

## Research Article

# Solving Dynamic Traveling Salesman Problem Using Dynamic Gaussian Process Regression

**Stephen M. Akandwanaho, Aderemi O. Adewumi, and Ayodele A. Adebisi**

*School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, University Road, Westville, Private Bag X 54001, Durban, 4000, South Africa*

Correspondence should be addressed to Stephen M. Akandwanaho; [stephen.akandwanaho@gmail.com](mailto:stephen.akandwanaho@gmail.com)

Received 4 January 2014; Accepted 11 February 2014; Published 7 April 2014

Academic Editor: M. Montaz Ali

Copyright © 2014 Stephen M. Akandwanaho et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper solves the dynamic traveling salesman problem (DTSP) using dynamic Gaussian Process Regression (DGPR) method. The problem of varying correlation tour is alleviated by the nonstationary covariance function interleaved with DGPR to generate a predictive distribution for DTSP tour. This approach is conjoined with Nearest Neighbor (NN) method and the iterated local search to track dynamic optima. Experimental results were obtained on DTSP instances. The comparisons were performed with Genetic Algorithm and Simulated Annealing. The proposed approach demonstrates superiority in finding good traveling salesman problem (TSP) tour and less computational time in nonstationary conditions.

## 1. Introduction

A bulk of research in optimization has carved a niche in solving stationary optimization problems. As a corollary, a flagrant gap has hitherto been created in finding solutions to problems whose landscape is dynamic, to the core. In many real-world optimization problems a wide range of uncertainties have to be taken into account [1]. These uncertainties have engendered a recent avalanche of research in dynamic optimization. Optimization in stochastic dynamic environments continues to crave for trailblazing solutions to problems whose nature is intrinsically mutable. Several concepts and techniques have been proposed for addressing dynamic optimization problems in literature. Branke et al. [2] delineate them through different stratifications, for example, those that ensure heterogeneity, sustenance of heterogeneity in the course of iterations, techniques that store solutions for later retrieval and those that use different multiple populations. The ramp up in significance of DTSP in stochastic dynamic landscapes has, up to the hilt, in the past two decades attracted a raft of computational methods, congenial to address the floating optima (Figure 1). An in-depth exposition is available in [3, 4]. The traveling salesman

problem (TSP) [5], one of the most thoroughly studied NP-hard theory in combinatorial optimization, arguably remains a main research experiment that has hitherto been cast as an academic guinea pig, most notably in computer science. It is also a research factotum that intersects with a wide expanse of research areas; for example, it is widely studied and applied by mathematicians and operation researchers on a grand scale. TSP's prominence ascribe to its flexibility and amenability to a copious range of problems. Gaussian process regression is touted as a sterling model on account of its stellar capacity to interpolate the observations, its probabilistic nature, versatility, practical and theoretical simplicity. This research lays bare a dynamic Gaussian process regression (DGPR) with a nonstationary covariance function to give foreknowledge of the best tour in a landscape that is subject to change. The research is in concert with the argumentation that optima are innately fluid, cognizant that size, nature, and position are potentially volatile in the lifespan of the optima. This skittish landscape, most notably in optimization, is a cue for fine-grained research to track the moving and evolving optima and provide a framework for solving a cartload of pent-up problems that are intrinsically dynamic. We colligate DGPR with nearest neighbor (NN) algorithm and the iterated

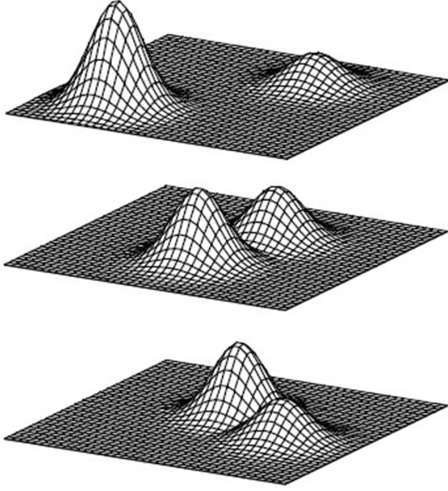


FIGURE 1: Nonstationary optima [6].

local search, a medley whose purpose is to refine the solution. We have arranged the paper in four sections. Section 1 is limited to introduction, Section 2's ambit includes review of all methods that form the mainspring of this work, which include Gaussian process, TSP, and DTSP. We elucidate DGPR for solving the TSP in Section 3. Section 4 discusses results obtained and draws conclusion.

## 2. The Traveling Salesman Problem (TSP)

The first researcher, in 1932, considered the traveling salesman problem [7]. Menger gives interesting ways of solving TSP. He lays bare the first approaches which were considered during the evolution of TSP solutions. An exposition on TSP history is available in [8–10].

*Basic Definitions and Notations.* It is imperative to note that in the gamut of TSP, both symmetric and asymmetric aspects are important threads in its fabric. We factor them into this work through the following expressions.

Basically, a salesman traverses across an expanse of cities culminating into a tour. The distance in terms of cost between cities is computed by minimizing the path length:

$$f(\pi) = \sum_{i=1}^{n-1} d_{\pi(i), \pi(i+1)} + d_{\pi(n), \pi(1)}. \quad (1)$$

We provide a momentary storage,  $D$  for cost distance. The distances between  $n$  cities are stored in a distance matrix  $D$ . For brevity, the problem can also be situated as an optimization problem. We minimize the tour length (Figure 5):

$$\sum_{i=1}^n d_{i, \pi(i)}. \quad (2)$$

The distance matrix of TSP has got certain features which come in handy in defining a set of classes for TSP [11]. If the city point,  $(x_i, y_i)$  in a tour is accentuated; then drawing from

Euclidean distance expression [11], we present the matrix  $C$  between separate distances as

$$c_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (3)$$

Affixed to TSP are important aspects that we bring to the fore in this paper. We adumbrate a brief overview of symmetric traveling salesman problem (STSP) and asymmetric traveling salesman problem (ATSP) as follows.

STSP, akin to its name, ensures symmetry in length. The distances between points are equal for all directions while ATSP typifies different distance sizes of points in both directions. Dissecting ATSP gives us a handle to hash out solutions.

Let ATSP be expressed, subject to the distance matrix. In combinatorial optimization, an optimal value is sought, whereby in this case, we minimize using the following expression:

$$w_{\pi(n), \pi(1)} + \sum_{i=1}^{n-1} w_{\pi(i), \pi(i+1)}. \quad (4)$$

Reference [12] formulates ATSP in integer programming  $n^2 - n$  zero-one variables  $x_{ij}$  or else it is defined as

$$y = \sum_{i=1}^n \sum_{j=1}^n w_{ij} x_{ij} \quad (5)$$

such that

$$\begin{aligned} \sum_{i=1}^n x_{ij} &= 1, \quad j \in [n], \\ \sum_{j=1}^n x_{ij} &= 1, \quad i \in [n], \end{aligned} \quad (6)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1, \quad \forall |S| < n,$$

$$x_{ij} = 0 \text{ or } 1, \quad i \neq j \in [n].$$

There are different rules affixed to ATSP, inter alia, to ensure a tour does not overstay its one-off visit to each vertex. The rules also ensure that standards are defined for subtours.

In the symmetry paradigm, the problem is postulated. For brevity, we present subsequent work with tautness:

$$y = \sum_{1 \leq i \leq j \leq n} w_{ij} x_{ij} \quad (7)$$

such that

$$\begin{aligned} \sum_{i=1}^n x_{ij} &= 2, \quad j \in [n], \\ \sum_{i \in S} \sum_{j \notin S} x_{ij} &\geq 2, \quad \forall 3 \leq |S| \leq \frac{n}{2}, \end{aligned} \quad (8)$$

$$0 \leq x_{ij} \leq 1, \quad i \neq j \in [n],$$

$$x_{ij} \quad \forall i \neq j \in [n].$$

TSP is equally amenable to the Hamiltonian cycle [11] and so we use graphs to ram home a different solution approach to the problem of traveling salesman. In this approach, we define  $G = (V, E)$  and  $(e_i \in E)w_i$ . This is indicative of the graph theory. The problem can be seen in the prism of a graph cycle challenge. Vertices and edges represent  $V$  and  $E$ , respectively.

It is also plausible to optimize TSP by adopting both an integer programming and linear programming approaches, pieced together in [13]:

$$\begin{aligned} \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij}, \\ \sum_{i=1}^n x_{ij} = 1, \\ \sum_{j=1}^n x_{ij} = 1. \end{aligned} \quad (9)$$

We can also view it with linear programming, for example,

$$\sum_{i=1}^m w_i x_i = w^T x. \quad (10)$$

Astounding ideas have sprouted, providing profound approaches in solving TSP. In this case, few parallel edges are interchanged. We use the Hamilton graph cycle [11] equality matrix

$$\begin{aligned} \forall i, j \ d_{ij} = d'_{ij}, \\ \sum_{i,j \in H} d_{ij} = \alpha \sum_{i,j \in H} d'_{ij} + \beta \end{aligned} \quad (11)$$

subject to  $\alpha > 0$ ,  $\beta \in \mathbb{R}$ . The common denominator of these methods is to solve city instances in a shortest time possible. A slew of approaches have been cobbled together extensively in optimization and other areas of scientific study. The last approach in this paper is to transpose asymmetric to symmetric. The early work of [14] explicates the concept. There is always a dummy city affixed to each city. The distances are the same between dummies and bona fide cities which makes distances symmetrical. The problem is then solved symmetrically thereby assuaging the complexities of NP-hard problems:

$$\begin{bmatrix} 0 & d_{12} & d_{13} \\ d_{21} & 0 & d_{23} \\ d_{31} & d_{32} & 0 \end{bmatrix} \longleftrightarrow \begin{bmatrix} 0 & \infty & \infty & -\infty & d_{21} & d_{31} \\ \infty & 0 & \infty & d_{12} & -\infty & d_{31} \\ \infty & \infty & 0 & d_{13} & d_{23} & -\infty \\ -\infty & d_{12} & d_{13} & 0 & \infty & \infty \\ d_{21} & -\infty & d_{23} & \infty & 0 & \infty \\ d_{31} & d_{32} & -\infty & \infty & \infty & 0 \end{bmatrix}. \quad (12)$$

**2.1. Dynamic TSP.** Different classifications of dynamic problems have been conscientiously expatiated in [15]. A wide array of dynamic stochastic optimization ontology ranges from a moving morphology to drifting landscapes. The dynamic optima exist owing to moving alleles in the natural

realm. Nature remains the fount of artificial intelligence. Optimization mimics the whole enchilada including the intrinsic floating nature of alleles, which provides fascinating insights into solving dynamic problems. Dynamic encoding problems were proposed by [16].

DTSP was initially introduced in 1988 by [17, 18]. In the DTSP, a salesman starts his trip from a city and after a complete trip, he comes back to his own city again and passes each city for once. The salesman is behooved to reach every city in the itinerary. In DTSP, cities can be deleted or added [19] on account of varied conditions. The main purpose for this trip is traveling the smallest distance. Our goal is finding the shortest route for the round trip problem.

Consider a city population,  $n$  and  $e$ , as the problem at hand where in this case we want to find the shortest path for  $n$  with a single visit on each. The problem has been modeled in a raft of prisms. A graph  $(N, E)$  with graph nodes and edges denoting routes between cities. For purpose of elucidation, the Euclidean distance between cities is  $i$  and  $j$  is calculated as follows [19]:

$$D_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (13)$$

**2.1.1. Objective Function.** The predictive function for solving the dynamic TSP is defined as follows.

Given a set of different costs  $(P_1, P_2, \dots, P_{n(t)})$ , the distance matrix is contingent upon time. Due to the changing routes in the dynamic setting, time is pivotal. So, it is expressed as a function of distance cost. The distance matrix has also been lucidly defined in the antecedent sections. Let us use the supposition that  $D = d_{ij}(t)$ , and  $i, j = 1, 2, \dots, n(t)$ . Our interest is bounded on finding the least distance from  $P_j$  and  $d_{ij}(t) = d_{ji}(t)$ . In this example, as aforementioned, time,  $t$  and of course, cost,  $d$ , play significant roles in the quality of the solution. DTSP is therefore minimized using the following expression:

$$d(T(t)) = \sum_{i=1}^{n(t)} d_{T_i, T_{i+1}}(t). \quad (14)$$

From Figures 2, 3, and 4, DTSP initial route is constructed upon visiting requests carried by the traveling salesman  $\{A, B, C, D, E\}$  [20]. As the traveling salesman sets forth, different requests  $(X, Y)$  come about which compels the traveling salesman to change the itinerary to factor in the new trip layover demands,  $\{A, B, C, D, X, E, Y\}$ .

**2.2. Gaussian Process Regression.** In machine learning, the primacy of Gaussian process regression cannot be overstated. The methods of linear and locally weighted regression have been outmoded by Gaussian process regression in solving regression problems. Gold mining was the major motivation for this method where Krige, whom Kriging is his brainchild [21], postulated that using posteriori, the cooccurrence of gold is encapsulated as a function of space. With Krige's interpolation mineral concentrations at different points can be predicted.

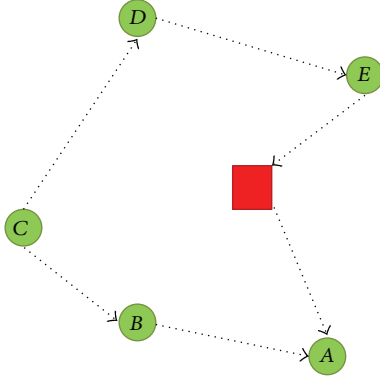


FIGURE 2: Initial request, A, B, C, D, E.

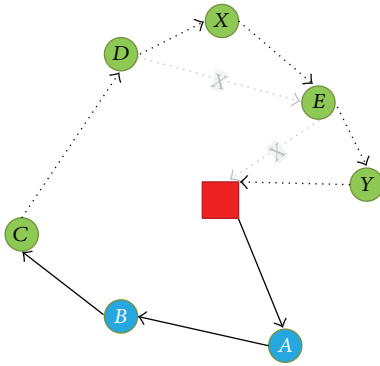


FIGURE 3: New requests for consideration.

In Gaussian process, we find a set of random variables. The specifications include covariance function  $n(x, x')$  and mean function  $p(x)$  that parameterize the Gaussian process. The covariance function determines the similarity of different variables. In this paper, we expand the ambit of study to nonstationary covariance:

$$p(f(x) \cdot f(x')) = N(\mu, \Sigma). \quad (15)$$

In the equation,  $\mu = \begin{pmatrix} \mu(x) \\ \mu(x') \end{pmatrix}$  and  $\Sigma = \begin{pmatrix} K(x, x) & K(x, x') \\ K(x', x) & K(x', x') \end{pmatrix}$ .

The matrices  $n \times 1$  for  $\mu$  and  $n \times n$  for  $\Sigma$  are presented in (15).

GPR (Figure 6) has been extensively studied across the expanse of prediction. This has resulted into different expressions to corroborate the method preference. In this study we have a constellation of training set  $P = (x^i, y^i)_{i=1}^m$ . The GPR model [22] then becomes

$$y^i = h(x^i) + \varepsilon^i \quad (16)$$

subject  $i = 1$  to  $m$ .

The probability density describes the likelihood for a certain value to be assumed by a variable. Given a set of observations bound by a number of parameters:

$$p(y | X, w) = \prod_{i=1}^n p(y_i | x_i, w) \sim N(X^T w, \sigma_n^2 I), \quad (17)$$

In this case, bias is denoted by  $w$ .

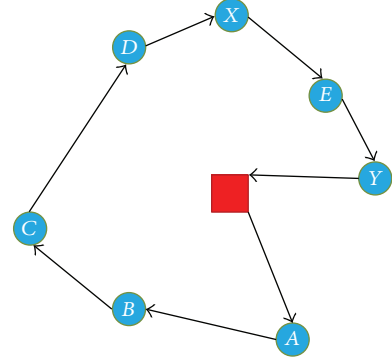


FIGURE 4: Previous route changed to meet new requests given to the traveling salesman.

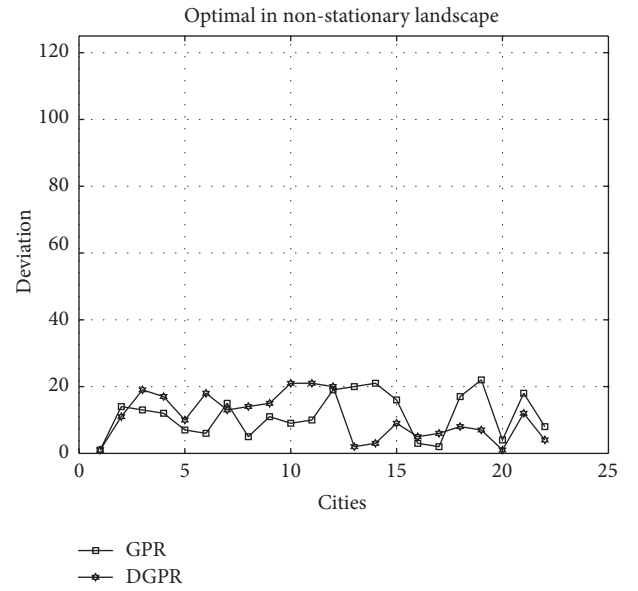


FIGURE 5: Minimum path generated by DGPR.

Gaussian process is analogous to Bayesian with a fractional difference [23]. In one of the computations by the Bayes' rule [23], is the Bayesian linear model parameterized by covariance matrix and mean denoted by  $A^{-1}$  and  $\bar{w}$ , respectively:

$$p(w | X, y) \sim N(\bar{w} = \sigma_n^{-2} A^{-1} X y, A^{-1}), \quad (18)$$

where

$$A = \sigma_n^{-2} X X^T + \sum_p^{-1}. \quad (19)$$

Using posterior probability, the Gaussian posterior is presented as

$$p(f_s | x_s, X, y) \sim N(\sigma_m^{-2} x_s^T A^{-1} X y, x_s^T A^{-1} x_s). \quad (20)$$

Also the predictive distribution, given the observed dataset,

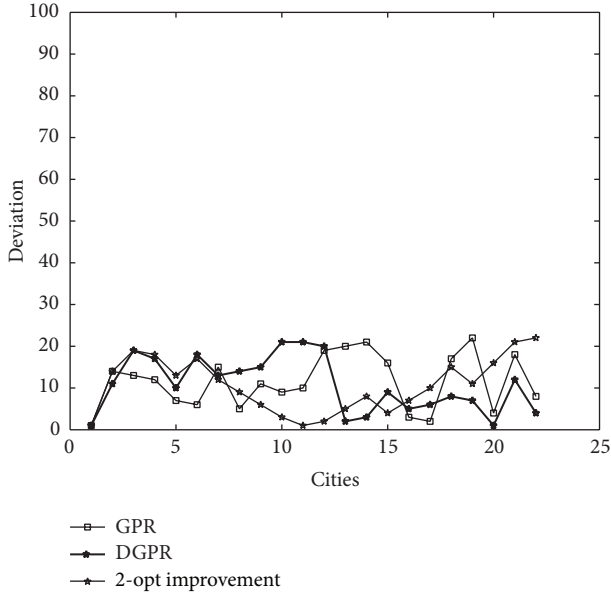


FIGURE 6: DGPR maintains superiority when juxtaposed with GPR and local search.

helps to model a probability distribution of an interval not estimating just a point:

$$p(f_s | x_s, X, y) \sim N(\sigma_m^{-2} \Phi_s^T A^{-1} \Phi y, \Phi_s^T A^{-1} \Phi_s), \quad (21)$$

where by  $\Phi = \Phi(X)$ ,  $\phi_s = \phi(x_s)$ , and  $A = \sigma_n^{-2} \Phi \Phi^T + \sum_p^{-1}$ . If  $A^{-1}$  of size  $n \times n$  is needed when  $n$  is large. We rewrite as

$$N \Phi_s^T \sum_p \Phi (K + \sigma_n^2 I)^{-1} y, \quad (22)$$

$$\phi_s^T \sum_p \phi_s - \phi_s^T \sum_p \Phi (K + \sigma_n^2 I)^{-1} \Phi^T \sum_p \phi_s.$$

The covariance matrix  $K$  is  $\Phi^T \sum_p \Phi$ .

**2.2.1. Covariance Function.** In simple terms, the covariance defines the correlation of function variables at a given time. A host of covariance functions for GPR have been studied [24]. In this example,

$$K(x_i, x_j) = v_0 \exp\left(-\left(\frac{x_i - x_j}{r}\right)^\sigma\right) + v_1 + v_2 \delta_{ij}, \quad (23)$$

the parameters are  $v_0$  (signal variance),  $v_1$  (variance of bias),  $v_2$  (noise variance),  $r$  (length scale), and  $\delta$  (roughness). However in finding solutions to dynamic problems, there is a mounting need for nonstationary covariance functions. The problem landscapes have increasingly become protean. The lodestar for this research is to use nonstationary covariance to provide an approach to dynamic problems.

A raft of functions have been studied. A simple form is described in [25]:

$$C^{NS}(x_i, x_j) = \sigma^2 |\Sigma_i|^{1/4} |\Sigma_j|^{1/4} \left| \frac{\Sigma_i + \Sigma_j}{2} \right|^{-1/2} \exp(-Q_{ij}), \quad (24)$$

With quadratic form,

$$Q_{ij} = (x_i - x_j)^T \left( \frac{\Sigma_i + \Sigma_j}{2} \right)^{-1} (x_i - x_j), \quad (25)$$

$\Sigma_i$  denotes the matrix of the covariance function.

### 3. Materials and Methods

Gaussian process regression method was chosen in this work, owing to its capacity to interpolate observations, its probabilistic nature, and versatility [26]. Gaussian process regression has considerably been applied in machine learning and other fields [27–29]. It has pushed back the frontiers of prediction and provided solutions to a mound of problems, for instance, making it possible to forecast in arbitrary paths and providing astounding results in a wide range of prediction problems. GPR has also provided a foundation for state of the art in advancing research in multivariate Gaussian distributions.

A host of different notations for different concepts are used throughout this paper:

- (i)  $T$  typically denotes the vector transpose,
- (ii)  $\hat{y}$  denotes the estimation,
- (iii) the roman letters typically denote what constitutes a matrix.

Our extrapolation is dependent on the training and testing datasets from the TSPLIB [30]. We adumbrate our approach as follows:

- (a) input distance matrix between cities,
- (b) invoke Nearest Neighbor method for tour construction,
- (c) tour encoding as binary for program interpretation,
- (d) as a drifting landscape, we set a threshold value  $\theta \in \mathbb{T}$ , where  $\mathbb{T}$  is the tour, and the error rate  $\varepsilon \in \mathbb{T}$  for the predicatability is

$$\forall_{1 \leq j \leq n} 0 < \text{severity } D_T(F_{ij}) \leq \theta, \quad (26)$$

$$\forall_{1 \leq j \leq n} 0 < \text{predict } D_{T,\varepsilon}(F_{ij}) \leq \theta,$$

- (e) get a cost sum,
- (f) determine the cost minimum and change to binary form,
- (g) present calculated total cost,
- (h) initialize the hyperparameters  $(\ell, \sigma_f^2, \sigma_n^2)$
- (i) we use the nonstationary covariance function  $K(X - X') = \sigma_o^2 + xx'$ . Constraints  $y_i = f(x_i + \varepsilon_i)$  realized in the TSP dataset,  $D = (x_i, y_i)_{i=1}^n$ ,  $y_i \in \mathbb{R}$  distances for different cities,  $x_i \in \mathbb{R}^d$ ,
- (j) calculate integrated likelihood in a dynamic regression,



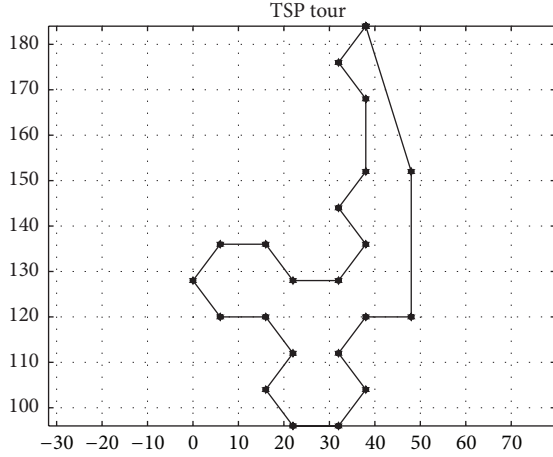


FIGURE 7: Generated tour in a drifting landscape for the best optimal route.

- (k) output the predicted optimal path  $\hat{x}^*$  and its length  $\hat{y}^*$ ,
- (l) implement the local search method  $x^*$ ,
- (m) estimate optimal tour  $\hat{x}^*$ ,
- (n) let the calculated route set the stage for iterations until no further need for refinement,
- (o) let the optimal value be stored and define the start for subsequent computations,
- (p) output optimal  $\hat{x}$  and cost ( $\hat{y}^*$ ).

**3.1. DTSP as a Nonlinear Regression Problem.** DTSP is formulated as a nonlinear regression problem. The nonlinear regression is part of the nonstationary covariance functions for floating landscapes [18]:

$$y_i = f(x_i + \varepsilon_i) \quad (27)$$

and  $D = \{(x_i, y_i)\}_{i=1}^n$  where  $y_i \in \mathbb{R}$ ,  $x_i \in \mathbb{R}^d$ . Our purpose is to define  $p(y^* | x^*, D)$ .

**3.1.1. Gaussian Approximation.** The Gaussian approximation is premised on the kernel, an important element of GPR.

The supposition for this research is that once  $x$  is known,  $y$  can be determined. By rule of thumb, the aspects of a priori (when the truth is patent, without need for ascertainment) and posteriori (when there is empirical justification for the truth or the fact is buttressed by certain experiences) play a critical role in shaping an accurate estimation. The kernel determines the proximate between estimated and nonestimated.

Nonstationarity, on the other hand, means that the mean value of a dataset is not necessarily constant and/or the covariance is anisotropic/varies with direction and spatially variant, as seen in [31]. We have seen a host of nonstationary

kernels in literature as discussed in previous chapters, for example in [32],

$$C^{\text{NS}}(x_i, x_j) = \int_{\mathcal{R}^2} K_{x_i}(u) K_{x_j}(u) du. \quad (28)$$

For  $(x_i, x_j, u) \in \mathcal{R}^2$ ,

$$f(x) = \int_{\mathcal{R}^2} K_x(u) \psi(u) du, \quad (29)$$

For  $\mathcal{R}^p$ ,  $p = 1, 2, \dots$ , we ensure a positive definite function between cities for dynamic landscapes:

$$\begin{aligned} & \sum_{i=1}^n \sum_{j=1}^n a_i a_j C^{\text{NS}}(x_i, x_j) \\ &= \sum_{i=1}^n \sum_{j=1}^n a_i a_j \int_{\mathcal{R}^p} K_{x_i}(u) K_{x_j}(u) du \\ &= \int_{\mathcal{R}^p} \sum_{i=1}^n a_i K_{x_i}(u) \sum_{j=1}^n a_j K_{x_j}(u) du \\ &= \int_{\mathcal{R}^p} \left( \sum_{i=1}^n a_i K_{x_i}(u) \right)^2 du \geq 0. \end{aligned} \quad (30)$$

In mathematics, convolution knits two functions to form another one. This cross relation approach has been successfully applied myriadly in probability, differential equations, and statistics. In floating landscapes, we see convolution at play which produces [31]

$$C^{\text{NS}}(x_i, x_j) = \sigma^2 \sigma^2 |\Sigma_i|^{1/4} |\Sigma_j|^{1/4} \left| \frac{\Sigma_i + \Sigma_j}{2} \right|^{-1/2} \exp(-\mathfrak{Q}_{ij}). \quad (31)$$

In mathematics, a quadratic form reflects the homogeneous polynomial expressed in

$$\mathfrak{Q}_{ij} = (x_i - x_j)^T \left( \frac{\Sigma_i + \Sigma_j}{2} \right)^{-1} (x_i - x_j). \quad (32)$$

A predictive distribution is then defined:

$$\begin{aligned} p(y^* | X^*, D, \theta) &= \int \int p(y^* | X^*, D, \exp(\ell^*), \exp(\ell), \theta_y) \\ &\quad \times p(\ell^*, \ell | X^*, X, \bar{\ell}, \bar{X}, \theta_\ell) d\ell d\ell^*. \end{aligned} \quad (33)$$

From the dataset, the most probable estimates are used, with the following equation:

$$p(y^* | X^*, D, \theta) \approx p(y^* | X^*, \exp(\ell^*), \exp(\ell), D, \theta_y). \quad (34)$$

**3.2. Hyperparameters in DGPR.** Hyperparameters define the parameters for the prior probability distribution [6]. We use  $\theta$  to denote the hyperparameters. From  $y$ , we get  $\theta$  that optimizes the probability to the highest point:

$$p(y | X, \theta) = \int p(y | X, \ell, \theta_y) \cdot p(\ell | X, \bar{\ell}, \bar{X}, \theta_{\bar{\ell}}) d\ell. \quad (35)$$

From the hyperparameters  $p(y | X, \theta)$ , we optimally define the marginal likelihood and introduce an objective function for floating matrix:

$$\begin{aligned} \log p(y | X, \exp(\ell), \theta_y) \\ = -\frac{1}{2} y^T (K_{x,x} + \sigma_n^2 I)^{-1} y \\ - \frac{1}{2} \log \{K_{x,x} + \sigma_n^2 I\} - \frac{n}{2} \log(2\pi), \end{aligned} \quad (36)$$

and  $|M|$  is the factor of  $M$ .

In this equation the objective function is expressed as

$$\begin{aligned} L(\theta) = \log p(\ell | y, X, \theta) = c_1 + c_2 \\ \cdot [y^T A^{-1} y + \log |A| + \log |B|] \end{aligned} \quad (37)$$

and  $A$  is  $K_{x,x} + \sigma_n^2 I$ ,  $B$  is  $K_{\bar{x},\bar{x}} + \sigma_n^2 I$ .

The nonstationary covariance  $K_{x,x}$  is defined as follows.  $\ell$  represents the cost of  $X$  point:

$$K_{x,x} = \sigma_f^2 \cdot P_r^{1/4} \cdot P_c^{1/4} \cdot \left(\frac{1}{2}\right)^{-1/2} P_s^{-1/2} \cdot E \quad (38)$$

with

$$\begin{aligned} P_r &= p \cdot 1_n^T, \\ P_c &= 1_n^T \cdot p^T, \\ P &= \ell^T \ell, \\ P_s &= P_r + P_c, \\ E &= \exp \frac{[-s(X)]}{P_s}, \\ \ell &= \exp \left[ \bar{K}_{x,\bar{x}}^T [\bar{K}_{\bar{x},\bar{x}} + \sigma_n^2 I]^{-1} \bar{\ell} \right]. \end{aligned} \quad (39)$$

After calculating the nonstationary covariance, we then make predictions [33]:

$$K_{x,\bar{x}} = \bar{\sigma} f^2 \cdot \exp \left[ -\frac{1}{2} s(\bar{\sigma}, \ell^{-2} X, \bar{\sigma}, \ell^{-2} \bar{X}) \right]. \quad (40)$$

## 4. Experimental Results

We use the Gaussian Processes for Machine Learning Matlab Toolbox. Its copious set of applicability dovetails with the purpose for this experiment. It was titivated to encompass all the functionalities associated with our study. We used

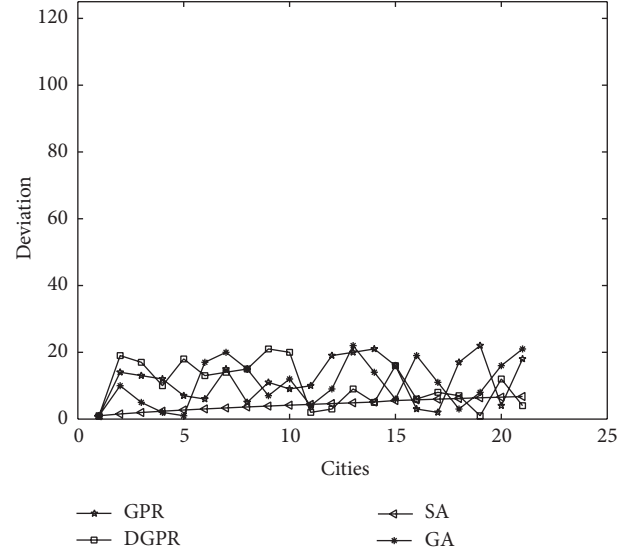


FIGURE 8: DGPR is juxtaposed with all other comparing methods in an instance of 22 cities for 200 iterations.

Matlab due to its robust platform for scientific experiments and sterling environment for prediction [26]. 22-city data instance were gleaned from the TSP library [34].

From the Dell computer, we set initial parameters:  $\ell = 2$ ,  $\sigma_f^2 = 1$ ,  $\sigma_n^2$ . The dynamic regression is lumped with the local search method to banish the early global and local convergence issues.

For global method (GA), the following parameters are defined. Sample = 22,  $(p_c) = 1$ ,  $(p_m) = 1.2$ , and 100 computations while SA parameters include  $T_{\text{int}} = 100$ ,  $T_{\text{end}} = 0.025$ , and 200 computations.

The efficacy level is always observed by collating the estimated tour with the nonestimated [35–37]:

$$\text{deviation (\%)} = \frac{\hat{y}^* - y^*}{y^*} \times 100. \quad (41)$$

The percentage of difference between estimated solution and optimal solution = 16.64%, which is indicative of a comparable reduction with the existing methods (Table 1). The computational time by GPR is 4.6402 and distance summation of 253.000. The varied landscape dramatically changes the length of travel for the traveling salesman. The length drops a notch suggestive of a better method and an open sesame for the traveling salesman to perform his duties.

The proposed DGPR (Figure 8) was fed with the sample TSP tours. The local search method constructs the initial route and 2-opt method used for interchanging edges. The local method defines starting point and all ports of call to painstakingly ensure that the loop goes to every vertex once and returns to the starting point. The 2-opt vertex interchange creates a new path through exchange of different vertices [38]. Our study is corroborated by less computation time and slumped distance when we subject TSP to predicting the optimal path. The Gaussian process runs on the shifting sands of landscape through dynamic instances. The nonstationary

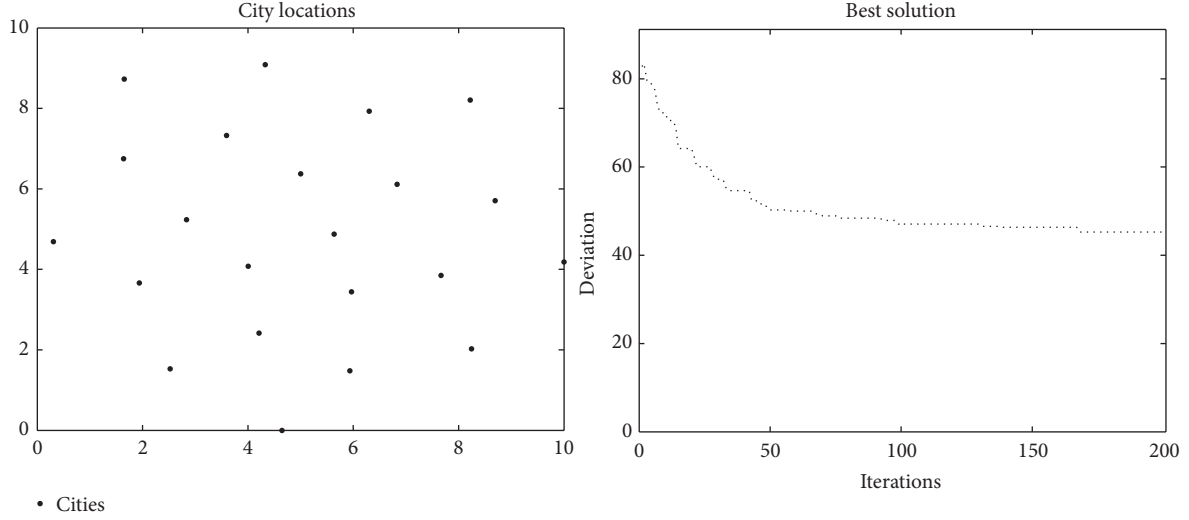


FIGURE 9: An example of best solution in stationarity. A sample of 22 cities generates a best route. As seen in the figure, there is a difference in optimality and time with nonstationarity.

TABLE 1: The extracted data is collated to show forth variations in different methods.

Method#	DTSP and DGPR collated data			
	Nodes#	Optimal#	T#	D#
GPR	22	253.00	4.64	0.42
DGPR	22	231.00	3.82	0.24
GA	22	288.817	5.20	0.40
SA	22	244.00	5.50	2.30
2-opt	22	240.00	4.20	0.30

functions described before brings to bare the residual, the similitude, between actual and estimate. In the computations, path is interpreted as  $\lceil \log_2 n \rceil$  and an ultimate route as  $n \lceil \log_2 n \rceil$ .

There are myriad methods over and above Simulated (Figure 10) Annealing and tabu search, set forth by the fecundity of researchers in optimization. The cost information determines the replacement of path in a floating turf. The lowest cost finds primacy over the highest cost. This process continues in pursuit of the best route (Figure 9) that reflects the lowest cost. In the dynamic setting, as the ports of call change, there is a new update on the cost of the path. The cost is always subject to change. The traveling salesman is desirous to travel the shortest distance which is the crux of this study (Figure 11). In the weave of this work, the dynamic facet of regression remains at the heartbeat of our contribution. The local methods are meshed together to ensure quality of the outcome. As a corollary our study has been improved with the integration of the Nearest Neighbor algorithm and the iterated 2-opt search method. We use the same number of cities; each tour is improved by 2-opt heuristics and the best result is selected.

In dynamic optimization, a complete solution of the problem at each time step is usually infeasible due to the floating optima. As a consequence, the search for exact global

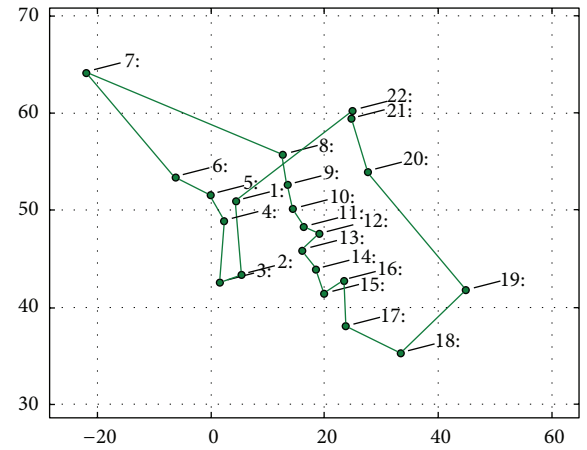


FIGURE 10: Optimal path is generated for Simulated Annealing in 22 cities.

optima must be replaced again by the search for acceptable approximations. We generate a tour for the nonstationary fitness landscape in Figure 7.

## 5. Conclusion

In this study, we use a nonstationary covariance function in GPR for the dynamic traveling salesman problem. We predict the optimal tour of 22 city dataset. In the dynamic traveling salesman problem where the optima shift due to environmental changes, a dynamic approach is implemented to alleviate the intrinsic maladies of perturbation. Dynamic traveling salesman problem (DTSP), as a case of dynamic combinatorial optimization problem, extends the classical traveling salesman problem and finds many practical importance in real-world applications, inter alia, traffic jams,



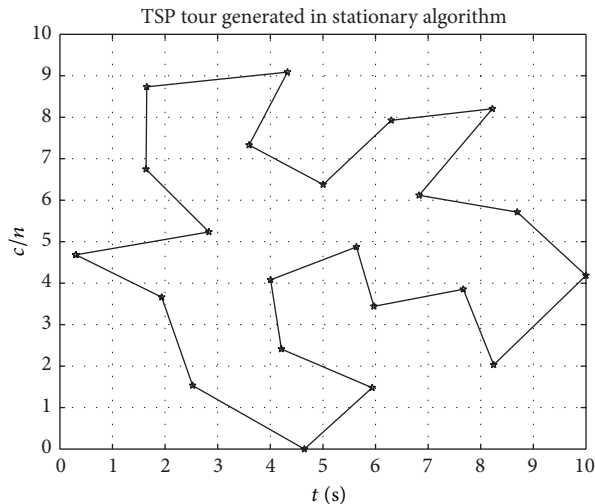


FIGURE 11: High amount of time and distance cost are needed to complete the tour vis-a-vis when prediction is factored.

network load-balance routing, transportation, telecommunications, and network designing. Our study produces a good optimal solution with less computational time in a dynamic environment. A slump in distance corroborates the argumentation that prediction brings forth a leap in efficacy in terms of overhead reduction, a robust solution born out of comparisons, that strengthen the quality of the outcome. This research foreshadows and gives interesting direction to solving problems whose optima are mutable. DTSP is calculated by the dynamic Gaussian process regression, cost predicted, local methods invoked, and comparisons made to refine and fossilize the optimal solution. MATLAB was chosen as the platform for the implementation, because development is straightforward with this language and MATLAB has many comfortable tools for data analysis. MATLAB also has an extensive cross-linking architecture and can interface directly with Java classes. The future of this research should be directed to design new nonstationary covariance functions to increase the ability to track dynamic optima. Also changes in size and evolution of optima should be factored in, over, and above changes in location.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgments

The authors would like to acknowledge W. Kongkaew and J. Pichitlamken for making their code accessible which became a springboard for this work. Special thanks also go to the Government of Uganda for bankrolling this research through the PhD grant from Statehouse of the Republic of Uganda. The authors also express their appreciation to the incognito reviewers whose efforts were telling in reinforcing the quality of this work.

### References

- [1] A. Simões and E. Costa, "Prediction in evolutionary algorithms for dynamic environments," *Soft Computing*, pp. 306–315, 2013.
- [2] J. Branke, T. Kaussler, H. Schmeck, and C. Smidt, *A Multi-Population Approach to Dynamic Optimization Problems*, Department of Computer Engineering, Yeditepe University, Istanbul, Turkey, 2000.
- [3] K. S. Leung, H. D. Jin, and Z. B. Xu, "An expanding self-organizing neural network for the traveling salesman problem," *Neurocomputing*, vol. 62, no. 1–4, pp. 267–292, 2004.
- [4] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.
- [5] C. Jarumas and J. Pichitlamken, "Solving the traveling salesman problem with gaussian process regression," in *Proceedings of the International Conference on Computing and Information Technology*, 2011.
- [6] K. Weicker, *Evolutionary Algorithms and Dynamic Optimization Problems*, University of Stuttgart, Stuttgart, Germany, 2003.
- [7] K. Menger, *Das botenproblem. Ergebnisse Eines Mathematischen Kolloquiums*, 1932.
- [8] G. Gutin, A. Yeo, and A. Zverovich, "Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP," *Discrete Applied Mathematics*, vol. 117, no. 1–3, pp. 81–86, 2002.
- [9] A. Hoffman and P. Wolfe, *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, Wiley, Chichester, UK, 1985.
- [10] A. Punnen, "The traveling salesman problem: applications, formulations and variations," in *The Traveling Salesman Problem and Its Variations*, Combinatorial Optimization, 2002.
- [11] E. Ozcan and M. Erenturk, *A Brief Review of Memetic Algorithms for Solving Euclidean 2d Traveling Salesrep Problem*, Department of Computer Engineering, Yeditepe University, Istanbul, Turkey.
- [12] G. Clarke and J. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Operations Research*, vol. 12, no. 4, pp. 568–581, 1964.
- [13] P. Miliotis, *Integer Programming Approaches To the Traveling Salesman Problem*, University of London, London, UK, 2012.
- [14] R. Jonker and T. Volgenant, "Transforming asymmetric into symmetric traveling salesman problems," *Operations Research Letters*, vol. 2, no. 4, pp. 161–163, 1983.
- [15] S. Gharan and A. Saberi, *The Asymmetric Traveling Salesman Problem on Graphs with Bounded Genus*, Springer, Berlin, Germany, 2012.
- [16] P. Collard, C. Escazut, and A. Gaspar, "Evolutionary approach for time dependent optimization," in *Proceedings of the IEEE 8th International Conference on Tools with Artificial Intelligence*, pp. 2–9, November 1996.
- [17] H. Psaraftis, "Dynamic vehicle routing problems," *Vehicle Routing: Methods and Studies*, 1988.
- [18] M. Yang, C. Li, and L. Kang, "A new approach to solving dynamic traveling salesman problems," in *Simulated Evolution and Learning*, vol. 4247 of *Lecture Notes in Computer Science*, pp. 236–243, Springer, Berlin, Germany, 2006.
- [19] S. S. Ray, S. Bandyopadhyay, and S. K. Pal, "Genetic operators for combinatorial optimization in TSP and microarray gene ordering," *Applied Intelligence*, vol. 26, no. 3, pp. 183–195, 2007.

- [20] E. Osaba, R. Carballedo, F. Diaz, and A. Perallos, "Simulation tool based on a memetic algorithm to solve a real instance of a dynamic tsp," in *Proceedings of the IASTED International Conference Applied Simulation and Modelling*, 2012.
- [21] R. Battiti and M. Brunato, *The Lion Way, Machine Learning Plus Intelligent Optimization. Applied Simulation and Modelling*, Lionsolver, 2013.
- [22] D. Chuong, *Gaussian Processes*, Stanford University, Palo Alto, Calif, USA, 2007.
- [23] W. Kongkaew and J. Pichitlamken, *A Gaussian Process Regression Model For the Traveling Salesman Problem*, Faculty of Engineering, Kasetsart University, Bangkok, Thailand, 2012.
- [24] C. Rasmussen and C. Williams MIT Press, Cambridge, UK, 2006.
- [25] C. J. Paciorek and M. J. Schervish, "Spatial modelling using a new class of nonstationary covariance functions," *Environmetrics*, vol. 17, no. 5, pp. 483–506, 2006.
- [26] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (GPML) toolbox," *Journal of Machine Learning Research*, vol. 11, pp. 3011–3015, 2010.
- [27] F. Sinz, J. Candela, G. Bakir, C. Rasmussen, and K. Franz, "Learning depth from stereo," in *Pattern Recognition*, vol. 3175 of *Lecture Notes in Computer Science*, pp. 245–252, Springer, Berlin, Germany, 2004.
- [28] J. Ko, D. J. Klein, D. Fox, and D. Haehnel, "Gaussian processes and reinforcement learning for identification and control of an autonomous blimp," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pp. 742–747, Roma, Italy, April 2007.
- [29] T. Idé and S. Kato, "Travel-time prediction using gaussian process regression: a trajectory-based approach," in *Proceedings of the 9th SIAM International Conference on Data Mining 2009 (SDM '09)*, pp. 1177–1188, May 2009.
- [30] G. Reinelt, "Tsp lib discrete and combinatorial optimization," 1995, <https://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>.
- [31] C. Paciorek, *Nonstationary gaussian processes for regression and spatial modelling [Ph.D. thesis]*, Carnegie Mellon University, Pittsburgh, Pa, USA, 2003.
- [32] D. Higdon, J. Swall, and J. Kern, *Non-Stationary Spatial Modeling*, Oxford University Press, New York, NY, USA, 1999.
- [33] C. Plagemann, K. Kersting, and W. Burgard, "Nonstationary Gaussian process regression using point estimates of local smoothness," in *Machine Learning and Knowledge Discovery in Databases*, vol. 5212 of *Lecture Notes in Computer Science*, no. 2, pp. 204–219, Springer, Berlin, Germany, 2008.
- [34] G. Reinelt, "The tsp lib symmetric traveling salesman problem instances," 1995.
- [35] J. Kirk, *Matlab Central*.
- [36] X. Geng, Z. Chen, W. Yang, D. Shi, and K. Zhao, "Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search," *Applied Soft Computing Journal*, vol. 11, no. 4, pp. 3680–3689, 2011.
- [37] A. Seshadri, *Traveling Salesman Problem (Tsp) Using Simulated Annealing*, IEEE, 2006.
- [38] M. Nuhoglu, Shortest path heuristics (nearest neighborhood, 2 opt, farthest and arbitrary insertion) for travelling salesman problem, 2007.