Least Squares Progressive Iterative Approximation for Spline Curve (Re-)Construction Using Adaptive Refinement*

Šarūnas Palianskas[1]

[1]Kaunas University of Technology, Lithuania

Abstract

Least squares progressive iterative approximation (LSPIA) method allows to approximate data points with curves and surfaces by employing splines that are fitted to the data. The accuracy of approximations is determined by the number of control points used in the splines, thus an appropriate selection of control points is needed to acquire the best approximation. The use of THB-splines as the underlying spline type inside the LSPIA method allows the resulting splines to better approximate data points with highly irregular sections by employing local resolution improvements. This decreases the total number of control points used globally, removes visual artefacts resulting from excessive control points and reduces the amount of resources used during calculations.

Keywords

Geometric iterative approximation, Least squares progressive iterative approximation, THB-splines, Curve fitting

1. Introduction

Least squares progressive iterative approximation (LSPIA) method allows to create approximations of data points that result in continuous functions [1]. The data is approximated with curves or surfaces by employing splines that are fitted to the data. This method creates a series of approximating spline curves or surfaces that approach the least squares fitting of the data as the iterations approach infinity. The number of control points used in the splines is a determining factor for the accuracy of approximations, thus to achieve the best approximation, an appropriate selection of control points is needed. After its introduction, the LSPIA method has garnered a lot of interest and is the subject of continuous studies to improve it in various aspects such as accuracy [2] or convergence [3].

THB-splines [4] are used in a modified LSPIA method as the underlying spline type which allows the method to employ local resolution improvements and better fit data points with highly irregular sections. The use of local resolution improvements decreases the number of control points used globally, removes visual artefacts resulting from excessive control points and reduces the amount of resources used during calculations.

This paper describes the use of LSPIA method for data point approximation with spline curves, the modifications made to the method to use THB-splines and provides a comparison between the original and modified methods by analyzing experiment results.

2. The LSPIA method

Least squares progressive iterative approximation method was introduced by Dent et.al. in 2014 [1], this method improved on the previous method proposed by Lin et.al. [5] by eliminating the need to use the same number of spline control points as there are data points. This significantly reduced the computational resources required by the method and allowed to approximate data with a number of control points that is significantly smaller than the number of data points.

The LSPIA method can be used to approximate both curve and surface data. Surface approximation is an extension of the curve approximation.

*IVUS 2025: Information Society and University Studies 2025, May 15, Kaunas, Lithuania
sarunas.palianskas@ktu.edu (Š. Palianskas)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-06-12

2.1. LSPIA application for curve fitting

The LSPIA method approximates data points given as an ordered set $\{\mathbf{Q}_j : \mathbf{Q}_j \in \mathbb{R}^2, j = 0..m\}$. Each of the data points is assigned a parameter t_j such that $\{0 = t_0 < t_1 < ... < t_{m-1} < t_m = 1\}$.

Data is approximated by selecting a set of control points $\{\mathbf{P}_i^0 : i = 0..n\}$ that are used to construct the initial spline curve $\mathbf{P}^0(t)$:

$$\mathbf{P}^0(t) = \sum_{i=0}^{n} B_i(t)\mathbf{P}_i^0, \quad t \in [t_0, t_m], \quad \mathbf{P}_i^0 \in \mathbb{R}^2. \tag{1}$$

The spline basis is $\{B_i(t) : i = 1..n\}$. Control points for the initial spline curve can be selected arbitrarily, however setting them as a subset of the data points speeds up the convergence of the method.

The method is then executed iteratively until an accepted error level is reached or a set number of iterations. The fitting error of an iteration is:

$$\boldsymbol{\delta}_j^k = \mathbf{Q}_j - \mathbf{P}^k(t_j), \quad j = 0..m, \quad k - \text{iteration number}. \tag{2}$$

After that the adjustment vector for the *i*-th control point is calculated as:

$$\boldsymbol{\Delta}_i^k = \mu \sum_{j=0}^{m} B_i(t_j)\boldsymbol{\delta}_j^k. \tag{3}$$

Here μ is the convergence constant satisfying $0 < \mu < \frac{2}{\lambda_0}$ [1], λ_0 is the largest eigenvalue of the iterative matrix $\mathbf{A}^T\mathbf{A}$, $\mathbf{A}$ is the spline collocation matrix:

$$\mathbf{A} = \begin{bmatrix} B_0(t_0) & B_1(t_0) & \dots & B_n(t_0) \\ B_0(t_1) & B_1(t_1) & \dots & B_n(t_1) \\ \dots & \dots & \dots & \dots \\ B_0(t_m) & B_1(t_m) & \dots & B_n(t_m) \end{bmatrix}. \tag{4}$$

The LSPIA method was proved to converge both when the iterative matrix $\mathbf{A}^T\mathbf{A}$ was non-singular [1] and singular [6].

Control points are updated using the adjustment vector as follows:

$$\mathbf{P}_i^{k+1} = \mathbf{P}_i^k + \boldsymbol{\Delta}_i^k. \tag{5}$$

Then, the new spline curve is:

$$\mathbf{P}^{k+1}(t) = \sum_{i=0}^{n} B_i(t)\mathbf{P}_i^{k+1}. \tag{6}$$

3. THB-splines

THB-splines are a subtype of hierarchical splines. They were described in 2012 by Giannelli et.al. [4], these splines were an improvement on HB-splines that were first introduced in 1988 by Forsey et.al. [7]. The use of THB-splines in computer graphics software was demonstrated by Kiss et.al. in 2014 [8]. Hierarchical splines use multi-level structures, the spline domain is the combination of the domains from each individual level, an illustrative example of the structure can be seen in Figure 1. During the construction of these structures the higher level domains are denser with respect to the basis functions inside them, this allows to represent smaller details as well as to perform adjustments to the spline's control points with a smaller area of impact around the adjusted control point.

Both HB-splines and THB-splines remove basis functions from lower levels if the domain of the basis function is fully covered by the upper level, however THB-splines also employ truncation of

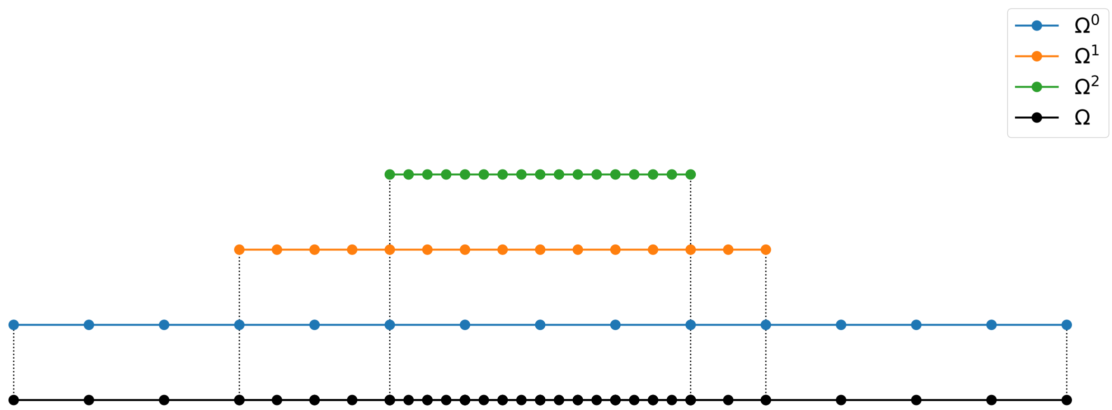

Figure 1: Illustrative example for a multi-level domain structure of THB-splines

basis functions in lower levels that are only partially covered by the upper level. This provides the THB-splines with the partition of unity feature which is present in B-splines, however which was lost in HB-splines due to the hierarchical structure.

THB-spline base is defined with a recursive structure $\mathcal{T}$ formed from L levels in the domain $\Omega^0 \subseteq \Omega^1 \subseteq ... \subseteq \Omega^{L-1}$. Each level contains a spline basis $\mathcal{B}^l,\ l = 0, .., L-1$.

THB-spline basis function for the lowest level is defined as a set of B-spline basis functions inside the full domain: $\mathcal{T}^0 = \{\beta \in \mathcal{B}^0 : S(\beta) \neq \emptyset\}$. Here $S(\beta)$ is a function denoting the domain for basis function β: $S(\beta) = \{x : \ \beta(x) \neq 0 \wedge x \in \Omega^0\}$.

Upper levels are defined with the use of lower ones: $\mathcal{T}^l = \mathcal{T}^l_A \cup \mathcal{T}^l_B,\ l = 1, ..., L-1$. Here $\mathcal{T}^l_A$ is a set of truncated l level basis functions, $\mathcal{T}^l_B$ is a set of basis functions that are fully covered by the l-th level domain:

$$\mathcal{T}^l_A = \{T^l(\tau) : \tau \in \mathcal{T}^{l-1} \wedge S(\tau) \not\subseteq \Omega^l\}, \qquad \mathcal{T}^l_B = \{\beta \in \mathcal{B}^l : S(\beta) \subseteq \Omega^l\}. \tag{7}$$

Function $T^l(\tau) = \sum_{\beta \in \mathcal{B}^l, S(\beta) \not\subseteq \Omega^l} c^l_\beta(\tau)\beta$, with c^l_β being control points of level l corresponding to the β basis function, is the truncation operation described in [4, 9]. This operation reduces basis functions fully covered by either upper or lower level domains to zero, while the remaining basis functions are reduced partially to decrease the basis function domain to only the upper level.

The THB-spline representation of a curve is:

$$\mathbf{P}(u) = \sum_{l=0}^{L} \sum_{j \in A_l} \mathbf{c}^l_j \tau^l_j(u), \quad u \in [0, 1]\,. \tag{8}$$

Here A_l is a set of basis function indices residing in the l-th level.

4. Modification of LSPIA to use THB-splines

The LSPIA method was adapted for the use with THB-splines by substituting the B-spline basis functions with THB-spline basis functions and modifying the iteration structure. Instead of the typical LSPIA iterations the adapted method performs two kinds of iterations. From now on the adapted LSPIA method which uses THB-spline basis functions and the new iteration structure will be referred to as LSPIA-THB method.

The first kind of iterations performed by the method are the typical LSPIA iterations, during their execution the spline control point positions are adjusted. Second type of iterations are performed by refining selected sections of the spline domain that have the largest fitting error. The refining strategy is described by Lin et.al. [10]. The spline domain is sectioned into s_n equally sized segments $T = \left\{ T_i = \left\{ t_j : t_j \in \left[\frac{i}{s_n}, \frac{i+1}{s_n} \right) \right\} : i = 0..s_n - 1 \right\}$. Each segment is assigned an error value calculated according to Equation (9). After segment error calculation the s_m segments with largest error values are selected and the spline is refined for the part of its domain enclosed by the selected segments. During refinement the basis functions and control points inside the refined domain are elevated to a higher level from l to $l + 1$. The number of control points after all refinement operations is denoted as n_r.

The iterations are performed as follows: firstly f iterations of the first kind are performed, then $r - 1$ iterations of the second kind are performed. After each iteration of the second kind f more first kind iterations are performed to fit the positions of new refined control points.

The error for i-th segment is calculated as the sum of Euclidean distances between the parameters in segment domain and their corresponding data points:

$$\epsilon_i = \sum_{t_j \in T_i} \parallel \mathbf{P}(t_j) - \mathbf{Q}_j \parallel . \tag{9}$$

The curve fitting error is calculated as the mean of Euclidean distances between data points and the spline curve value with the corresponding parameter:

$$\epsilon = \frac{1}{m} \sum_{i=0}^{m} \parallel \mathbf{P}(t_i) - \mathbf{Q}_i \parallel . \tag{10}$$

5. Application of LSPIA-THB method for curve approximation

The LSPIA-THB method was analysed in experiments by approximating data points on a plane that describe various curves. The experiments allowed to discover the strengths and weaknesses of the modified method when compared to the original LSPIA method. The experiment results were analysed in a quantitative manner by measuring the fitting error ϵ during iterations and in a qualitative manner by visually comparing the resulting curve approximations.

The selection of the method parameters (n, r, f, s_n, s_m) was performed based on individual datasets with a combination of trial-and-error and expert knowledge. The starting control point number (n) for the LSPIA-THB method's configuration would be selected as a number few times smaller than the number of data points, which would allow ample space for resolution increases. The number of first kind iterations (f) inbetween refinement operations was selected large enought to allow the approximation to reach a near optimum state, which is reached when the change in approximation error subsides. The amount of second kind iterations (r) was selected large enough to allow the method to achieve sufficient resolution to approximate small details and small enough so that the total number of first kind iterations is not too large when compared to the number of iterations needed for the LSPIA method configurations to reach their near optimum state.

5.1. Experiment 1

The experiment was performed with a data set containing $m = 362$ points that resembles an illustration of a dog's profile. The first configuration of parameters corresponded to the original LSPIA method, it used $f = 160$ fitting iterations and $n = 249$ control points. This configuration did not include any refinement operations. The second configuration corresponded to the LSPIA-THB method and started off with $n = 100$ control points that had increased up to $n_r = 249$ after all refinement operations, $r = 8, f = 20, s_n = 13, s_m = 2$ were the other parameters for this configuration. The number of iterations and control points for the first configuration was selected to be similar to the total number of first kind iterations and the final number of control points in the second configuration. The control

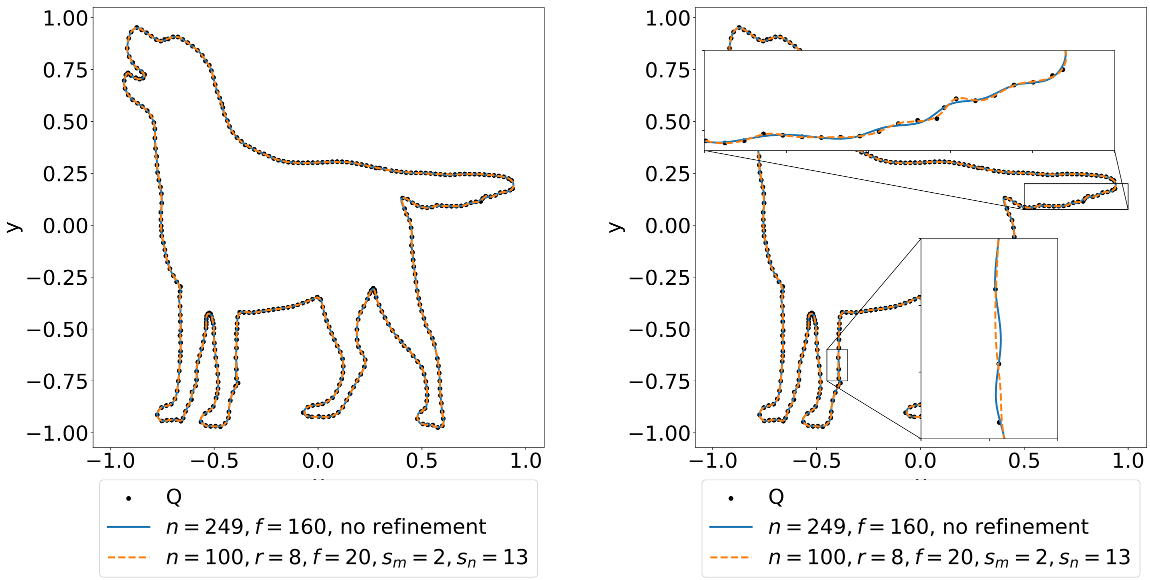

Figure 2: Curve approximation results for the configurations of Experiment 1

Table 1
Fitting error convergence during iterations of Experiment 1

Iteration	First configuration error	Second configuration error
0	0.040610	0.039086
40	0.001128	0.003172
80	0.001011	0.002098
120	0.000983	0.001400
159	0.000968	0.000952

points of the first configuration are distributed nearly evenly, meanwhile the second configuration has an increased control point density after all the refinement operations are performed around highly irregular curve sections.

The resulting approximation curves of the experiment can be observed in Figure 2. Both of the configurations can be seen to result in an adequately fitting curve, however key differences can be observed in curve sections that are highly irregular.

It can be observed that the curve sections of the first configuration curve near the "tail" lack precision to properly approximate the highly varied data. Meanwhile the second configuration shows much better results with the spline curve approximating the data very well after the refinement operations.

While some sections of the curve from first configuration were lacking precision, others can be observed to be the opposite and had too many control points which resulted in artefacts when the resulting curve zig-zags although the data does not require that. Such artefacts can be observed in the neighbourhood of point $(-0.4, -0.7)$

The fitting errors were measured for both configurations during each iteration, the results can be seen in Table 1. The first configuration error curve is smooth and converges towards $\epsilon = 9.68 \times 10^{-4}$, meanwhile the second configuration error converges towards $\epsilon = 9.52 \times 10^{-4}$. The error convergence rate of the second configuration begins to dissipate multiple times, but is increased again after refinement, when the number of control points increases and the fitting can be further improved. The fitting error convergence can be seen in Figure 3.

The final fitting error for both configurations is close with LSPIA-THB method achieving slightly lower errors. However the main difference between the two can be observed by visually comparing the

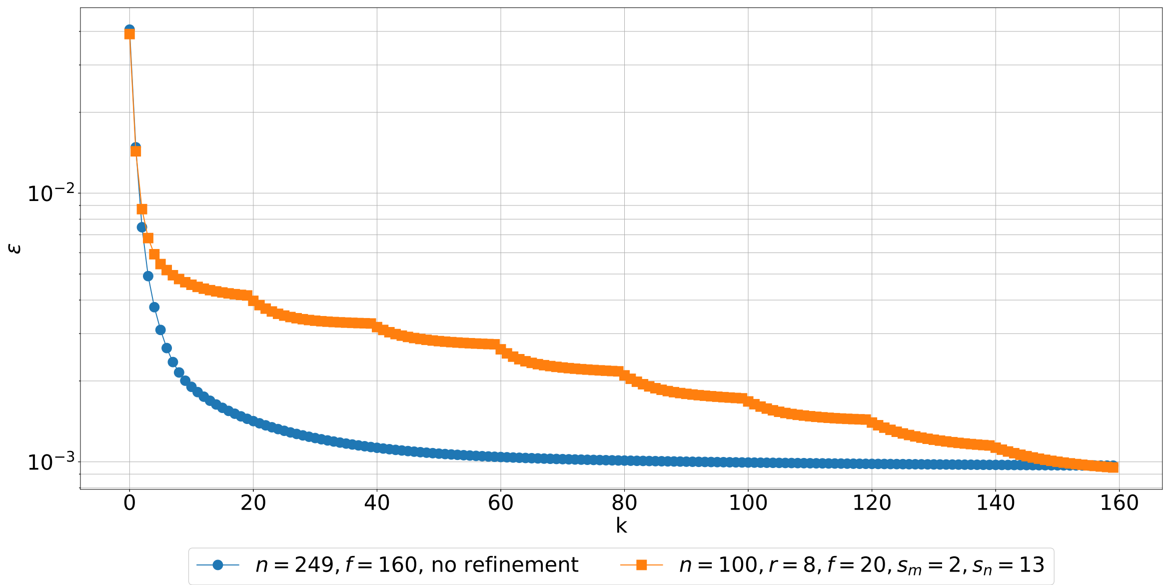

Figure 3: Fitting error convergence during iterations of Experiment 1

Table 2
Fitting error convergence during iterations of Experiment 2

Iteration	First configuration error	Second configuration error	Third configuration error
0	2.576373	1.500081	2.296320
100	0.081260	0.006696	0.047879
200	0.049299	0.004333	0.047583
300	0.021666	0.003868	0.047495
399	0.011260	0.003704	0.047463

resulting curves.

5.2. Experiment 2

The second experiment was performed with a dataset of $m = 568$ points. The first configuration was: $n = 150, n_r = 574, f = 40, r = 10, s_n = 15, s_m = 3$, it corresponded to the LSPIA-THB method. The experiment contained two more configurations that represented the LSPIA method and contained no refinement operations. The second configuration was $n = 574$, $f = 400$, it contained more control points than there are data points which was the control point count of the first configuration after all refinement operations. The third configuration was $n = 250$, $f = 400$ with the number of control points close to half that of data points.

The resulting spline curves can be seen in Figure 4. The results show how the LSPIA method with a number of control points exceeding that of the data points causes significant loss of fitting quality due to the introduced degrees of freedom. Meanwhile the results of first configuration that contained more control points than data points after all refinement operations approximated the data successfully. The curve resulting from the third configuration was fitted to the data in an acceptable manner, however by examining sections of data with high irregularity, a lack of precision can be observed as there are not enough control points to accurately approximate the data.

The fitting errors measured during iterations are present in Table 2. The measurements show that the lowest fitting error was achieved by the second configuration and was $\epsilon = 3.704 \times 10^{-3}$, the other configuration for LSPIA method achieved its lowest error of $\epsilon = 4.7463 \times 10^{-2}$ that was significantly higher. The lowest fitting error achieved by the first configuration corresponding to the LSPIA-THB

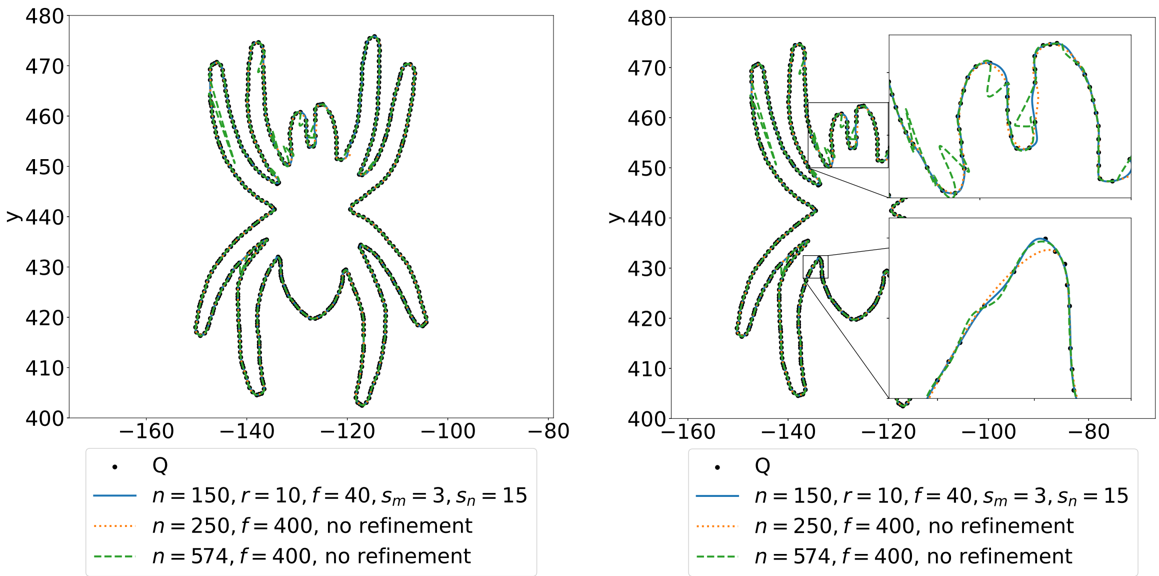

Figure 4: Curve approximation results for the configurations of Experiment 2

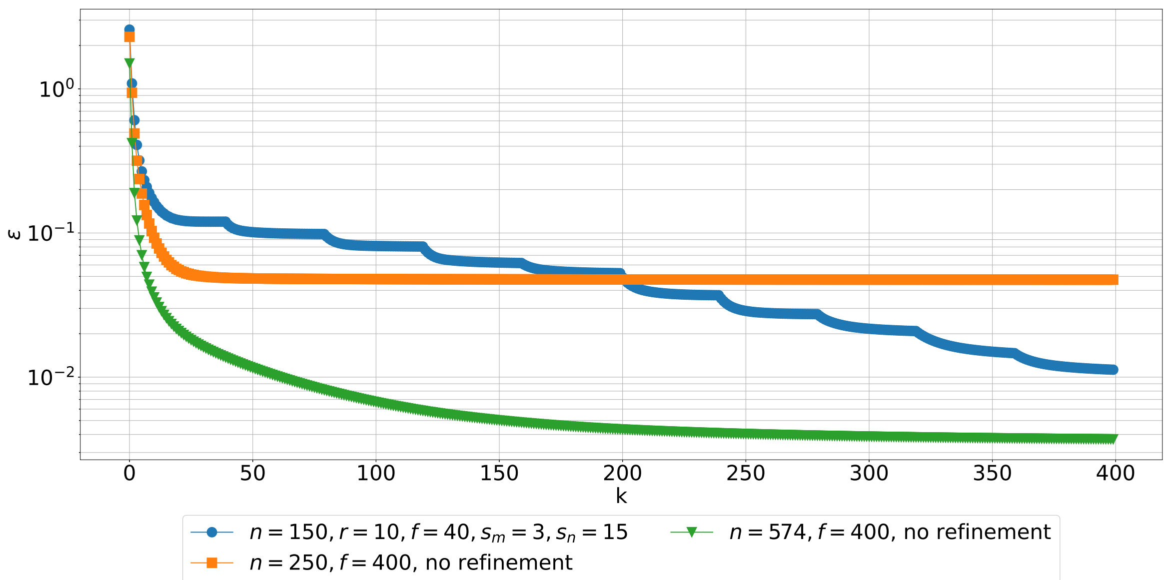

Figure 5: Fitting error convergence during iterations of Experiment 2

method was $\epsilon = 1.126 \times 10^{-2}$. It is higher than the second configuration's, however the low fitting error achieved by introducing degrees of freedom for the fitting curve of LSPIA method results in poor qualitative results. Fitting error convergence curves can be seen in Figure 5.

5.3. Experiment 3

The third experiment was performed with a dataset of $m = 320$ points that represents a polygonal shape. First configuration was: $n = 100, n_r = 382, r = 10, f = 20, s_n = 15, s_m = 3$, which corresponded to the LSPIA-THB method. The second and third configurations corresponded to the LSPIA method and did not contain refinement operations. The second configuration was: $n = 100$, $f = 200$, it contained the same number of control points as the first one before any refinement. The third configuration was:

Table 3
Fitting error convergence during iterations of Experiment 3

Iteration	First configuration error	Second configuration error	Third configuration error
0	3.387906	3.387906	1.129672
50	0.366269	0.539331	0.067548
100	0.237659	0.539281	0.060421
150	0.111075	0.539281	0.058247
199	0.033517	0.539281	0.057104

$n = 300$, $f = 200$, the control point number was selected to be slightly smaller than the number of data points.

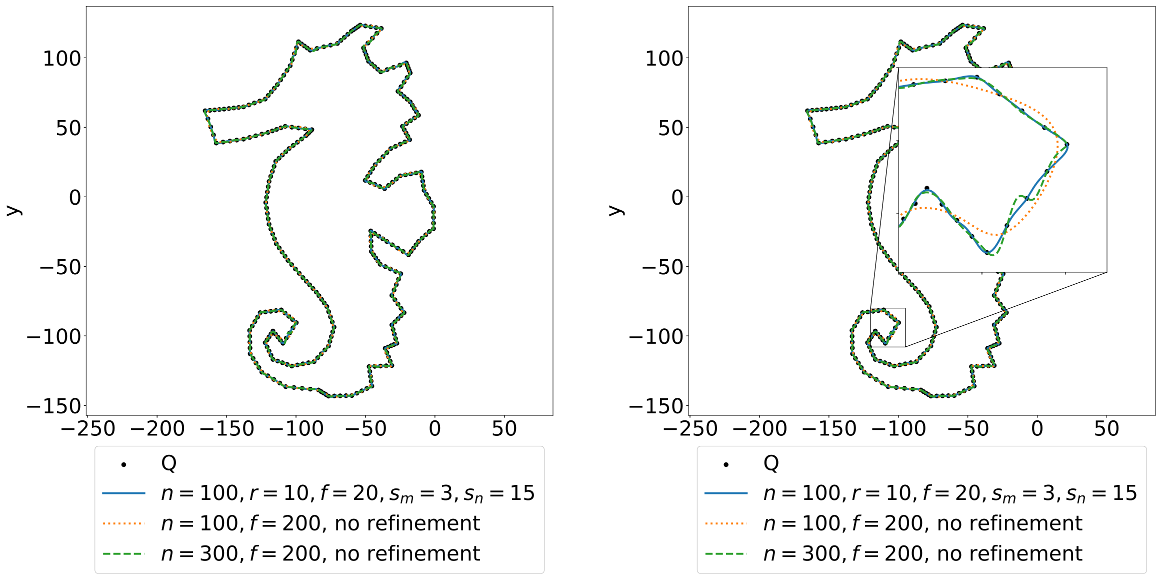

Figure 6: Curve approximation results for the configurations of Experiment 3

The approximation results that can be seen in Figure 6 show how all of the curves maintained a smooth shape and did not fully conform to the straight lines that the data represents. This is to be expected, as sharp corners can only be achieved by duplicating control points, which the methods used do not have the ability to do. However it can be observed that the curve resulting from the first configuration approximates the data near sharp corners much better and reduces the curving sections to a minimum.

The fitting errors measured during iterations can be seen in Table 3. The smallest fitting error is achieved by the first configuration representing the LSPIA-THB method. The convergence of fitting errors during iterations can be observed in Figure 7, the fitting errors of the second and third configurations have nearly converged to their respective values and their rate of change is near zero, however the rate of change for the first iteration can be observed to be globally increasing, which suggests that further refinement operations would allow to approximate the data even better.

6. Conclusions

This paper explored a modification of the LSPIA method to use THB-splines in approximating sets of data points with spline curves. This modification allows the use of refinement operations that increase the approximation accuracy in a local area. The modifications made to the method and the operating principles of it were described. Experiments were performed with datasets representing various kinds

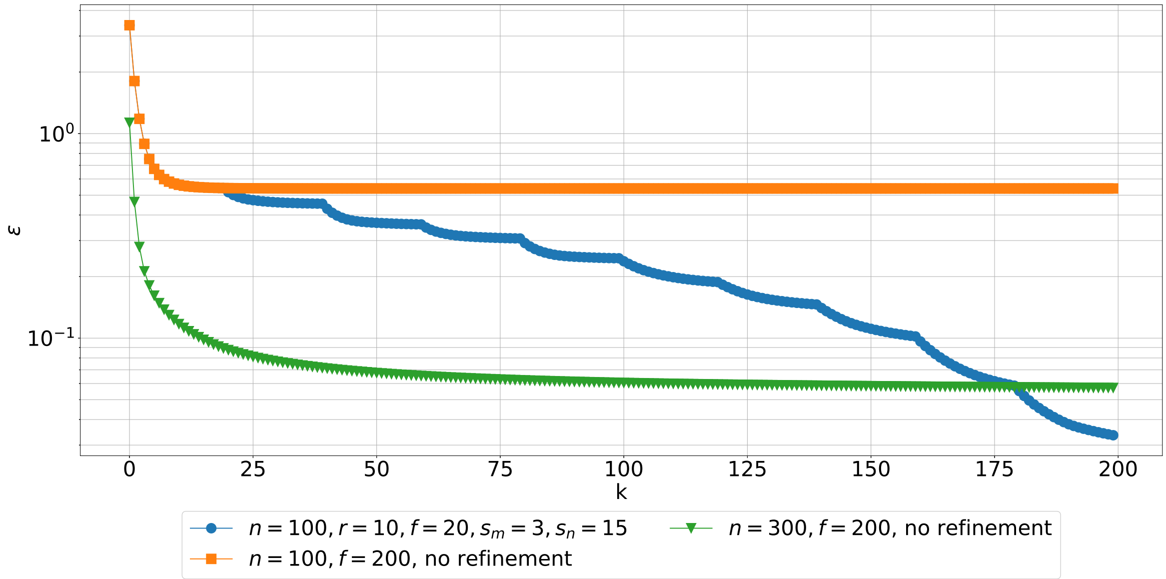

Figure 7: Fitting error convergence during iterations of Experiment 3

of plane point collections that assessed and compared the capabilities of both the original LSPIA and the modified LSPIA-THB method in approximating the data. The experiments showed how the LSPIA-THB method manages to more accurately approximate data with sections of high irregularity that require fine adjustments of the approximating curve in local areas. The LSPIA-THB method was observed to achieve significantly better qualitative results when approximations were performed with degrees of freedom as well as construct an approximating curve that fits a polygonal shape much better than similar configurations of the original LSPIA method. The promising results achieved by applying the LSPIA-THB method to curve approximations show the potential of this method in areas like computer graphics, while further research into applications of the method for surface fitting would provide the ability to expand the usefulness of the method.

Declaration on Generative AI

The author has not employed any Generative AI tools.

References

[1] C. Deng, H. Lin, Progressive and iterative approximation for least squares B-spline curve and surface fitting, Computer-Aided Design 47 (2014) 32–44. doi:https://doi.org/10.1016/j.cad.2013.08.012.

[2] L. Lan, Y. Ji, M.-Y. Wang, C.-G. Zhu, Full-lspia: A least-squares progressive-iterative approximation method with optimization of weights and knots for nurbs curves and surfaces, Comput. Aided Des. 169 (2024). doi:10.1016/j.cad.2023.103673.

[3] S. Sajavičius, Hyperpower least squares progressive iterative approximation, J. Comput. Appl. Math. 422 (2023) 114888. doi:10.1016/j.cam.2022.114888.

[4] C. Giannelli, B. Jüttler, H. Speleers, THB-splines: The truncated basis for hierarchical splines, Computer Aided Geometric Design 29 (2012) 485–498. doi:10.1016/j.cagd.2012.03.025.

[5] H. Lin, G. Wang, C. Dong, Constructing iterative non-uniform B-spline curve and surface to fit data points, Science in China Series F: Information Sciences 47 (2004) 315–331. doi:10.1360/02yf0529.

[6] H. Lin, Q. Cao, X. Zhang, The convergence of least-squares progressive iterative approximation with singular iterative matrix (2017). arXiv:1707.09109.
[7] D. R. Forsey, R. H. Bartels, Hierarchical B-spline refinement, ACM SIGGRAPH Computer Graphics 22 (1988) 205–212. doi:10.1145/378456.378512.
[8] G. Kiss, C. Giannelli, U. Zore, B. Jüttler, D. Großmann, J. Barner, Adaptive CAD model (re-)construction with THB-splines, Graphical Models 76 (2014) 273–288. doi:10.1016/j.gmod.2014.03.017.
[9] C. Giannelli, B. Jüttler, H. Speleers, Strongly stable bases for adaptively refined multilevel spline spaces, Advances in Computational Mathematics 40 (2013) 459–490. doi:10.1007/s10444-013-9315-2.
[10] H. Lin, Z. Zhang, An efficient method for fitting large data sets using T-splines, SIAM Journal on Scientific Computing 35 (2013) A3052–A3068. doi:10.1137/120888569.