ISSN No:-2456-2165 https://doi.org/10.5281/zenodo.14898334

Investigating the Impact of Labeling and Triangulation on the Effectiveness of Fixed Point Algorithms

Nazir Ahmad Parray[1]; Dr. R.S. Patel[2]

[1]Ph.D. (Mathematics) Research Scholar Department of Mathematics,
Govt. Autonomous P.G. College Satna (M.P.) India
[2]Research Supervisor Department of Mathematics,
Govt. Autonomous P.G. College Satna (M.P.) India

Publication Date: 2025/02/20

Abstract: This study explores the influence of labeling techniques and triangulation structures on the performance of fixed point algorithms. A detailed investigation demonstrates how the choice of labeling directly impacts the convergence rate and solution accuracy of these algorithms. Furthermore, two distinct triangulation methods are examined, and theoretical differences between them are hypothesized. These theoretical findings are corroborated by empirical evidence obtained through the implementation of both triangulation methods within a fixed point algorithm. The research aims to provide insights into the interplay between labeling, triangulation, and algorithmic efficiency, thereby advancing the understanding of fixed point computations.

Keywords: *Labeling, Triangulation, Convergence, Fixed Point, Approximation*.

How to Cite: Nazir Ahmad Parray; Dr. R.S. Patel (2025). Investigating the Impact of Labeling and Triangulation on the Effectiveness of Fixed Point Algorithms. *International Journal of Innovative Science and Research Technology*, 10(2), 170-175. https://doi.org/10.5281/zenodo.14898334

I. INTRODUCTION

We are interested in algorithms that determine a fixed point, or a point in R_n where $(y) = y$, of a continuous mapping g from R_n (the n-dimensional Euclidean space) to R_n .[2] and [8] are two prominent algorithms in this area.

These methods deform $g_p\, as\, p \to \infty\, to\, g_\infty = g$, and then follow the path y_p of fixed points of g_p after starting with a constant map $g_0(y) = y_0$. y_∞ is a fixed point of g if on $y_p \to y_\infty$ on a series of $p's$ going to infinity. The algorithm fails in the opposite scenario. Several g features that ensure these algorithms won't fail are listed in [2].

The purpose of this research is to investigate how labeling and triangulation affect the algorithm's effectiveness. There are three sections in it. First examines the impact of labeling on the algorithm, Second examines the method's convergence characteristics, and third examines the impact of triangulation.

II. LABELING

- *Definition:*

A labeling of the triangulation T is a function given a triangulation T of R_n, [10], with the set of vertices $T^0 \subset T$, and an arbitrary collection L.

$\ell: T \to L$.

This labeling function ℓ is created in a way that distinguishes some Simplexes in T and links them to the fixed points of a mapping g.

- *Integer Labeling:*

Given a mapping $g: R_n \to R_n$ by integer labeling we mean that $L = \{1,2, \dots, n+1\}$. The integer labeling that we examine in this article is on the vertices $y\, in\, T^0$. The distinguished Simplexes in this situation are referred to as completely labeled, and they are Simplexes where each vertex receives a different label. Such completely labeled Simplexes can be found using fixed point techniques. These completely labeled simplexes have the following connections to fixed points:

ISSN No:-2456-2165 https://doi.org/10.5281/zenodo.14898334

$$g(y) = \begin{cases} i \ , if\ g_i(y) > y_i\ and\ g_j(y) \le y_j\ for\ all\ i < j \\ n+1\ , \quad if\ g_j(y) \le y_j\ for\ all\ \ j = 1, \dots, n \end{cases} .$$

Given a uniformly continuous mapping : $R_n \to R_n$, and ε > 0 , we say ∂ is determined by ε and the uniform continuity of g whenever $\|y - x\| \le \partial \to \|g(y) - g(x)\| \le \varepsilon$ (here $\|\cdot\|$ denotes the max norm).

- *Theorem 1.1*

[Jeppson, 4] Let $g: R_n \to R_n$ be uniformly continuous on R_n, and ε > 0 be given. Let T be a triangulation of R_n with diameter (mesh) δ , where ∂ is determined by ε and the uniform continuity of g . Let $\ell: T^0 \to L$ be the integer labeling defined above. Then for any completely labeled simplex $= (v^1, \dots, v^{n+1}) \in T$, we have.

$$y \in \sigma \to \|g(y) - y\| \le \varepsilon + \delta.$$

- *Proof: From the definition of ℓ we have*

$$g_j(x) - x_j = g_j(x) - g_j(v^{n+1}) + g_j(v^{n+1}) - v_j^{n+1} + v_j^{n+1} - x_j$$

$$g_j(x) - x_j \le g_j(x) - g_j(v^{n+1}) + v_j^{n+1} - x_j \le \varepsilon + \delta$$

Similarly

$$g_j(x) - x_j = g_j(x) - g_j(v^j) + g_j(v^j) - v_j^i + v_j^i - x_j$$

$$g_j(x) - x_j \ge g_j(x) - g_j(v^j) + v_j^i - x_j \ge -\varepsilon - \delta$$

Hence

$$\|g(x) - x\| \le \varepsilon + \delta.$$

- *Vector Labeling:*

Given a mapping $g: R_n \to R_n$, and $L = R_n$,by a vector labeling we mean the function.

$$\ell(x) = g(y) - y$$

On the vertices $y\ in\ T^0$. The distinguished simplexes in this case are also called completely labeled, and are simplexes $\sigma = (v^1, \dots, v^{n+1})\ in\ T$ such that $0 \in$ convex hull $\{\ell(v^1), \ell(v^2), \dots, \ell(v^{n+1})\}$, i.e., the following system of equations.

$$\sum_{i=1}^{n+1} \lambda_1 \ell(v^i) = 0$$

$$\sum_{i=1}^{n+1} \lambda_1 = 1$$

$$\lambda_i \ge 0, \ \ i = 1, \dots, n+1$$

Has a solution.

If σ is such a completely labeled simplex one define.

$$y^* = \sum_{i=1}^{n+1} \lambda_i v^i$$

To be an approximate fixed point of g. The justification for calling y^* an approximate fixed point of g is the following :

- *Theorem 1.2*

Let $g: R_n \to R_n$ be uniformly continuous on R_n , and let $\varepsilon > 0$ be given . Let T be a triangulation of R_n with mesh ∂ where ∂ is determined by ε and the uniform continuity of g. Let $\ell : T^0 \to L$ be the vector labeling defined above. Then for any completely labeled simplex $\sigma = (v^1, \dots, v^{n+1}) \in T$ and any approximate fixed point $y^* \in \sigma$ we have.

$$\|g(y^*) - y^*\| \le \varepsilon$$

- *Proof : Since $y^* \in \sigma = (v^1, \dots, v^{n+1})$ we have*

$$-\varepsilon \le g_j(y^*) - g_j(v^i) \le \varepsilon\ for\ all\ i = 1, \dots, n+1.$$

Hence

$$-\varepsilon \le g_j(y^*) - \sum_{i=1}^{n+1} g_j(v^i) \le \varepsilon$$

But

$$\sum_{i=1}^{n+1} \lambda_i g_j(v^i) = \sum_{i=1}^{n+1} \lambda_i v_j^i = y_j^*$$

- *And the Result follows.*

The result of Theorem 1.2 can be considerably improved if g is assumed to be twice continuously differentiable with $g_j, j = 1, \dots, n$ having bounded second derivatives ; i.e., there exists an $\alpha > 0\ such\ that\ g\ or\ j = 1, \dots, n$ $|u^t H_j(y) u| \le \alpha ||u||$ for all y and u in R_n where $H_j(y)$ is the Hessian of $g_j at\ y$.

III. CONVERGENCE PROPERTIES

The fixed point algorithms produce a sequence of approximate fixed points y_k , k = 1,2,... on a sequence of grids of diameter δ_k , k = 1,2,.. such that the sequence δ_k converges to zero. It can be readily shown (see in sec. 1st) that all cluster points of the sequence y are fixed points of g.

In this section we will show that the rate of convergence of the sequence $y_k$is, at least, linear. For this we make the following assumptions.

- The sequence $y_k$converges to y_∞.
- g is twice continuously differentiable.
- The matrix $D(g - I)(y_\infty)$ is nonsingular. (Here Dh is the Jacobian matrix of the mapping h and I is the identity map).

ISSN No:-2456-2165 https://doi.org/10.5281/zenodo.14898334

- The Hessian matrix H_j of g_j, j = 1,…,n has the property that $|u^t H_j(y)u| \le \alpha||u||^2$ for all u and y.

We are now ready to prove

- *Theorem 2.1:-*

Let the assumptions 2,1-2.4 hold on a sequence of approximate fixed points generated by a fixed point algorithm, and let $||y_k - g(y_k)|| < \theta_k$. Then, for large enough k , there is a $\rho > 0$ Such that

$$||y_k - y_\infty|| \le \rho\theta_k.$$

- *Proof:*

Taking the second order Taylor's expansion of g_j about y_∞, we get

$$g_j(y_k) = g_j(y_\infty) + \nabla g_j(y_\infty)(y_k - y_\infty) + \frac{1}{2}(y_k - y_\infty)^T H_j(u)(y_k - y_\infty)$$

Since y_∞ is a fixed point of g, rewriting we get

$$-\frac{1}{2}(y_k - y_\infty)^T H_j(u)(y_k - y_\infty) - y_{k,j} + g_j(y_k) = -y_{k,j} + y_{\infty,j} + (y_k - y_\infty)\nabla g_j(y_\infty)$$

Or

$$\frac{1}{2} \propto ||y_k - y_\infty||^2 + \theta_k \ge |(\nabla g_j(y_\infty) - e_j)(y_k - y_\infty)|$$

Or

$$\frac{1}{2} \propto ||y_k - y_\infty||^2 + \theta_k \ge ||D(g - I)(y_\infty)(y_k - y_\infty)||$$

Since $A = D(g - I)(y_\infty)$ is nonsingular, there is a $\varepsilon > 0$ such that

$$||A\, u|| \ge \theta||u||.$$

Hence , we obtain

$$\theta_k \ge \theta||y_k - y_\infty|| - \frac{1}{2} \propto ||y_k - y_\infty||^2$$

Since the sequence y_k converges to y_∞ , for large enough k the quadratic term will be neglibible, giving the result with $\rho = \frac{2}{\theta}$.

As is seen in section ist, θ_k can be computed as a function of δ_k and ε_k . Its exact form depends on the labeling and assumptions about the function.

IV. TRIANGULATION

In this part, we examine how triangulations affect the fixed point algorithm's effectiveness. More results will be reported elsewhere, and the ones we offer here are quite incomplete.

We focus only on Merrill's [8] , Eaves and Saigal [2] algorithms. The following triangulation of R_n, which we'll refer to as H, was used by both of these algorithms. It's produced by choosing any positive real number D at random.

The set of all points in R_n whose coordinates are integer multiples of D constitutes the triangulation's vertices, or H_0. Each simplex in T has a pair (v, π) that uniquely represents it, $where\ v \in H_0$ and π is a permutations of the values $\{1,2,\dots,n\}$. The vertices of the simplex are produced as follows given the pair (v, π).

ss

$$Q = \begin{bmatrix} -D & 0 & 0 & . & . & . & 0 & 0 \\ D & -D & 0 & . & . & . & 0 & 0 \\ 0 & D & -D & . & . & . & 0 & 0 \\ . & . & . & . & & & . & . \\ . & . & . & & . & & . & . \\ . & . & . & & & . & . & . \\ 0 & 0 & 0 & . & . & . & -D & 0 \\ 0 & 0 & 0 & . & . & . & D & -D \end{bmatrix}$$

Then

$$(v, \pi) = (v^1, v^2, \dots, v^{n+1})\ where\ v^1 = v$$

And

$$v^{i+1} = v^1 + Q_{\pi_i}, \qquad i = 1,2,\dots,n.$$

$where\ Q_{\pi_i} is\ the\ \pi_i^{th}$ column of Q.

When this triangulation is only applied to the unit cube $C = \{x \in R_n\ |0 \le x_i \le 1, \quad i = 1,2,\dots,n\}$ it results in $n!$ Simplexes in C. Figure 1 displays these Simplexes for the instance $n = 3$.

The six Simplexes in this instance are the convex hull of the points in the sets $\{1,2,4,5\}, \{3,6,7,8\}, \{2,3,4,8\}, \{2,3,6,8\}, \{2,4,5,8\}, \{2,5,6,8\}$ respectively, since the vertices of C are labeled as in Figure 1.

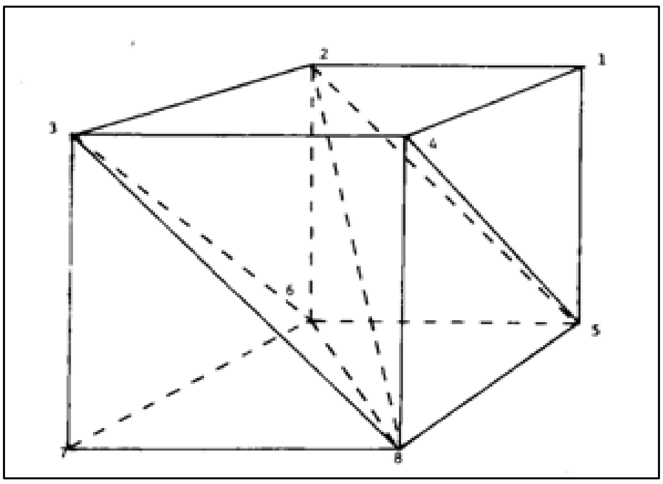

Fig 1 Triangulation

ISSN No:-2456-2165 https://doi.org/10.5281/zenodo.14898334

H could be replaced with another triangulation of R_n called Kuhn [6]. This triangulation, which we refer to as I, was created as shown below. The vertices I_0 of this triangulation are all vectors in R_n whose coordinates are integer multiples of a positive real number D (as in H). Every simplex in I has a unique representation (v, π) *where* $v \in I_0$ and π is a permutations of the values $\{1,2, \ldots, n\}$. The vertices of this triangulation are produced as follows given the pair (v, π).

Let

$$P = \begin{bmatrix} -D & 0 & 0 & . & . & . & 0 \\ 0 & -D & 0 & . & . & . & 0 \\ 0 & 0 & -D & . & . & . & 0 \\ . & . & . & . & & & . \\ . & . & . & & . & & . \\ . & . & . & & & . & . \\ 0 & 0 & 0 & . & . & . & -D \end{bmatrix}$$

Then

$$(v, \pi) = (v^1, v^2, \ldots, v^{n+1}) \text{ where } v^1 = v$$

$$v^{i+1} = v^1 + P_{\pi_i}, \quad i = 1,2, \ldots, n.$$

where P_{π_i} *is the* π_i^{th} column of P.

When this triangulation is limited to the unit cube C , it also produces $n!$ Simplexes. These Simplexes are depicted in Figure 2 for the scenario where $n = 3$. (contrast with Figure 1)

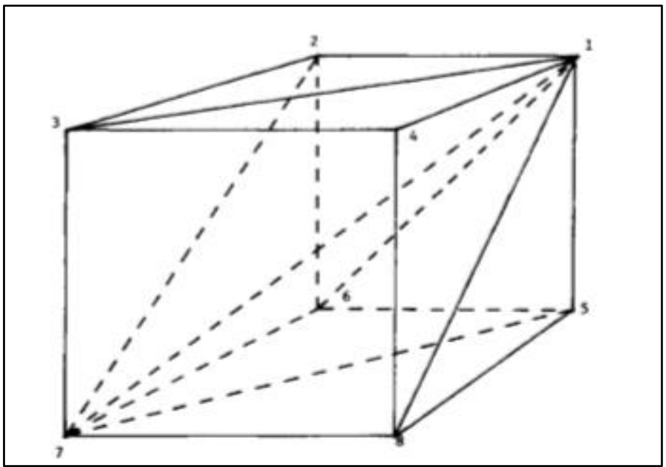

Fig 2 Triangulation

The six Simplexes of this triangulation of C are $C\{1,3,4,7\},\{1,2,3,7\},\{1,4,8,7\},\{1,5,8,7\}, \{1,6,5,7\},\{1,2,6,7\}$ with the vertices of C labeled in Figure 2.

Now, we investigate how the triangulations H and I affect the effectiveness of fixed point algorithms. It should be noted that the triangulation regulates the amount of effort required to move from y_k *to* y_{k+1} (in the sequence examined in 2). The fixed point algorithm pivots traverses numerous cubical regions of the space as it rotates over various Simplexes in R_n. By examining the effort needed to move through a cubical portion of the space, it is possible to determine the effort required to move through the space. This thus enables us to focus our research on how various triangulations of C affect the amount of work needed to pivot across C . The worst situation in both H and I is to pass through $n!$ of these Simplexes. We now define a measure that appears to explain the behavior of the algorithm better than $n!$. To accomplish this, we focus just on the cube C and the triangulation caused by H and , which we will also refer to as H and I, respectively.

- *Boundary Facets of a Triangulations of C:*
  The facets of a triangulation of C that are on its boundary are said to be its boundary facets.

- *Simple Path in the Triangulation:*
  Is a sequence of different facets with $\tau_i, i = 0,1, \ldots, k$ and different simplexes with σ_i, such that τ_{i-1} *and* τ_i are facets of $\sigma_i, i = 1,2, \ldots, k$.

This path is described as being between τ_0 and τ_k and having length k.

$$\tau_0, \sigma_1, \tau_1, \ldots, \sigma_k, \tau_k$$

- *Minimal Path between Boundary Facets:*
  Given a pair of boundary facets τ and $\bar{\tau}$, by $\ell(\tau, \bar{\tau})$ we represent the length of the minimal length simple path between τ and $\bar{\tau}$ in the triangulation.

- *Diameter (Dia) of the Triangulation:*
  Is the maximal of $\ell(\tau, \bar{\tau})$ between any pair of boundary facets τ and $\bar{\tau}$. i.e.,

$$Dia = \max_{\tau, \bar{\tau}} (\tau, \bar{\tau})$$

We propose that the effectiveness of the algorithm is controlled by the diameter of the induced triangulation of C. The triangulations H and I were implemented in Merrill's algorithm to test this. Starting from the same point in each algorithm, different problems were resolved utilizing both of them. It is pretty easy to calculate the diameters for both of these triangulations.

- *They are:*

Triangulation	H	I
Diameter	$\geq O(n^3)$	$\frac{n(n-1)}{2}$

In Table 1(A), the outcomes of using these techniques to solve four nonlinear programming problems are compiled. The problems have dimensions of 5, 6, 8, and 15 respectively

ISSN No:-2456-2165 https://doi.org/10.5281/zenodo.14898334

Table 1(A) Techniques to Solve Four Nonlinear Programming Problems

Dimension (n)	**# starting points**	**Triangulation**	**Total Pivots for all starting points**	**Ratio=$\frac{Pivots\ H}{Pivots\ I}$**	**Ratio for each run**
5	8	I	11,166	1.56	1.12-2.23
		H	17,416		
6	10	I	14,964	2.94	1.66-4.65
		H	41,095		
8	6	I	8,463	3.26	2.15-4.30
		H	27,590		
15	1	I	990	3.59	
		H	3,570		

Wilmuth [11] has discovered similar outcomes when comparing triangulations H and I in the Eaves and Saigal [2] technique. He figured out Scarf's [9] three economic equilibrium issues, which had dimensions 4, 7, and 9 in each case. These triangulations were also tested on a challenge supplied by Kellogg [5]. There are versions of this issue in 20 and 80 dimensions. Merrill's algorithm continued to use the triangulations H and I. Table 1(B) provides a summary of the outcomes for the 20-dimensional problem.

Table 1(B) 20 Dimensional Fixed Point Problem

Run	**Starting point**	**Pivots H**	**Pivots I**	**Pivots H / Pivots I**
1	000...0 0	586	138	4.25
2	111...1 1	1446	234	6.18
3	-1 1 -1…-1 1	562	318	1.77
4	-1-1-1…-1-1	2292	112	20.46
5	-1-1-1-1-1 11111 -1-1…1	706	338	2.09
6	1 11 -1-1-1 111…1	628	580	1.08
7	11 -1 11 -1 … -1	262	310	0.85
8	1 -1 1 -1 … -1	588	292	2.01

The 80-dimensional problem's outcomes are particularly striking. Triangulation I, starting from the point (1,1,...,1,1), solved the problem in 2341 pivots (108.95 seconds on an IBM 370/68), whereas triangulation H did not reach the first level even after 49,000 pivots (40 minutes on an IBM 370/68). With the beginning point, both parties were able to solve the one run (1,1,-1,1,1,-1,...,1). Triangulation I required 4349 pivots compared to 14,988 for triangulation H, for a ratio of 3.42. Table 1(C) lists the additional triangulation I runs.

Table 1(C) Triangulation

Run	**Starting points**	**# Pivots with I**
1	0,0,0,…	1413
2	1,1,1,…	2341
3	1,-1,1,…	4345
4	-1,-1,1,1,-1,-1,…	4595
5	0,-1,0,-1,…	2649
6	11-1,11-1,..	4349
7	-1,-1,-1,…	1393
8	-1,1,-1,1,…	4571

V. CONCLUSION

We have described the influence of labeling on the algorithm, the methods of convergence characteristics and the influence of triangulation, we have defined various theoretical measures of the efficiency of labeling and triangulation for computing fixed point.

REFERENCES

[1]. Patel, R.S., Rusia, R and Patel, P., “Computation of Fixed Point by Using Efficient Algorithm”, J. Comp. and Math. Sci, Vol.4, Issue 4, 31 aug 2013, pp 202-321.

[2]. Eaves, B.C, and Saigal, R., "Homotopies for Computation of Fixed Points on Unbounded Regions," Math. Prog., 3, 1972.

[3]. Patel, R.S., Agerwal, A. and Bhahel, P.S., “Comparison of A.D.D. of Triangulations”, J.nanabha,vol.41, 2011.

[4]. Jeppson, M., "A Search for the Fixed Points of a Continuous Mapping," Mathematical Topics in Economic Theory and Computations, SIAM, Philadelphia, 1972.

[5]. Kellogg, R. B., private communication.

[6]. Kuhn, H. W., "Some Combinatorial Lemmas in Topology," IBM Journal of Research and Development, 4, 1960.

ISSN No:-2456-2165 https://doi.org/10.5281/zenodo.14898334

[7]. Mara, P. S., "Triangulations of a Cube," M.S. Thesis, Colorado State University, Fort Collins, Colorado, 1972.
[8]. Merrill, 0. H., "Applications and Extensions of an Algorithm that Computes Fixed Points of Certain Upper Semi continuous.
[9]. Point to Set Mappings," Ph.D. Thesis, University of Michigan, Ann Arbor, 1972.
[10]. Scarf, H. Computation of Economic Equilibrium, Yale University Press, New Haven, 1973.
[11]. Spanier, E. H. Algebraic Topology, McGraw-Hill, New York, 1966.
[12]. Wilmuth, R. J., private communication.