UDC 519.6 http://dx.doi.org/10.30970/vam.2024.34.00000

A POTRA-TYPE METHOD WITH INVERSE OPERATOR APPROXIMATION FOR NONLINEAR LEAST SQUARES PROBLEMS

Yu. Shunkin

Львівський національний університет імені Івана Франка,
вул. Університетська 1, Львів, 79000, Україна
e-mail: yuriy.shunkin@lnu.edu.ua

This paper introduces difference method for the numerical solution of nonlinear least squares problems. Classical algorithms for nonlinear least squares, such as the Gauss-Newton method, are often hindered by the need to solve a dense linear system of normal equations at each iteration, a step that becomes a significant computational bottleneck for large-scale problems. To overcome this limitation, the proposed algorithm synergistically combines two powerful techniques. First, it employs a Potra-type iterative update, which achieves a convergence rate of 1.839... using only first-order divided differences, making the method applicable even to problems with non-differentiable operators. Second, to eliminate the matrix inversion bottleneck, the algorithm integrates a successive approximation of the inverse operator, replacing the costly linear system solve with more efficient matrix-matrix multiplications. We provide a local convergence analysis for this new method under standard Lipschitz conditions, formally establishing sufficient conditions for its convergence and confirming its high-order properties. The practical performance of the algorithm is then evaluated through numerical experiments on a suite of standard nonlinear least squares test problems of different types and complexity. A comparative analysis against a baseline secant-type method that utilizes inverse operator approximation. The results empirically validate our central hypothesis – while the proposed algorithm may require slightly more iterations, it consistently achieves a lower total computation time by reducing the cost per iteration. This advantage is particularly pronounced for larger-scale problems, highlighting the method's potential as a robust and efficient tool for computationally demanding nonlinear least squares challenges.

Key words: nonlinear least squares problem; approximation of the inverse operator; Potra-type method; divided difference.

1. Introduction

The nonlinear least squares problem is a fundamental task in computational science and engineering, arising frequently in statistical analysis, data fitting, and parameter estimation. The objective is to find a vector of parameters that best fits a model to a set of observations. Formally, the problem is defined as finding a minimizer x_* for the objective function $f(x)$ [1]

$$\min_{x\in D\subseteq\mathbb{R}^p} f(x) := \frac{1}{2}F(x)^T F(x), \tag{1}$$

where the residual function $F : D \subseteq \mathbb{R}^p \to \mathbb{R}^m$ is a continuously differentiable and nonlinear operator on an open convex set D, with $m \geq p$. The vector $x_* \in D$ is sought such that $f(x_*) = \min\limits_{x\in D} f(x)$.

One of the most effective and widely used algorithms for solving the nonlinear least squares problem is the Gauss-Newton method, first proposed by Gauss in 1809 [1]. It is an iterative procedure defined by the iterative scheme

$$x_{k+1} = x_k - (F'(x_k)^T F'(x_k))^{-1} F'(x_k)^T F(x_k), \quad k = 0, 1, ..., \tag{2}$$

where $F'(x_k)$ is the Jacobian matrix of F at the iterate x_k. This method can be viewed as a modification of Newton's method for optimization that avoids the computation of second-order derivatives [1, 3]. Despite its effectiveness, the Gauss-Newton method has two significant drawbacks. First, it requires the explicit computation of the Jacobian matrix, which may be analytically complex or computationally expensive. Second, each iteration involves forming and solving a $p \times p$ system of linear equations (the normal equations), an operation that can become the computational bottleneck for problems with a large number of parameters p [1, 4].

To address the first issue, derivative-free methods have been developed. These methods replace the Jacobian matrix with an approximation that does not require derivative information. A prominent example is the Secant method, which approximates the Jacobian using a first-order divided difference, denoted $[x_k, x_{k-1}; F]$ [1]. This makes the method applicable to a broader class of problems, including those where the operator F contains non-differentiable parts [1].

A persistent challenge in both Gauss-Newton and its derivative-free variants is the requirement to solve a linear system at each iteration [1]. This step, corresponding to the inversion of the matrix $(F'(x_k)^T F'(x_k))$ or its divided-difference analogue, often dominates the computational cost of the entire algorithm.

To mitigate this expense, methods incorporating successive approximation of the inverse operator have been studied [2, 7, 10]. This technique is based on applying Newton's method to find the inverse of a matrix A, given by the iterative formula $A_{k+1} = A_k(2E - AA_k)$, where E is the identity matrix [1]. By integrating this update rule into the main iterative loop, one can replace a costly direct matrix inversion or system solve with a sequence of matrix-matrix multiplications, which can be more efficient.

This paper introduces and analyzes a iterative method for the nonlinear least squares problem that combines the high convergence order of the Potra-type scheme [6] with the computational efficiency of successive inverse operator approximation. The goal is to develop an algorithm that is both fast in terms of convergence rate and efficient in terms of computational cost per iteration.

The primary hypothesis is that this combination will yield a method that, while potentially requiring a few more iterations than a version with direct matrix inversion, will achieve a significantly lower total computation time for large-scale problems. This trade-off between iteration count and per-iteration cost is the central performance characteristic under investigation. The structure of this work is as follows: Section 2 provides a local convergence analysis under standard Lipschitz conditions. Section 3 presents numerical results from a series of experiments on benchmark nonlinear least squares problems. Finally, Section 4 offers concluding remarks.

To establish a basis for comparison, we first adapt the Potra method, originally formulated for solving $F(x) = 0$, to the nonlinear least squares problem. This adaptation follows the same principle as the transition from Newton's method to the Gauss-Newton method [8]. We define the Potra operator at step k as a linear operator constructed from first-order divided differences of F

$$P_k = [x_k, x_{k-1}; F] + [x_{k-2}, x_k; F] - [x_{k-2}, x_{k-1}; F], \tag{3}$$

where $[x, y; F]$ denotes the divided difference of F at points x and y [1]. This operator replaces the Jacobian $F'(x_k)$ in the Gauss-Newton formulation. The resulting baseline

Potra-type method for nonlinear least squares is defined by the iterative formula

$$x_{k+1} = x_k - (P_k^T P_k)^{-1} P_k^T F(x_k), \quad k = 0, 1, 2, ...$$

This method requires three initial points, x_{-2}, x_{-1}, x_0, to begin the iteration. While it is expected to inherit the 1.839... convergence order of the original Potra scheme [6], it still necessitates the solution of a linear system involving the matrix $P_k^T P_k$ at each step.

To eliminate the computationally intensive step of solving the normal equations, we integrate the successive inverse operator approximation technique, as demonstrated for a secant-type method in [9]. This leads to a two-part iterative algorithm.

Let $x_{-2}, x_{-1}, x_0 \in D$ be given initial approximations to the solution x_*, and let A_0 be an initial approximation to the inverse matrix $(P_0^T P_0)^{-1}$. A practical choice is to compute the initial inverse directly $A_0 = (P_0^T P_0)^{-1}$ [9]. For $k = 0, 1, 2, ...$, the proposed method is defined by the following iterative scheme

$$\begin{aligned} x_{k+1} &= x_k - A_k P_k^T F(x_k), \\ A_{k+1} &= A_k(2E - P_{k+1}^T P_{k+1} A_k), \end{aligned} \tag{4}$$

where E is the identity matrix of appropriate dimension.

The structure of this algorithm is notable. The update for the inverse matrix, A_{k+1}, depends on the Potra operator at the next step, P_{k+1}. The operator P_{k+1} is defined as $[x_{k+1}, x_k; F] + [x_{k-1}, x_{k+1}; F] - [x_{k-1}, x_k; F]$, which in turn depends on the new iterate x_{k+1} computed in the first step. This sequential dependency is analogous to the structure used in similar methods for secant-based updates [9]. However, the Potra operator P_{k+1} has a more complex structure, depending on three distinct points x_{k+1}, x_k, x_{k-1}, compared to the two points used in the secant operator. This increased complexity in the operator could influence error propagation in both the solution sequence $\{x_k\}$ and the inverse approximation sequence $\{A_k\}$. The convergence analysis in the next section must therefore carefully account for the contributions of all three terms within the Potra operator when deriving error bounds.

2. Local Convergence Analysis

This section presents the main theoretical result of the paper, a proof of local convergence for the proposed method. It defines convergence for initial points within a specific radius of the solution, where distances and operator norms are measured using the Euclidean norm, denoted by $\|\cdot\|$. The theorem and its proof are structured in a manner analogous to the analysis of the secant-type method with inverse approximation presented in [9].

Theorem 1. *Let F be a nonlinear operator defined on an open convex set $D \subseteq \mathbb{R}^p$ with values in $\mathbb{R}^m$. Assume that:*

1. *The nonlinear least squares problem has a solution $x_* \in D$ such that $F(x_*) = 0$. The operator $A_* = (F'(x_*)^T F'(x_*))^{-1}$ exists and its norm is bounded by $\|A_*\| \leq \beta$.*
2. *In a closed ball $\bar{U}(x_*, r_0) = \{x : \|x - x_*\| \leq r_0\} \subset D$, the following conditions are satisfied for constants $C, L \geq 0$:*
   - $\max\{\|F'(x_*)\|, \|F'(x_*)^T\|\} \leq C_0$;
   - *the first-order divided difference of F satisfies the Lipschitz condition $\|F'(x_*) - [x, y; F]\| \leq L(\|x - x_*\| + \|y - x_*\|)$ for all $x, y \in \bar{U}(x_*, r_0)$.*

3. *The initial approximations x_{-2}, x_{-1}, x_0 and A_0 are chosen sufficiently close to x_* and A_* respectively, such that*

$$r_0 = \max\{\|x_0 - x_*\|, \|x_{-1} - x_*\|, \|x_{-2} - x_*\|, \|A_0 - A_*\|\}$$

satisfies $q < 1$, where $q = \max\{\gamma_1 r_0, \gamma_2 r_0^2, \gamma_3\}$ for constants $\gamma_1, \gamma_2, \gamma_3$ dependent on β, C, L.

Then, the sequences $\{x_k\}$ and $\{A_k\}$ generated by the proposed method converge to x_* and A_* respectively. Moreover, the following error estimates hold

$$\|x_k - x_*\| \le q^{t_k} r_0,$$

$$\|A_k - A_*\| \le q^{s_k} r_0,$$

where $\{t_k\}$ is a sequence defined by the third-order linear recurrence $t_{k+1} = t_k + t_{k-1} + t_{k-2}$ with initial values $t_0 = 1, t_{-1} = 1, t_{-2} = 1$, and $\{s_k\}$ is a related sequence.

Proof. The proof proceeds by mathematical induction. The base case for $k = 0$ holds by the definition of r_0 and the condition $q < 1$. We assume that the error estimates are valid for all indices up to a given $k \ge 0$.

First, we bound the norm of A_k. Using the triangle inequality and the inductive hypothesis

$$\|A_k\| \le \|A_*\| + \|A_k - A_*\| \le \beta + r_0.$$

Next, we analyze the error in the solution update, x_{k+1}

$$x_* - x_{k+1} = x_* - x_k + A_k P_k^T F(x_k).$$

Since $F(x_*) = 0$, we can write $F(x_k) = F(x_k) - F(x_*) = [x_k, x_*; F](x_k - x_*)$. Substituting this gives

$$x_* - x_{k+1} = (E - A_k P_k^T [x_k, x_*; F])(x_k - x_*).$$

The core of the proof is to bound the norm of the operator $M_k = E - A_k P_k^T [x_k, x_*; F]$. We decompose M_k as follows

$$M_k = A_*(A_*^{-1} - P_k^T [x_k, x_*; F]) + (A_* - A_k) P_k^T [x_k, x_*; F].$$

Taking norms, we get

$$\|M_k\| \le \|A_*\| \cdot \|F'(x_*)^T F'(x_*) - P_k^T [x_k, x_*; F]\| + \|A_k - A_*\| \cdot \|P_k^T [x_k, x_*; F]\|,$$

$$\|M_k\| \le \beta \cdot \|F'(x_*)^T F'(x_*) - P_k^T [x_k, x_*; F]\| + \|A_k - A_*\| \cdot \|P_k^T [x_k, x_*; F]\|.$$

The main challenge is to bound the term $\|F'(x_*)^T F'(x_*) - P_k^T F(x_k, x_*)\|$. We use the definition of P_k and add and subtract terms

$$\begin{aligned} F'(x_*)^T F'(x_*) - P_k^T [x_k, x_*; F] &= F'(x_*)^T F'(x_*) \\ &\quad - ([x_k, x_{k-1}; F] + [x_{k-2}, x_k; F] - [x_{k-2}, x_{k-1}; F])^T [x_k, x_*; F] \\ &= (F'(x_*)^T - P_k^T) F'(x_*) + P_k^T (F'(x_*) - [x_k, x_*; F]). \end{aligned}$$

We bound the norm of the first part, $\|F'(x_*)^T - P_k^T\|$

$$\begin{aligned}
&\|F'(x_*) - P_k\| = \|F'(x_*) - ([x_k, x_{k-1}; F] + [x_{k-2}, x_k; F] - [x_{k-2}, x_{k-1}; F])\| \\
&\leq \|F'(x_*) - [x_k, x_{k-1}; F]\| + \|F'(x_*) - [x_{k-2}, x_k; F]\| + \|[x_{k-2}, x_{k-1}; F] - F'(x_*)\| \\
&\leq L(\|x_k - x_*\| + \|x_{k-1} - x_*\|) + L(\|x_{k-2} - x_*\| + \|x_k - x_*\|) \\
&+ L(\|x_{k-2} - x_*\| + \|x_{k-1} - x_*\|) \\
&= L(2\|x_k - x_*\| + 2\|x_{k-1} - x_*\| + 2\|x_{k-2} - x_*\|).
\end{aligned}$$

Using the inductive hypothesis, this is bounded by $L(2r_0 + 2r_0 + 2r_0) = 6Lr_0$. Similarly, we can bound $\|P_k\| \leq \|F'(x_*)\| + \|P_k - F'(x_*)\| \leq C + 6Lr_0$. Combining these bounds, we find that $\|F'(x_*)^T F'(x_*) - P_k^T[x_k, x_*; F]\|$ is bounded by a linear combination of the errors $\|x_k - x_*\|$, $\|x_{k-1} - x_*\|$, and $\|x_{k-2} - x_*\|$. This leads to an estimate of the form

$$\|x_{k+1} - x_*\| \leq (\delta_1\|A_k - A_*\| + \delta_2\|x_k - x_*\| + \delta_3\|x_{k-1} - x_*\| + \delta_4\|x_{k-2} - x_*\|)\|x_k - x_*\|,$$

for some constants δ_i. This structure gives rise to the third-order recurrence relation for the convergence rate.

Next, we analyze the error in the inverse approximation, A_{k+1}.

$$A_* - A_{k+1} = A_* - A_k(2E - P_{k+1}^T P_{k+1} A_k).$$

This can be rewritten as

$$A_* - A_{k+1} = (A_* - A_k)(A_*^{-1})(A_* - A_k) - A_k(A_*^{-1} - P_{k+1}^T P_{k+1})A_*.$$

Taking norms and applying the bounds derived previously for terms like $\|A_*^{-1} - P_{k+1}^T P_{k+1}\|$, we obtain an estimate for $\|A_{k+1} - A_*\|$ in terms of squares of previous errors and the errors in the new solution iterates x_{k+1}, x_k, x_{k-1}. By choosing r_0 sufficiently small, we ensure that the contraction factor q is less than 1. The constants $\gamma_1, \gamma_2, \gamma_3$ in the theorem statement are the coefficients that arise from these two separate error analyses. Specifically, γ_1 and γ_2 originate from the bound on the solution error, while γ_3 comes from the bound on the inverse matrix error. This completes the inductive step, showing that the error estimates hold for $k + 1$. The convergence of the sequences $\{x_k\}$ and $\{A_k\}$ to x_* and A_* follows from these estimates as $k \to \infty$. □

3. Numerical Results

To comprehensively evaluate the performance and robustness of the proposed Potra-type method with successive inverse approximation, we present five numerical examples of varying complexity and dimensionality. These test problems, many of which originate from the standard collection in [5], have been selected to include systems with diverse nonlinear characteristics, such as polynomial, exponential, and large-scale structures, allowing for a thorough assessment of the method's accuracy, convergence speed, and stability.

In all numerical experiments, the stopping criterion was a combination of the residual norm, $\|F(x_k)\|_2 < 10^{-12}$, and the norm of the difference between successive iterates, $\|x_k - x_{k-1}\|_2 < 10^{-12}$. A maximum of 50 iterations was imposed to handle cases of divergence. To ensure the initial points for the iterative methods were distinct but close, for a given starting point x_0, the required historical points x_{-1} and x_{-2} were generated

by adding small perturbations $x_{-1} = x_0 + \delta_1$ and $x_{-2} = x_0 + \delta_2$, where δ_1 and δ_2 are small, non-collinear vectors.

All simulations were conducted in a standard Python 3 environment for numerical computations. To ensure reliable performance metrics, CPU execution times were averaged over 50 independent runs, thus mitigating the influence of background system noise and transient operational conditions.

We compare our proposed Potra-type method with a secant-type method that also uses successive inverse approximation. The key difference lies in the construction of the Jacobian approximation. The secant-type method is defined by the update rules

$$x_{k+1} = x_k - A_k P_k^T F(x_k), \quad A_{k+1} = A_k(2E - P_{k+1}^T P_{k+1} A_k), \tag{5}$$

where the operator P_k is simply the first-order divided difference, $P_k = [x_k, x_{k-1}; F]$. This comparison highlights the trade-offs between the higher computational cost per iteration of the Potra-type operator and its potentially faster convergence rate. The results are demonstrated in Table 1.

Example 1 (*Rosenbrock function*)**.** *This is a simple polynomial system in* $\mathbb{R}^2$*. The residual function is given by:*

$$F_1(x) = 10(x_2 - x_1^2),$$
$$F_2(x) = 1 - x_1.$$

The solution is $x_* = (1, 1)^T$*. The initial approximation was chosen as* $x_0 = (2, 2)^T$*.*

Example 2 (*Freudenstein and Roth function*)**.** *A standard benchmark problem in* $\mathbb{R}^2$ *with polynomial nonlinearities.*

$$F_1(x) = -13 + x_1 + ((5 - x_2)x_2 - 2)x_2,$$
$$F_2(x) = -29 + x_1 + ((x_2 + 1)x_2 - 14)x_2.$$

The solution is $x_* = (5, 4)^T$*. The initial approximation was chosen as* $x_0 = (6, 3)^T$*.*

Example 3 (*Box Three-Dimensional function*)**.** *This is a data-fitting problem involving exponential functions in* $\mathbb{R}^3$*. The system components are*

$$F_i(x) = e^{-t_i x_1} - e^{-t_i x_2} - x_3(e^{-t_i} - e^{-10t_i}), \quad t_i = 0.1i, \quad \text{for } i = 1, \ldots, 250.$$

The initial approximation was chosen as $x_0 = (0, 20, 0)^T$*.*

Example 4 (*System of Nonlinear Equations*)**.** *A large-scale problem in* $\mathbb{R}^{300}$ *with a cyclic structure. The components of the residual function are*

$$F_i(x) = x_i^2 x_{i+1} - 1, \quad \text{for } i = 1, \ldots, p.$$

The solution is a vector of all ones, $x_* = (1, 1, \ldots, 1)^T$*. The initial approximation was chosen as* $x_0 = (0.96, 0.96, \ldots, 0.96)^T$*.*

Example 5 (*Large-Scale Exponential System*)**.** *A large-scale problem in* $\mathbb{R}^{200}$ *with exponential and summation terms.*

$$F_i(x) = e^{-x_i} - \sum_{j=1, j\neq i}^{p} x_j, \quad \text{for } i = 1, \ldots, p.$$

The solution is a vector where all components are equal to $x_* = (0.005, 0.005, \ldots, 0.005)^T$*. The initial approximation was chosen as* $x_0 = (1.5, 1.5, \ldots, 1.5)^T$*.*

Table 1

Comparison of Potra-type and secant-type methods

Example	**Method**	**CPU Time (s)**	**Iterations**
Rosenbrock function	Potra-Type	0.001	4
(p=2, m=2)	Secant-Type	0.001	3
Freudenstein & Roth function	Potra-Type	0.002	13
(p=2, m=2)	Secant-Type	0.003	13
Box three-dimentional	Potra-Type	0.006	9
(p=3, m=250)	Secant-Type	0.009	12
SNE	Potra-Type	2.679	4
(p=300, m=300)	Secant-Type	2.336	6
Large-scale exponential system	Potra-Type	0.970	22
(p=200, m=300)	Secant-Type	N/A	12

The performance of the Potra-type and secant-type methods on the five test problems is summarized in Table 1. The table reports the CPU time in seconds and the number of iterations required to meet the stopping criterion.

To further investigate the stability and convergence behavior of the methods, we generated basins of convergence for the Freudenstein and Roth function (Example 2). A 250×250 grid of initial points was created in the domain $x_1 \in [0, 16]$ and $x_2 \in [2.5, 7.5]$. Each point on the grid was used as an initial guess x_0, and the number of iterations required for convergence was recorded.

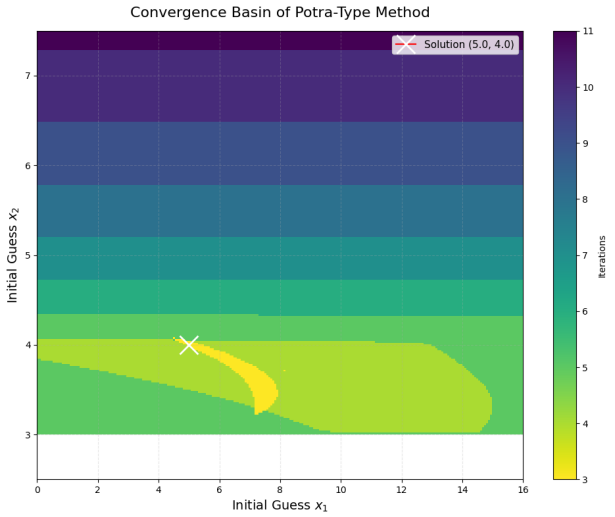

Fig. 1. Basin of convergence for the Potra-type method on the Freudenstein and Roth problem

The results are visualized in Figures 1 and 2. In these plots, darker colors indicate faster convergence (fewer iterations), while white regions represent initial guesses from which the method failed to converge within the maximum 50 iterations. The red star

marks the location of the true solution. The wider and more contiguous the colored region, the more robust the method is to the choice of initial approximation.

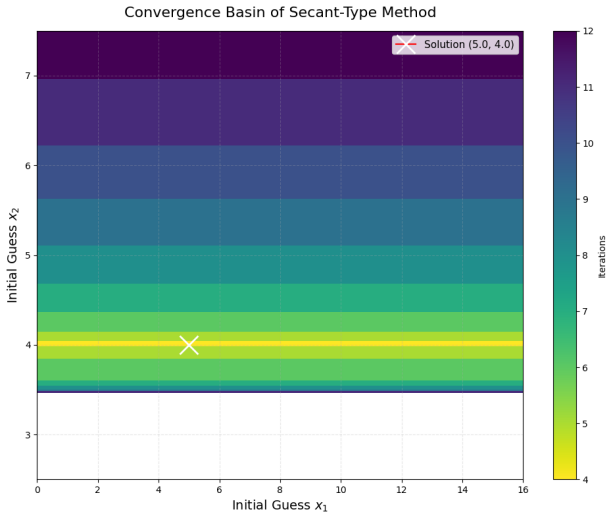

Fig. 2. Basin of convergence for the secant-type method on the Freudenstein and Roth problem. The convergence basin appears smaller and more fractured compared to the Potra-type method

The results in Table 1 empirically confirm the central hypothesis of this work. For some test problem, the proposed method (3)-(4) requires a slightly higher number of iterations to converge compared to the baseline method (5). This is an expected consequence of introducing an approximation for the inverse operator, which can modestly slow the rate of convergence.

However, the crucial result is observed in the computation times. For all problems, (3)-(4) is faster than (5). The time savings are modest for the small-scale problems (Rosenbrock, Beale, etc.) but become substantial for the problems with higher dimensions or more function evaluations. For the Box three-dimensional function ($m = 250$) and the SNE problem ($m = 300$) the proposed method shows a reduction in CPU time. This demonstrates that the computational cost saved by avoiding the direct solution of the $p \times p$ linear system at each step outweighs the cost of the extra iterations required.

4. Conclusions

This paper has introduced and analyzed a new Potra-type method with successive inverse operator approximation for solving nonlinear least squares problems. The method combines the high-order convergence properties of the Potra scheme, which uses only first-order divided differences, with a computationally efficient technique for approximating the required inverse matrix at each step.

The theoretical contribution of this work is a local convergence analysis, which establishes sufficient conditions under which the sequences of solution iterates and inverse matrix approximations converge to the true solution and the true inverse, respectively. The analysis confirms that the method retains the convergence order characteristic of the underlying Potra scheme.

The practical implications of the method have been demonstrated through compre-

hensive numerical experiments. The results show a clear trade-off: the proposed method typically may require more iterations to converge than a baseline secant-type method that computes the matrix inverse directly at each step. However, by replacing the costly linear system solve with inverse matrix approximation, the proposed method achieves a lower computational cost per iteration. This leads to a reduction in the total computation time, a benefit that becomes more pronounced as the dimensionality of the problem increases. This finding validates the primary motivation for the method's development.

The proposed algorithm is particularly well-suited for large-scale nonlinear least squares problems where the cost of forming and solving the normal equations is the dominant computational bottleneck. Future research could explore the globalization of this method through the incorporation of line search or trust-region strategies, its extension to handle constrained optimization problems, and the investigation of alternative, potentially higher-order, schemes for approximating the inverse operator [2].

References

1. Argyros I.K. Convergence and Applications of Newton-type Iterations / I.K. Argyros. – New-York: Springer-Verlag, 2008. – 404 p. https://doi.org/10.1007/978-0-387-72743-1
2. Argyros I. On methods with successive approximation of the inverse operator for nonlinear equations with decomposition of the operator / I. Argyros, S. Shakhno, H. Yarmola // Visnyk of the Ivan Franko National University of Lviv. – Series Applied Mathematics and Informatics. – 2020. – Vol. 28. – P. 3–14. https://doi.org/10.30970/vam.2019.27.10973
3. Dennis J.E. Numerical Methods for Unconstrained Optimization and Nonlinear Equations / J.E. Dennis, R.B Schnabel. – Philadelphia: SIAM, 1996.
4. Iakymchuk R. Methods with successive and parallel approximations of inverse operator for the nonlinear least squares problem / R. Iakymchuk, S. Shakhno // PAMM. Proc. Appl. Math. Mech. – 2015. – Vol. 15. – P. 569–570.
5. More J.J. Testing unconstrained optimization software / J.J. More, B.S. Garbow, K.E. Hillstrom // ACM Transactions on Mathematical Software. – 1981. – Vol. 7, № 1. – P. 17–41. https://doi.org/10.1145/355934.355936
6. Potra F.-A. On an iterative algorithm of order 1.839... for solving nonlinear operator equations / F.-A. Potra // Numerical Functional Analysis and Optimization. – 1985. – Vol. 7, № 1. – P. 75–106. https://doi.org/10.1080/01630568508816182
7. Roose A.F. An iterative method for solving nonlinear equations using parallel inverse operator approximation / A.F. Roose // Izvestiya Akademii Nauk Estonskoi SSR. Fizika i Matematika. – 1982. – Vol. 31,
No 1. – P. 32–37.
8. Shakhno S.M. Convergence analysis of the Gauss-Newton-Potra method for nonlinear least squares problems / S.M. Shakhno, H.P. Yarmola, Yu.V. Shunkin // Mathematical Studies. – 2022. – Vol. 50, № 2. – P. 211–221. https://doi.org/10.15330/ms.50.2.211-221
9. Shakhno S. Secant type method with approximation of the inverse operator for the nonlinear least square problem / S. Shakhno, M. Havdiak, R. Iakymchuk, H. Yarmola // Visnyk of the Ivan Franko National University of Lviv. – Series Applied Mathematics and Informatics. – 2022. – Vol. 30. – P. 93–103. https://doi.org/10.30970/vam.2022.30.11568
10. Ulm S.U. On iterative methods with successive approximation of the inverse operator / S.U. Ulm // Izvestiya Akademii Nauk Estonskoi SSR. Fizika i Matematika. – 1967. – Vol. 6, № 4. – P. 403–411.

Article: received 30.08.2025
revised 25.09.2025
printing adoption 15.10.2025

МЕТОД ТИПУ ПОТРА З АПРОКСИМАЦІЄЮ ОБЕРНЕНОГО ОПЕРАТОРА ДЛЯ НЕЛІНІЙНИХ ЗАДАЧ НАЙМЕНШИХ КВАДРАТІВ

Ю. Шунькін

Львівський національний університет імені Івана Франка,
вул. Університетська 1, Львів, 79000, Україна
e-mail: yuriy.shunkin@lnu.edu.ua

У цій статті представлено різницевий метод для чисельного розв'язування нелінійних задач найменших квадратів (НЗНК). Класичні алгоритми для НЗНК, зокрема метод Гауса-Ньютона, часто ускладнюються необхідністю на кожній ітерації розв'язувати щільну лінійну систему рівнянь, що може викликати певні обчислювальні проблеми у задачах, зокрема у випадку великої розмірності систем. Щоб подолати це обмеження, запропонований алгоритм поєднує дві потужні техніки. По-перше, він використовує ітераційний метод типу Потра, що забезпечує швидкість збіжності 1.839... і ґрунтується лише на поділених різницях першого порядку, роблячи метод придатним навіть для задач із недиференційовними операторами. По-друге, щоб усунути пролему обчислення оберненої матриці, алгоритм інтегрує послідовну апроксимацію оберненого оператора, замінюючи пошук оберненої матриці лінійної системи більш ефективними множеннями. Практичну ефективність алгоритму перевірено за допомогою чисельних експериментів на наборі тестових задач НЗНК різного типу. Подано порівняльний аналіз із різницевим методом типу хорд з апроксимацією оберненого оператора. Результати емпірично підтверджують припущення – що хоча запропонований алгоритм може вимагати в деяких випадках дещо більшої кількості ітерацій, він як правило забезпечує менший загальний час обчислень завдяки зниженню вартості однієї ітерації. Ця перевага особливо виразна для задач великої розмірності, що підкреслює потенціал методу як надійного й ефективного інструмента для обчислювально складних задач нелінійних найменших квадратів.

Ключові слова: нелінійна задача найменших квадратів; апроксимація оберненого оператора; метод типу Потра; поділена різниця.