core Processor(2.10 GHz). The platform runs the Linux Ubuntu operating system. The codes in the AMD platform were compiled using GCC v8.2.0 and MPICH v3.3.2.

We choose two ill-conditioned matrices for testing. Solverchallenge21_01 is derived from the three-dimensional photon equation of radiation fluid mechanics and is a structural grid. Solverchallenge21_03 is derived from the linear elastic equation of the contact mechanics of the centrifuge device and is a first-order nodal finite element. Table 1 shows the test results. Our XHYPRE can solve the above two linear systems, while the original HYPRE can not.

Table 1: Test results of HYPRE and XHYPRE (number of iterations)

Name	HYPRE	XHYPRE	Remark
solverchallenge21_01	not convergent	4431	without precondition
solverchallenge21_03	not convergent	704	with ILU(0)



(a) (b)

Figure 1: (a)The calculation time and speedup(N=122500).(b) The calculation time and speedup(N=2^{26}).

Although the accuracy of the calculation results has been improved, the calculation time has been increased. Therefore, we use an example to perform a detailed experiment to illustrate the performance of XHYPRE, which can be solved by both HYPRE and XHYPRE. It can be seen from Fig.1(a) that with the parallel computing of multiple processes, the calculation time is also reducing, and XHYPRE is gradually approaching HYPRE. In order to prove the same property when solving large sparse linear algebraic equations, we increased the matrix size to the maximum acceptable value of server memory(32G), N=67108864 (2^{26}). As the running process increases, their number of iterations is the same, the calculation time decreases, and the calculation time of XHYPRE is gradually approaching HYPRE. The test result is Fig.1(b). Therefore, while the accuracy is improved to make the result more accurate, the calculation time of XHYPRE does not increase too much. As the number of nodes increases, the calculation time of XHYPRE decreases faster compared with HYPRE.

4 CONCLUSION

This paper uses error-free transformation techniques to design a high-precision SpMV function. Based on the HYPRE software architecture, we design and implement a high-precision numerical software package XHYPRE for large-scale sparse linear equations. Numerical experiment results show that XHYPRE can effectively improve the accuracy of numerical calculations while ensuring the convergence of the calculation results.

5 ACKNOWLEDGMENTS

This work was supported by the NuSCAP (ANR-20-CE48-0014) project of the French National Agency for Research (ANR), the 173 program (2020-JCJQ-ZD-029), Science Challenge Project (TZ2016002).

REFERENCES

[1] W. Kahan. Pracniques: further remarks on reducing truncation errors. *Communications of the ACM*, 8 (1): 40, 1965.
[2] D. E. Knuth. The art of computer programming. *Pearson Education*, 1997.
[3] T. Ogita, S. Rump, S. Oishi. Accurate sum and dot product. *SIAM Journal on Scientific Computing*, 26(6): 1955-1988, 2005.
[4] HYPRE. *https://computing.llnl.gov/projects/hypre-scalable-linear-solvers-multigrid-methods*