

Methods with Successive and Parallel Approximations of Inverse Operator for the Nonlinear Least Squares Problem

Roman Iakymchuk, Stepan Shakhno

► **To cite this version:**

Roman Iakymchuk, Stepan Shakhno. Methods with Successive and Parallel Approximations of Inverse Operator for the Nonlinear Least Squares Problem. PAMM. Vol. 15. Special issue: 86th Annual Meeting of the International Association of Applied Mathematics and Mechanics (GAMM), Mar 2015, Lecce, Italy. pp.569-570. hal-01175314

HAL Id: hal-01175314

<https://hal.archives-ouvertes.fr/hal-01175314>

Submitted on 10 Jul 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Methods with Successive and Parallel Approximations of Inverse Operator for the Nonlinear Least Squares Problem

Roman Iakymchuk^{1,2*} and Stepan Shakhno^{3**}

¹ Sorbonne Universités, UPMC Univ Paris 06, ICS, 4 place Jussieu, 75005 Paris, France

² Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jussieu, 75005 Paris, France

³ Ivan Franko National University of Lviv, Universitetska Str. 1, 79000 Lviv, Ukraine

For solving the nonlinear least-square problem, we propose iterative methods that use successive and parallel approximations of the inverse operator instead of solving a linear system of equations. The convergence order as well as the convergence radius of the proposed methods are studied. Finally, we carry out numerical experiments on a set of test problems.

Copyright line will be provided by the publisher

1 Introduction

In this paper, we are concerned to find an approximate solution of the nonlinear least squares problem [1]:

$$\min_{x \in R^n} f(x) := \frac{1}{2} F(x)^T F(x), \quad (1)$$

where F is a Fréchet differentiable operator defined on R^n with its values on R^m , $m \geq n$. For solving the problem (1), well-known methods are the Gauss-Newton and Levenberg-Marquardt methods as well as their modifications [1]. At each iteration of these methods, a linear system of equations is solved, which is not always an easy to carry task. Because of that, we propose to use successive and parallel approximations of the inverse operator in order to construct new methods for finding the solution of the problem (1). The idea and the advantage of these approximations is to build the approximate solution of the problem (1) without solving a linear system of algebraic equations.

Our study is focused on analyzing the convergence of the derived methods as well as on parallel implementations of them. We investigate the local convergence of the successive and two parallel methods with approximations of the inverse operator under the classic Lipschitz conditions. In addition, we examine the convergence order of the proposed methods. As numerical experiments, we carry out a set of standard tests, comparing the proposed methods against the well-known Gauss-Newton method and its modification.

We foresee to apply parallelization techniques – such as a parallel approximation of the inverse operator – to the other methods. In particular, we plan to study the two-step modification of the Gauss-Newton method [2] as it uses only one inverse over two function evaluations.

2 Methods with Successive and Parallel Approximations and their Convergence

Let us consider the Gauss-Newton type method with the successive approximation

$$x_{k+1} = x_k - A_k J(x_k)^T F(x_k), \quad (2)$$

$$A_{k+1} = A_k [2E - J(x_{k+1})^T J(x_{k+1}) A_k], \quad k = 0, 1, 2, \dots \quad (3)$$

where $J(x) = F'(x)$; E is an identity matrix; x_0 and A_0 are given initial approximations to the exact solution x^* and to the inverse operator $A^* = [J(x^*)^T J(x^*)]^{-1}$, accordingly. The method (2)-(3) consists of two branches, meaning computations of x_{k+1} and A_{k+1} , that are executed by turns. The next theorem provides convergence analysis of the method (2)-(3) in case of zero residual $F(x_*) = 0$.

Theorem 2.1 *Let us assume that the problem (1) has the solution x^* and there exists $A^* = [J(x^*)^T J(x^*)]^{-1}$ such that $\|A^*\| \leq B$ and $\max\{\|J(x^*)\|, \|J(x^*)^T\|\} \leq C$. In the domain $\Omega(x_*, r_0) = \{x : \|x - x_*\| \leq r_0\}$, the estimate $\|J(x)\| = \|F'(x)\| \leq L$ and the classic Lipschitz condition $\|J(x)^T - J(x^*)^T\| \leq L\|x - x^*\|$ hold. Additionally, $h_0 = \max\{K_0, C^2 + (B + r_0)^2 L K_0 (2C + L r_0)\} \leq 1/r_0$ holds, where $r_0 = \max\{\|x_0 - x^*\|, \|A_0 - A^*\|\}$ and $K_0 = C^2 + (B + r_0)[C(2 + L/2) + L(1 + L/2)r_0]$.*

Then, sequences $\{x_k\}$ and $\{A_k\}$ converge to x^ and A^* , accordingly, satisfying the following estimate*

$$r_k = \max\{\|x_k - x^*\|, \|A_k - A^*\|\} \leq (h_0 r_0)^{2^k - 1} r_0, \quad k = 0, 1, 2, \dots$$

* Corresponding author: e-mail: roman.iakymchuk@lip6.fr, Phone: +33 (0)1 44 27 88 76

** e-mail: s.shakhno@franko.lviv.ua, Phone: +38 03 22 39 43 04

For solving the problem (1) on shared memory architectures, when multiple cores of one or many CPU share the same memory, we derive a parallel variant of the Gauss-Newton type method (2)-(3)

$$x_{k+1} = x_k - A_k J(x_k)^T F(x_k), \quad (4)$$

$$A_{k+1} = A_k [2E - J(x_k)^T J(x_k) A_k], \quad k = 0, 1, 2, \dots \quad (5)$$

The method (4)-(5) as well as the method (2)-(3) consists of two brunches, however these branches can be processed in parallel. In addition, computations at each branch can be parallelized using the standard implementations from linear algebra libraries such as Basic Linear Algebra Subprograms (BLAS) and Linear Algebra PACKage (LAPACK).

Although the method (4)-(5) is suitable for parallelization, its convergence order is decreased to 1, 618.... Moreover, due to the different amount of calculations in formulas (4) and (5), there is a load imbalance that may idle threads/CPU's. Despite inefficient usage of computing resources, the total computation time should be reduced because of the parallelization applied.

In order to avoid the above-mentioned shortcomings, we accomplish to derive the computational process, which does not require synchronization of computations among different processors. Such approach results in the asynchronous version of the method (4)-(5)

$$x_{k+1}^{m+1} = x_{k+1}^m - A_k J(x_k^{m_k})^T F(x_{k+1}^m), \quad m = 0, 1, \dots, m_{k+1} - 1, \quad (6)$$

$$A_{k+1} = A_k [2E - J(x_k^{m_k})^T J(x_k^{m_k}) A_k], \quad k = 0, 1, 2, \dots \quad (7)$$

where $x_k^{m_k}$ is the last approximation to the exact solution x^* of the problem (1) for computing which we apply the approximation of the inverse operator A_{k-1} ; $x_{k+1}^0 = x_k = x_k^{m_k}$; both x_0 and A_0 are given initial approximations to x^* and $A^* = [J(x^*)^T J(x^*)]^{-1}$. Since the iterative process (6)-(7) uses asynchronous approximation of the inverse operator, its execution on shared memory architectures will not stand idle. Moreover, the convergence order of the method (6)-(7) is 2.

Due to the space limit, only the convergence analysis of the method (2)-(3) is presented in this article. The other theorems and corollaries as well as proofs will be included in the extended paper, which will be finished in the near future.

3 Numerical Results

We compare the proposed methods against the Gauss-Newton method and its two-step modification in terms of the execution time and the number of iterations to the solution of the Extended Rosenbrock function ($x_0 = \{\xi_j\}, \xi_{2j-1} = 1 - 10^{-2}, \xi_{2j} = 1$), see Tab. 1. We provide sequential implementations of the first two methods and parallel for the rest.

Table 1: Numerical results of the Extended Rosenbrock function ($n = 1000$) on an 8-core Intel Xeon E5-4650L (Sandy Bridge).

Method	$\epsilon = 1e - 5$		$\epsilon = 1e - 15$	
	Iterations	Time	Iterations	Time
Gauss-Newton	3	0.391930	3	0.391887
Two-step modification of the Gauss-Newton m-d [2]	2	0.263885	3	0.393943
M-d with successive approximation (2)-(3)	4	0.268449	6	0.381806
M-d with synchronous parallel approximation (4)-(5)	5	0.323528	7	0.439239
M-d with asynchronous parallel approximation (6)-(7)	1	0.229656	2	0.333703

To sum up, the method (6)-(7) with asynchronous parallel approximation of the inverse operator runs faster than the classic methods in terms of both the number of iterations and the execution time. Moreover, it is well-suited in case the inverse operator is either difficult or impossible to compute.

Acknowledgements This work undertaken (partially) in the framework of CALSIMLAB is supported by the public grant ANR-11-LABX-0037-01 overseen by the French National Research Agency (ANR) as part of the ‘‘Investissements d’Avenir’’ program (reference: ANR-11-IDEX-0004-02). This work was also granted access to the HPC resources of ICS financed by Region Île-de-France and the project Equip@Meso (reference ANR-10-EQPX-29-01) overseen by ANR as part of the ‘‘Investissements d’Avenir’’ program.

References

- [1] J. M. Dennis and R. B. Schnabel. Prentice-Hall, New York, 1983.
- [2] R. Iakymchuk and S. Shakhno. In *PAMM. Vol. 14. Special issue: The 85th Annual Meeting of the International Association of Applied Mathematics and Mechanics (GAMM). Erlangen, Germany, 2014*, pages 813–814.