

Modified brute force algorithm to solve the closest pair of points problem based on dynamic warping

Rhowel M. Dellosa¹, Arnel C. Fajardo², Ruji P. Medina³

^{1,3}Graduate School, Technological Institute of the Philippines Quezon City, Philippines

²School of Engineering and Information Technology, Manuel L. Quezon University, Philippines

Article Info

Article history:

Received Oct 1, 2018

Revised Dec 10, 2018

Accepted Jan 25, 2019

Keywords:

Brute force

Closest pair of points

Divide-and-conquer

Dynamic warping

Position estimation

ABSTRACT

This paper introduces an algorithm to solve the closest pair of points problem in a 2D plane based on dynamic warping. The algorithm computes all the distances between the set of points $P(x, y)$ and a reference point $R(i, j)$, records all the result in a grid and finally determines the minimum distance using schematic steps. Results show that the algorithm of finding the closest pair of points has achieved less number of comparisons in determining the closest pair of points compared with the brute force and divide-and-conquer methods of the closest pair of points.

Copyright © 2019 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Rhowel M. Dellosa,

Graduate School,

Technological Institute of the Philippines-Quezon City,

938 Aurora Boulevard, Cubao, Quezon City, Philippines.

Email: rhoweldellosa@asiatech.edu.ph

1. INTRODUCTION

The determination of the closest pair of points (CPP) in a two-dimensional plane is one of the basic problems of computational geometry. Similarity matching, pattern recognition, computer vision and position estimation are some of the applications of CPP [1] and [2]. One of the typical approaches to solving the problem on the closest pair of points is the brute force method. In this method, the computation of distances of all pairs of points is necessary to find out the closest pair. To compute for the total number of comparisons of brute force method, equation (1) was used.

$$TC = \frac{(n-1)n}{2} \quad (1)$$

where:

TC - The number of comparisons

n - Total number of distance to be compared.

The divide-and-conquer algorithm was used to improve the brute force method. The study of [3] used this approach to solve the closest pair problem in a 2-dimensional plane. It utilized the concept introduced by [4] on the closest pair of points problem and modified the divide-and-conquer algorithm by computing at least seven pairwise comparisons in the central slab. Another enhancement was presented by [5] wherein an improved version of the divide-and-conquer algorithm on the planar set used at most four pairwise comparisons in the combining step for each point in the central slab. In 2006, the study of [6] showed that the divide-and-conquer algorithm pairwise comparison is possible to be reduced in the

combining step in the central slab for their enhanced version. Later, the study of [7] presented another enhancement of divide-and-conquer algorithm called the Basic-2 where two pairwise comparisons were used in the combining step for each point in the central lab.

In 2015, the Pair-Pruned_CP algorithm of [8] was introduced. It is a simple and efficient algorithm to solve the closest pair of points problem. The algorithm functionalities focused on the computation of the distances of the pair of points which may be closer than the initial closest pair of points. It is a comparison-based deterministic approach to solve the CPP problem.

Dynamic warping (DW) is a dynamic programming technique used to match the sequence for a specific application. Some of the applications of DW include the undirected graph model for calibrating and storing sequence-type radio-map which reduced the disparities of RSS. The study of [9] showed an increase in accuracy of 20% using dynamic warping. The interesting feature of dynamic warping is its heuristic approach to pattern searching. The warpmap of [9] is an accurate and efficient indoor localization in sequence type radio map. The key feature of this study was the capability of its system of accurate and efficient position estimation.

Another CPP algorithm is the study of [10] that makes use of an algorithm to solve the closest pair problem with the aid of the tree model. It was also proven by the study of [11] that with the use of a divide-and-conquer approach, the closest pair of point problem can be initially solved in much higher dimensions for planar case. The divide-and-conquer algorithm makes it possible to lessen the number of comparisons which contribute to the efficiency of the approach. The study of [12] the divisions into several numbers of groups were considered in their experiment that decreases the number of comparisons. It was also proven by [13] and [14] used plane-sweep and optimal algorithm to solve the closest pair of point's problem.

The theory behind [15, 16] on the closest pair of points problems solutions had considered the application of several algorithms to reduce the expected running time. These algorithms are the same algorithm used for the locality sensitive hashing [17], light bulb problem [18] and coin flip of Rabin [19].

One of the similar closest pairs of point's problem is the time series motif mining. It is an application of CPP in large dimensional space that needs to have an efficient and accurate result. The study of [20] presented the first exact motif search algorithm for time series motif mining that outperformed the brute-force algorithm. This algorithm became the best performing algorithm in practice for time series motif mining. The same problem was also explored by several studies that used probabilistic and approximate algorithms for solving time series motif mining. The study of [21] applied it to the motion-motif graph. The study of [22] enhanced the time series motifs using the probabilistic algorithm. The study of [23] explored the possibility of application of time series in knowledge construction using a collaborative exploration system. The paper of [24] applied it to human motion and [25] have tried to find out that multivariate Motifs using subsequence density estimation and greedy mixture learning is possible and useful for time series motif mining reference. Both of the papers of [21] and [22] are concerned on the application of time series motif mining principle to machine learning wherein a large scale time series database are processed.

Taking an account on its functionalities on the number of comparisons, the brute force method has the most number of comparisons of the distances while the divide-and-conquer algorithm has reduced such. In the case of coordinate-based similarity determination, the closer the reference point to the set of points, the more similar the reference point.

Consider the reference point $R(i, j)$ and the set of points $\{A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$ and $D(x_4, y_4)\}$. The brute force method will find out all the distances of the reference point $R(i, j)$ from the five points $\{R, A, B, C, D\}$ which accumulates 10 distance computations $\{RA, RB, RC, RD, AB, AC, AD, BC, BD, CD\}$ and determine the minimum distance to find out the corresponding closest point from the reference point. The divide-and-conquer algorithm will find out the number of possible divisions and compute for the distances of each of the division of points. As a result, the number of comparisons between the distances is reduced.

In this study, a new approach to solve the CPP problems is presented where the dynamic warping is applied to minimize the number of comparisons of distances between the reference point $R(i, j)$ and the set of points $P(x_n, y_n)$. The computed distances are presented in coordinates-map to be used for similarity determination. The numbers of comparisons of the proposed algorithm are compared with brute force and divide-and-conquer methods of CPP. Future applications may consider the use of radio frequency identification and visible light communication for location estimation as studied by [26].

2. PROPOSED ALGORITHM

The proposed modification of the brute force method of solving the CPP problem is presented by introducing the coordinates-map grid. A coordinates-map grid with 16 sample cells is illustrated in Figure 1. Figure 2 shows the coordinates-map grid. Each cell of the coordinates-map grid is the container of the

distance between the reference point $R(i, j)$ and the set of points assigned to a particular cell. For example, the D11 cell contains the distance between the reference point $R(i, j)$ and the coordinates $P(1, 1)$ and D12 cell contains the distance between the reference point $R(i, j)$ and the coordinates $P(1, 2)$.

D14	D24	D34	D44
D13	D23	D33	D43
D12	D22	D32	D42
D11	D21	D31	D41

Figure 1. Coordinates-map Grid

0.6	1.2	2.1	3.1
0.4	1.1	2.0	3.0
1.4	1.7	2.4	3.3
2.4	2.6	3.1	3.8

Figure 2. Sample Distance Output of Coordinates-map Grid

2.1. Algorithm for Distance Computation and Storing of Distance in the Coordinates-Map Grid

Input : Reference Point $\{R(i, j)\}$ and the set of points $\{(1, 1)..(1, 4), (2, 1)..(2, 4)..(4, 1)..(4, 4)\}$
 Output : Distances between the reference point (R) and the set of points (P) in the coordinate-map grid

```

Loop0:
For x =1 to 4
Loop1:
For y = 1 to 4
    Dxy = distance between R(i, j) and P (x, y)
    inc y
    Store Dxy in the coordinates-map grid
    if x<=4 then
    Goto loop 1
    Else
        x=1
        inc y
        Goto loop0
    Endif
End for
End for
    
```

Sample output:

Input : Reference Point $\{R(1, 3.4)\}$ and the set of points $\{(1, 1)..(1, 4), (2, 1)..(2, 4)..(4, 1)..(4, 4)\}$
 Output : Distances between the reference point (R) and the set of points (P) in the coordinate-map grid

2.2. Schematic Steps that Determine the Minimum Distance and the Set of Point P(X, Y) Closer to Reference Point R(I, J)

Input : Distances between the reference point $R(i, j)$ and $P(x, y)$ in the coordinates-map grid.
 Output : Closest point $P(x, y)$.

- Using the coordinates-map grid of Figure 1, start from the upper right corner of the coordinates-map grid using the schematic step presented in Figure 3.

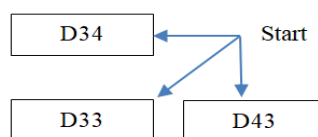


Figure 3. The schematic step

2. Follow the schematic step of Figure 3.

D14	D24	D34	D44
D13	D23	D33	D43
D12	D22	D32	D42
D11	D21	D31	D41

Figure 4. Illustration of the schematic step to the coordinates-map grid

3. Determine the minimum value of the four cells (D44, D34, D33, and D43).
4. Move to the next step by considering the minimum distance value as the “Start” of the next schematic step. (Assumption: D34 is the minimum value).

D14	D24	D34	D44
D13	D23	D33	D43
D12	D22	D32	D42
D11	D21	D31	D41

Figure 5. Illustration of the next Schematic Step

5. If the minimum distance value did not change anymore, the corresponding coordinates of the minimum distance are the closest coordinates of the reference point R (i, j).

Sample output:

Input : Distances between the reference point R(1, 3.4) and P(x, y) displayed in the coordinates-map grid.

Output : Closest point P(x, y).

Steps:

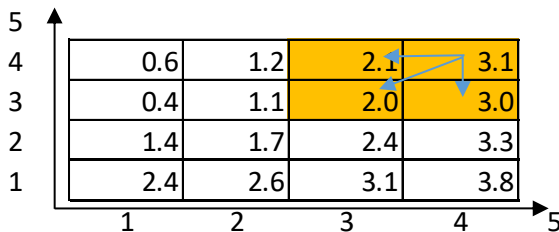


Figure 6. First Step



Figure 7. Second Step

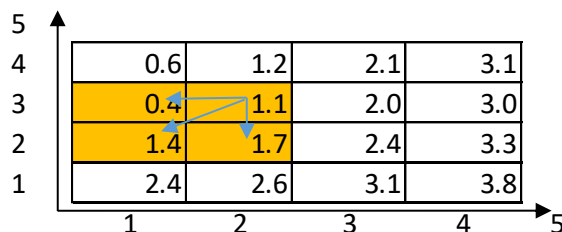


Figure 8. Third Step

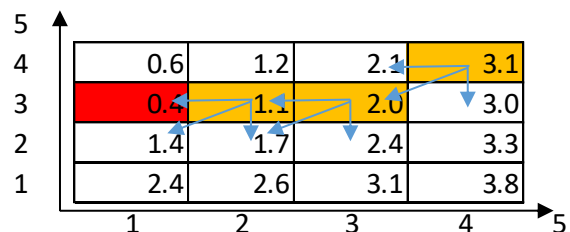


Figure 9. Summary of Steps Process

Output: P(x, y) = P (1, 3). Therefore, the reference points R (1, 3.4) and P (1, 3) are the closest pairs of points.

3. RESULTS AND ANALYSIS

The proposed algorithm was executed with the use of the Java programming language. Point R (1.3, 2.2) was used as the reference point (R) and 100 points {(1, 1)..(1, 10), (2, 1)..(2, 10)...(10,1)..(10, 10)} samples were used as the set of points (P). These reference points (R) and the sample set of points (P) were used to demonstrate how the proposed algorithm works.

3.1. Computation of Distance

Input : Reference Point {R(1.3, 2.2)} and the set of 100 points {(1, 1)..(10, 10)}

Output : Distances between {R(1.3, 2.2)} and the set of 100 points

Table 1 shows the summary of computed distances of coordinates-map. A 10x10 coordinates-map grid was used to present the computed distances between the reference point and the set of points. Figure 11 shows that the minimum distance is the distance located at P(1, 2) which is equivalent to 0.4. Hence, the closest pair of points is R(1.3, 2.2) and P(1, 2).

Table 1. Summary of Computed Distance of Coordinate-Map

7.8	7.8	8.0	8.3	8.6	9.1	9.7	10.3	11.0	11.7
6.8	6.8	7.0	7.3	7.7	8.3	8.9	9.5	10.3	11.0
5.8	5.8	6.0	6.4	6.9	7.5	8.1	8.9	9.6	10.5
4.8	4.9	5.1	5.5	6.1	6.7	7.5	8.2	9.1	9.9
3.8	3.9	4.2	4.7	5.3	6.0	6.9	7.7	8.6	9.5
2.8	2.9	3.3	3.9	4.6	5.5	6.4	7.3	8.2	9.1
1.8	1.9	2.5	3.2	4.1	5.0	6.0	6.9	7.9	8.9
0.9	1.1	1.9	2.8	3.8	4.8	5.8	6.7	7.7	8.7
0.4	0.7	1.7	2.7	3.7	4.7	5.7	6.7	7.7	8.7
1.2	1.4	2.1	3.0	3.9	4.9	5.8	6.8	7.8	8.8

3.2. Determination of Closest Point to the Reference Point

Input : Distances between the reference point R(1.3, 2.2) and 100 set of points.

Output : Closest point P(x, y).

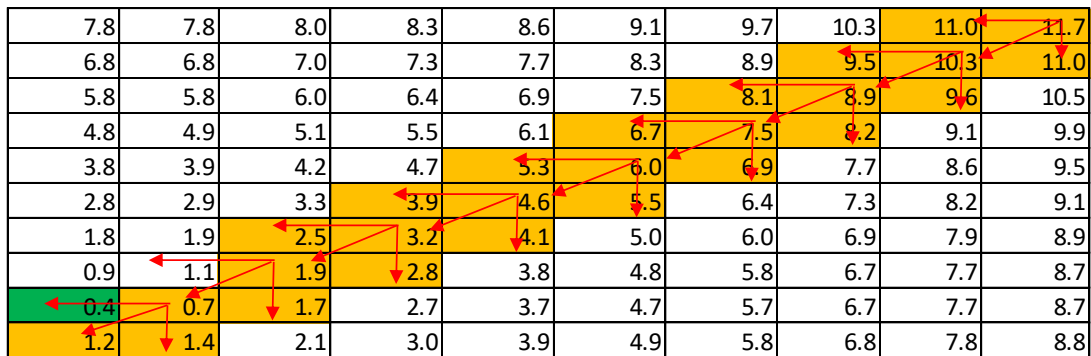


Figure 10. Steps of Execution Process during the Determination of Closest Point

7.8	7.8	8.0	8.3	8.6	9.1	9.7	10.3	11.0	11.7
6.8	6.8	7.0	7.3	7.7	8.3	8.9	9.5	10.3	11.0
5.8	5.8	6.0	6.4	6.9	7.5	8.1	8.9	9.6	10.5
4.8	4.9	5.1	5.5	6.1	6.7	7.5	8.2	9.1	9.9
3.8	3.9	4.2	4.7	5.3	6.0	6.9	7.7	8.6	9.5
2.8	2.9	3.3	3.9	4.6	5.5	6.4	7.3	8.2	9.1
1.8	1.9	2.5	3.2	4.1	5.0	6.0	6.9	7.9	8.9
0.9	1.1	1.9	2.8	3.8	4.8	5.8	6.7	7.7	8.7
0.4	0.7	1.7	2.7	3.7	4.7	5.7	6.7	7.7	8.7
1.2	1.4	2.1	3.0	3.9	4.9	5.8	6.8	7.8	8.8

Figure 11. Summary of Execution Process during the Determination of Closest Point

3.3. Derivation of the Number of Comparisons of the Proposed CPP Algorithm

The brute force method compares all the distances between each pair of points while the divide-and-conquer algorithms divide the set of distances into two and compare all the distances between each pair of points of each division. Applying equation (1) of the brute force method and divide-and-conquer for 100 distances to be compared resulted in $TC = 4950$ and $TC = 2450$, respectively. In the proposed CPP algorithm, the number of comparisons only depends on the number of schematic steps. The equation (2) is derived to compute for the number of comparisons:

$$Ku = A \binom{(n-1)n}{2} \quad (2)$$

where:

Ku - The number of comparison of the proposed CPP algorithm

A - Number of schematic steps

n - Total number of distances to be compared

Applying the equation (2), the number of comparisons of the proposed CPP algorithm is $Ku = 60$ with $n = 4$ and the worst case value of $A = 10$ for 100 distances value.

3.4. Data on the Comparisons of other Number of Distances

Table 2 shows the presentation of the number of comparisons between the brute force, divide-and-conquer algorithm and the proposed CPP algorithm using 100, 10,000 and 1,000,000 distances, respectively. To determine the number of comparisons of the distances between the reference point and set of points for brute force and divide-and-conquer algorithm, the equation (1) was used. Moreover, to determine the number of comparisons of the distances between the reference point and set of points for the proposed CPP algorithm, the equation (2) was used. In addition, the proposed CPP algorithm used 10x10, 100x100 and 1000x1000 coordinates-map grid and considered the worst case value of schematic steps (A) of 10, 100 and 1000 respectively.

Table 2. Number of Comparisons of Brute Force Method and the Proposed CPP

Number of Distances to be Compared	Brute Force Method	Divide-and-conquer	Proposed CPP algorithm
	(Number of Comparisons)	Algorithm (Number of Comparisons)	(Number of Comparisons) With $A=10, 100, 1000$ respectively
100	4,950	2,450	60
10,000	49,995,000	24,995,000	600
1,000,000.00	4.99×10^{11}	2.44×10^{11}	6,000

4. CONCLUSION

The modified brute force algorithm is the proposed algorithm for solving the CPP problem. The proposed algorithm is simple and easy to follow. The experiment results showed that the number of comparisons of the proposed CPP algorithm is less than the brute force method and the divide-and-conquer algorithm. Moreover, the proposed CPP algorithm is efficient in terms of the number of comparisons as proven by the experiments of different sets of computed distances. The proposed algorithm may be implemented on coordinate based applications like position estimation and similarity matching to further test its functionalities.

ACKNOWLEDGEMENTS

The authors would like to thank their family, loved ones, the Commission on Higher Education, Technological Institute of the Philippines-Quezon City, Manuel L. Quezon University and Asia Technological School of Science and Arts for being the inspirations on this research endeavor.

REFERENCES

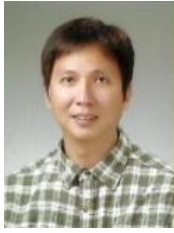
- [1] E. Dalkılıç and S. Er, "A simplest fast algorithm for the closest-pair problem using sorted projections on multi-dimensions," Proceedings of the 28th International Symposium on Computer and Information Sciences Springer, pp. 23-34, 2013.
- [2] J. C. Pereira and F. G. Lobo, "An optimized divide-and-conquer algorithm for the closest-pair problem in the planar case," *Journal of Computer Science and Technology*, vol. 27, pp. 891-896, 2012.

- [3] W. Huang, "Walkielokie: Sensing Relative Positions of Surrounding Presenters by Acoustic Signals," in Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 439-450, 2016.
- [4] M. Shamos and D. Hoey, "Closest Point Problem," *16th Annual Symposium on Foundations of Computer Science (sfcs 1975)*, Oct 1975.
- [5] Y. Zhou, et al., "An improved algorithm about the closest pair of points on plane set," *Computer Research and Development*, vol. 35, pp. 957-960, 1998.
- [6] Q. Ge, et al., "An Improved Algorithm for Finding the Closest Pair of Points," *Journal of Computer Science and Technology*, vol. 21, pp. 27-31, Jan 2006.
- [7] J. Jiang and Gillespie, "Engineering the divide-and-conquer closest pair algorithm," *Journal of Computer Science and Technology*, vol. 22, pp. 532-540, 2007.
- [8] Y. Zhou and H. Yu, "An Efficient Comparison-Based Deterministic Algorithm to Solve the Closest Pair Problem," 2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA), Nanchang, pp. 145-148, 2015.
- [9] Y. Wang, et al., "WarpMap: Accurate and Efficient Indoor Location by Dynamic Warping in Sequence-Type Radio-Map," 2016 13th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), pp. 1-9, 2016.
- [10] A. C. Yao, "Lower bounds for algebraic computation trees with integer inputs," *SIAM Journal on Computing*, vol. 20, pp. 655-668, 1991.
- [11] J. L. Bentley and M. I. Shamos, "Divide-and-conquer in multidimensional space," in Proceedings of ei ghth annual ACM Symposium on Theory of Computing (STOC), New York, NY, USA, pp. 220-230, 1976.
- [12] M. Z. Karim and N. Akter, "Optimum partition parameter of divide-and conquer algorithm for solving closest-pair problem," *International Journal of Computer Science & Information Technology*, vol. 3, pp.211-219, 2011.
- [13] K. Hinrichs, et al., "Plane-sweep solves the closest pair problem elegantly," *Information Processing Letters*, vol. 26, pp. 255-261, 1988.
- [14] C. Schwarz, et al., "An optimal algorithm for the on-line closest-pair problem," in Proceedings of the 8th Annual Symposium on Computational Geometry, ACM Press, New York, pp. 330-336, 1992.
- [15] X. Zhang, et al., "TEAM: efficient two-locus epistasis tests in human genome-wide association stud," *Bioinformatics (Oxford, England)*, vol. 26, pp. 217-227, Jun 2010.
- [16] X. Zhang, et al., "Fastanova: an efficient algorithm for genome-wide association study," in Proc. 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD), Las Vegas, Nevada, USA, pp. 821-829, 2008.
- [17] R. Paturi, et al., "The light bulb problem," in Proc. 2nd Annu. Workshop on Comput. Learning Theory, San Mateo, CA, pp. 261-268, 1989.
- [18] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," Proc. ACM Symposium on Theory of Computing (STOC), 2002.
- [19] M. Rabin, "Probabilistic Algorithms, Algorithms and Complexity," *Recent Results and New Directions, Academic Press*, pp. 21-39, 1976.
- [20] A. Mueen, et al., "Exact Discovery of Time Series Motifs," SIAM International Conference on Data Mining (SDM) 2009.
- [21] P. Beaudoin, et al., "Motion-Motif Graphs," *Symposium on Computer Animation*, 2008.
- [22] B. Chiu, et al., "Probabilistic discovery of time series motifs," Proc. of the 9th International Conference on Knowledge Discovery and Data mining (KDD), pp. 493-498, 2003.
- [23] T. Guyet, et al., "Knowledge construction from time series data using a collaborative exploration system," *Journal of Biomedical Informatics*, vol. 40, pp. 672-687, 2007.
- [24] J. Meng, et al., "Mining motifs from human motion," Proc. of EUROGRAPHICS, 2008.
- [25] D. Minnen, et al., "Discovering multivariate motifs using subsequence density estimation and greedy mixture learning," Proc. 22nd Conference on Artificial Intelligence (AAAI07), 2007.
- [26] M. S. Rahman, et al., "Indoor positioning by LED visible light communication and image sensors," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 1, 2011.

BIOGRAPHIES OF AUTHORS



Rhowel M. Dellosa obtained his Bachelor of Science degree in Computer Engineering from Saint Joseph Institute of Technology, Philippines in 2000. In 2002, he obtained a separate Bachelor of Science degree in Electronics and Communications Engineering from the same institution. He finished his Master's degree in Engineering with specialization in Computer Engineering in 2007 from the University of the City of Manila. He is currently taking up Doctor of Engineering with specialization in Computer Engineering from the Technological Institute of the Philippines in Quezon City, Philippines. His research interests include computational geometry, wireless sensor networks, statistical analysis and computer forensics.



Arnel C. Fajardo received his B.S. Electrical Engineering degree in 1991 from the Mapúa Institute of Technology in Intramuros, Manila. He obtained his M.S. Computer Science degree in 1999 from De La Salle University Manila. He obtained his Ph.D. in Computer Engineering degree from Hanbat National University, Daejeon, South Korea in 2014. He is the Senior Assistant Vice President and Research Director of Manuel L. Quezon University and Professorial lecturer of various universities in the Philippines. He is an author of more than 30 articles in journals and conference proceedings. His research interest includes speech recognition, image processing artificial intelligence and engineering education.



Ruji P. Medina is Dean of Graduate Programs of the Technological Institute of the Philippines in Quezon City, Philippines. He received his B.S. Chemical Engineering degree from the University of the Philippines – Diliman, Quezon City in 1992. He graduated summa cum laude from the Mapúa Institute of Technology in Intramuros, Manila in 2000 with a M.S. in Environmental Engineering degree. He holds a Ph.D. in Environmental Engineering from the University of the Philippines with sandwich program at the University of Houston, Texas. He counts among his expertise environmental and mathematical modeling, urban mining, and nanomaterials. Apart from his active role in research and graduate engineering education, he is an excellent technical mentor with numerous publications under his name.