THE 7TH INTERNATIONAL CONFERENCE ON COMPUTING AND INFORMATION TECHNOLOGY

PROCEEDINGS OF IC²IT 2011

THE 7TH INTERNATIONAL CONFERENCE ON COMPUTING AND INFORMATION TECHNOLOGY

11-12 MAY 2011

FACULTY OF INFORMATION TECHNOLOGY
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY NORTH BANGKOK (KMUTNB)
BANGKOK, THAILAND

WWW. IC2IT.ORG

IEEE
THAILAND SECTION

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S UNIVERSITY OF TECHNOLOGY NORTH BANGKOK

Solving the Traveling Salesman Problem with Gaussian Process Regression

Jarumas Chantapanich
International Graduate Program in Industrial Engineering
Department of Industrial Engineering
Kasetsart University, Bangkok, Thailand
jasom145@hotmail.com

Juta Pichitlamken
International Graduate Program in Industrial Engineering
Department of Industrial Engineering
Kasetsart University, Bangkok, Thailand
juta.p@ku.ac.th

Abstract — **We present a new heuristic method for solving Traveling salesman problem which is NP-hard. Given a small set of data, we first fit a Gaussian process regression function and then find a route that minimizes this regression function. The route is further transform into a TSP tour. The numerical experiment shows that our approach can find a reasonably good solution.**

Keywords-Traveling salesman problem; Gaussian process regression.

I. INTRODUCTION

The traveling salesman problem (TSP) can simply be stated as: if a traveling salesman wishes to visit each city exactly once [1]. The problem is widely studied by mathematicians and operation researchers because it is commonly applied on real world problems, such as finding minimum distance on logistic problems [2] and optimizing production sequence for scheduling problems [3].

TSP is a representative of a larger class of problems known as combinatorial optimization problems. The class of TSP is NP-complete. Thus, the running times for any heuristic algorithms to solve the TSP increases exponentially with the number of cities [4].

Although the problem is difficult, a large number of heuristics methods perform well; some instances with many thousands of cities can be solved. In 1994, Applegate, Bixby, Chvátal and Cook [5] solve a traveling salesman problem which models the production of printed circuit boards having 7,397 holes. Later, they solve another problem with over the 13,509 largest cities in the U.S. .

Gaussian process regression (GPR) provides a powerful methodology for modeling data that exhibit complex characteristics such as nonlinear behaviors while retaining mathematical simplicity. Gaussian process is a collection of random variables, any finite number of which has (consistent) Gaussian distribution. An example of Gaussian process applications is in prediction control [6].

We employ some random tours (independent variables) and total costs or total distances (dependent variables) to generate a total cost function. Therefore, TSP can be viewed as a regression problem. The relationship between dependent variables and independent variables are likely to be nonlinear; thus, multiple linear regressions cannot be applied. GPR is capable of fitting arbitrary-shaped functions, so it is selected to fit a response function for TSP.

II. LITERATURE REVIEW

Development of TSP and related works on GPR are described as follows: Sections A and C. We describe back ground theory GPR on Sections B.

A. Traveling Salesman Problems

The classical TSP is symmetric. Lui, Ng and Ong [7] propose a heuristic for the classical symmetric TSP. Their method is to split a TSP tour into overlapped blocks and then improve each block separately. By doing a local search using the Generalized Crossing method, each block is explored intensively in order to improve the existing solution. When comparing with an adaptive neural network method, this algorithm obtains a better solution [7].

The constraint for TSP is not only to visit all the cities exactly once but sometimes TSP also adds other conditions on distance or cost such as the Orienteering and Discounted-Reward TSP, where both are NP-hard [8]. The goal of the Orienteering TSP is to find the path with maximum reward collected, subject to a hard limit on total distance. While in the Discounted-Reward TSP, the length limit is given a discount factor in order to maximize total discount reward collected [8].

TSP is applied to many real-world situations. One of popular situations is when cities can be dynamically added or removed. Varga, Chira, and Dumitrescu [9] develop their agent approach to solve this problem. Proposed Multi-agent approach is based on the Sensitive Stigmergic Agent System model refined with new types of messages between agents. The agent sends messages every time change occurs, for instance, when an agent observes that the city has disappeared or appeared. After testing under various pheromone sensitivity levels and learning abilities for agents, the proposed model have good performance [9].

Hasegawa [10] shows that TSP can be applied to complex physical problems. The temperature cycling experiments is formulated as a random Euclidean TSP and is solved with the Metropolis algorithm.

B. Background Theory on GPR

Gaussian process is a collection of random variables, any finite number of which has (consistent) Gaussian distributions. Multivariate Gaussian distribution is specified by a mean vector μ and covariance matrix Σ .

$$f_i = (f_1, \quad f_n)^T \sim N(\mu, \Sigma). \tag{1}$$

A Gaussian processes are parameterized by a mean function $\mu(x)$ and covariance function $k(x, x')$ as follow:

$$f(x) \sim \mathcal{GP}(\mu(x), k(x, x')). \tag{2}$$

A normal linear regression model is

$$y = f(x) + \varepsilon \,, \quad \text{where } \varepsilon \square \; N(0, \sigma^2). \tag{3}$$

A regression model in terms of GPR is

$$Y^T \sim N(\mu(x), k(x, x')). \tag{4}$$

The model of y as a noise realization of μ is $p(y|\mu) = N(y|\mu, \sigma_n^2)$. Define $X = (x_1, \ldots, x_n)$, $[k_*]_i = k(x_*, x_i)$ and $[K_{xx}]_{ij} = k(x_i, x_j)$. When x^* is a test point and latent function $\mu_* = \mu(x_*)$

$$\mu_* = k_*(K_{xx} + \sigma_n^2 I)^{-1} Y^T, \tag{5}$$

$$\sigma_*^2 = k(x_*, x_*) - k_*^T (K_{xx} + \sigma_n^2 I)^{-1} k_*, \tag{6}$$

$$K_{yy} = K_{xx} + \sigma_n^2 I. \tag{7}$$

A maginal likelihood of GPR is

$$\log(p(Y|X)) \propto -Y^T K_{yy}^{-1} Y - \log|K_{yy}| + c, \tag{8}$$

where c is a constant that is independent of the hyperparameters [11].

C. GPR Extensions

Ebden [12] illustrates the GPR concept in a typical prediction problem. Given a set of random variables Y, he explains that the behavior of Y can be described by an underlying function $f(x)$ through the relation $Y = f(x) + N(0, \Sigma)$, where $N(0, \Sigma)$ is a normal random vector with mean of zero and covariance matrix Σ. Statistical methods can be used to approximate $E(Y|x^*)$ by estimating $f(x)$ from the given set Y.

Sollich and Williams [13] use the equivalent kernel (EK) to understand GPR for large sample sizes based on continuum limit. They use EK to estimate learning curves fo GPR. EK provides a simple means to understand the learni curve of the behavior of GPR, even in the case where t learner's covariance function is not well matched with t structure of the target function.

Normally, Gaussian process terms have single outp with a stationary covariance function and continuities becau the covariance matrix must be positive definite. Meed an Osindero [14] develop an infinite mixture model for mul model outputs of Gaussian processes with non-stationa covariance functions, discontinuities, multimodality a overlapping output signals. An infinite mixture model is f generative. This model is shown to be better than Rasmuss and Ghahramni [15] model which is a conditional model.

Boyle and Frean [16] present an alternative to achie Gaussian process model with multiple outputs by treating Gaussian process with white noise convolved sources w smoothing kernels, and parameterizing the kernel instead.

Generally, GPR inputs must be statistically independe Williams [17] presents GPR with noise whose varianc depend on input. They use a natural non-parametric prior f variable noise rates and give an effective method of sampli the posterior distribution by using the Markov chain Mo Carlo. When applied to the data set with varying noise, t posterior noise rates obtained are well matched to the know structure.

III. PROPOSED GPR FOR TSP

We use GPR technique for creating a prediction functi and solving the TSP. We define our notations as follows:

y_r = total distance of sampling tours r, $r = 1, 2, 3, \ldots, R$

$x_{r,i}$ = path i of the sampling tour r,

$x_{r,i}$ = $\begin{cases} 0: \text{if path } i \text{ does not exist on tour } r \\ 1: \text{if path } i \text{ exists on tour } r \end{cases}$

a_i = distance of path i.

Our objective function is

$$\text{Min } y_r = \sum_{i=1}^{m} a_i x_{r,i}.$$

We apply GPML toolbox developed by Rasmussen a Williams [18] to fit the GPR function and optimize prediction function. GPML toolbox defines Equation (5) term of α below

$$\alpha = (K_{xx} + \sigma_n^2 I)^{-1} Y^T,$$

thus prediction term is

$$\mu_* = k_* \alpha. \quad (11)$$

The objective function for applied GPR on TSP is

$$\text{Min } \mu_* = k_* \alpha, \quad (12)$$

where x^* is a TSP tour.

GPML toolbox use GPR with a squared exponential covariance function and allow a separate length scale for each input with ARD (Automatic relevance determination) and independent noise. The Equation shown below:

$$k_* = \sigma_f^2 \exp(-\frac{1}{2}(x_* - x_i)^T M (x_* - x_i)) + \sigma_n^2 \delta_{*,i}, \quad (13)$$

where σ_f is a hyperparameter, δ_{pq} is an indicator function ($\delta_{pq} = 1$ if $p = q$ and 0 otherwise), $M = diag(\ell)^{-2}$, and ℓ is a vector of positive value hyperparameter. The steps to apply GPR for TSP are explained in details below:

A. Construct a subtour by a greedy heuristic

When the number of nodes in a TSP decreases, α as defined can be calculated faster. To reduce the size of the search space or reduce the number of nodes, we construct a subtour with n_f nodes. First, we search a distance matrix to find a minimum distance a_i. Then these two nodes are connected to form a subtour and more nodes with minimum distance are added until we get n_f nodes.

B. Generate sampling tours and their corresponding total distance

Tours and their corresponding total distance are generated by starting with a subtour of size n_f. We randomly permute the remaining $n - n_f$ nodes, by using "Randperm" function in MATLAB, to form a complete tour. We repeat this step R times. Sample tours are combined to form a data matrix X, whose dimension is $[R \times m]$, $m = \sum_{i=1}^{n-1}(n-i)$ and its corresponding total distance into a response vector Y, whose dimension is $[R \times 1]$.

C. Use a GPR function to approximate an optimal TSP tour

A GPR function is created from sampling tours and their corresponding total distance for predicting a minimum total distance. The steps are described as follows:

- Determine a starting solution, which is a TSP tour, for a GPML toolbox function "minimize".
- Estimate parameters α and covariance function k_* in Equations (11) and (13) from sampling tours and their corresponding total distance.
- Using a GPR function, determine a route with minimum distance μ.

D. Transform an optimal solution into a TSP tour

The route we obtain in Part C may not be a TSP tour, so we need to do a tour construction from it. We first build disjoint subtours and connect them to form a TSP tour. The most obvious subtour is what we have created in Part IIIA. The remaining subtours are constructed by a greedy heuristic, that is selecting links with small distance first. More details can be found in Chantapanich's thesis [19].

IV. NUMERICAL EXPERIMENTS

We fist describe details of test problems in Section A and the results of experiments are shown in Section B.

A. Test problems

We use the geographical TSP below to be test problems [20]:

1. Ulysses22: 22 cities with a possible number of solution of 1.12×10^{21} (22!) with the optimal solution of 6,945.2 km.
2. Ulysses16: 16 cities with a possible number of solution of 2.09×10^{13} (16!) with the optimal solution of 6,795.8 km.
3. Burma14: 14 cities with a possible number of solution of 8.71×10^{10} (14!) with the optimal solution of 3,356.1 km.

In the experiment, sample sizes of R are 500, 1500, 2500 and 3500.

B. Results and discussion

In our experiment, we have two control parameters: n_f is number of nodes in a subtour and R is the number of random tours in a sample for creating a GPR function. The results are shown in Tables I-II and Fig. 1.

Table I shows the predicted minimum distance from the GPR functions, but these tours may not be form TSP tours. The underlined numbers are minimum distance for a given n_f. The best cases seem to be when the sample size is largest (R =3500). We can conclude that when R and n_f increase our method seem to predict a better solution. The best scenarios for each test problems in Table I are close to the true optimal solutions, so we can use these lowest predictions to be our target to find TSP tours as shown in Table II.

TABLE I. PREDICTED MINIMUM DISTANCE FROM A GPR FUNCTION BEFORE BEING TRANSFORMED TO TSP TOURS.

n_f	R = 500	1,500	2,500	3,500
Ulysses22				
0	12,493.4	12,246.8	11,966.7	11,694.5
2	11,193.1	11,188.5	10,584.0	10,579.6
4	10,175.1	9,578.1	9,565.3	10,137.2
6	9,929.3	8,869.2	9,398.7	9,013.7
10	8,316.8	9,162.2	9,131.3	7,043.2
12	6,699.8	6,697.2	6,379.9	9,138.9
14	9,222.8	6,137.8	9,217.5	6,093.0
Ulysses16				
0	9,080.1	10,308.0	9,829.0	9,289.8
2	9,180.7	8,362.2	8,474.6	8,955.4
4	7,454.6	7,590.5	7,571.8	7,763.3
6	7,944.5	7,492.9	7,040.0	6,898.9
Burma14				
0	4,305.5	4,131.9	4,253.3	3,803.7
2	4,237.7	4,305.2	3,852.2	4,136.7
4	4,044.3	3,454.5	4,018.3	3,830.4

TABLE II. DISTANCE OF TSP TOURS.

n_f	R = 500	1,500	2,500	3,500
Ulysses22				
0	13,174.3	12,592.2	14,206.2	11,694.5
2	13,037.2	13,326.6	13,059.6	11,879.7
4	11,905.6	10,900.8	12,292.0	12,236.0
6	12,006.4	12,156.9	12,237.4	15,064.9
10	9,917.0	10,034.7	11,529.5	10,610.4
12	9,624.2	9,847.6	9,476.0	9,583.9
14	9,222.8	10,712.9	10,528.5	9,222.8
Ulysses16				
0	9,572.2	11,282.0	12,991.0	9,289.8
2	9,761.5	11,993.8	9,994.8	10,393.2
4	7,454.6	8,387.1	8,853.7	10,507.5
6	9,729.9	8,949.0	8,440.0	6,898.9
Burma14				
0	4,305.5	4,237.9	4,705.7	3,886.1
2	6,285.2	5,130.6	6,005.5	5,250.1
4	5,602.5	4,289.9	5,483.9	4,742.6

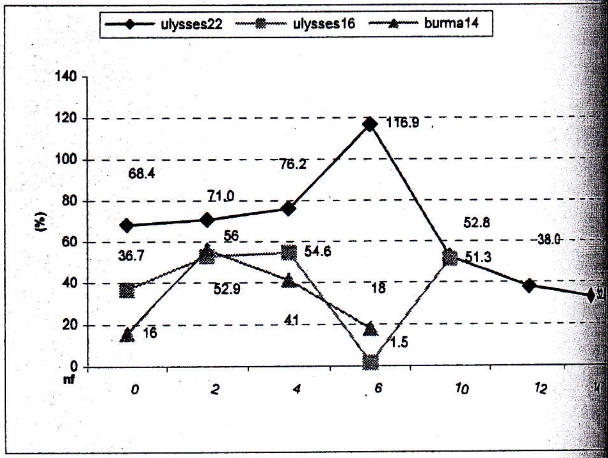

Figure 1. Percentage of deviation between predicted total distance and optimal values when R =3500.

As expected, we see that as the test problems beco[me] more difficult (higher number of cities), the distances of TSP tours deviate from the optimal tours significantly. For Burma14 problem, the best relative deviation is only 16 whereas for the Ulysses22 problem, the best relative deviati[on] is 33%. The benefit of reducing the search space by initia[lly] creating a subtour ($n_f > 0$) is greater for a large test proble[m]. In Fig. 1, we see that the relative deviations of $n_f = 10$ and 14 are smaller than the case with $n_f = 0$. For Ulyssess22 problem, while setting n_f =2 and 4 do not seem help improve the quality of the solutions.

Considering that our heuristic sees only a very sm[all] fraction of the search space, less than 4×10^{-6}%, performance is impressive. For the test problem with cities, our TSP tour gives the distance within 33% to 76% the optimal distance, ignoring the possible outlier at 11 The numerical results seem to suggest that we get a be[tter] solution when we increase the sample size R .

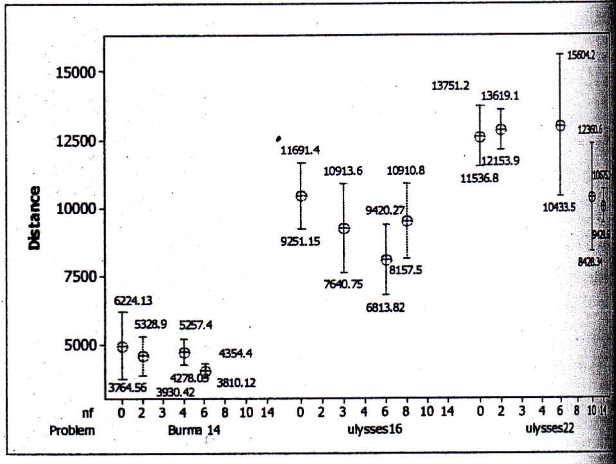

Figure 2. 95% Confidence Intervals of repeating experiment at R =

However those results shown in Table II and Fig. 1 based on one experiment for each case. To see the tren[d] results, we repeat experiments again and the result shown in Fig. 2.

From Fig. 2, we see that if n_f increases, the 95% of confident intervals are shorter. As a result, our predictions are more precise when we increase n_f. Moreover, 95% intervals include the lowest optimal solution. Thus, if we repeat the experiment, we will have more opportunity to get better predicted TSP tours.

V. CONCLUSION

We apply GPR to TSP with an initial numerical experiment. Our idea is to reduce the size of the problem by initially creating a subtour. Then we use a small sample of TSP tours to create a GPR function and minimize it to get the solution with the minimum distance. Then it is transformed into a TSP tour. In our numerical experiment, we show that our heuristic performs quite well. In particular, when the sample size R increases, the predicted results are better. Moreover, when the number of nodes in a subtour n_f increases, the distances of our TSP tours are closer to the true optimal ones. We expect our heuristic to improve if we can find a better method for a tour construction in the last step.

VI. ACKNOWLEDGMENT

Chantapanich would like to thank Dr. Peerayuth Charnsertikul for his suggestion and International Graduate Program (IGP) in Industrial Engineering of Kasetsart University for the financial support and allowing us to use computer laboratory facilities.

VII. REFERENCES

[1] K. Hoffman and M. Padberg. (1985), "Traveling Salesman Problem" ,Encyclopedia of Operations research,Springer-Verlag.

[2] M. Dorigo and L. M. Gambardella (1996). "Ant colonies for the traveling salesman problem." ,Biosystem, Libre de Bruxelles University, pp. 73-81

[3] E. Y. Jeong (1997). "Application of traveling salesman problem (TSP) for decision of optimal production sequence" , Korean Journal of Chemical Engineering Vol. 14, No. 5: pp. 416-421.

[4] L. A. Hall. Approximation algorithms for scheduling. In Dorit S. Hochbaum, Approximation Algorithms for NP-Hard Problems. PWS Publishing Company, 1995.

[5] D. L. Applegate, R. M. Bixby, V. Chvátal, and W. J. Cook, "The Traveling Salesman Problem, "A Computational Study, Princeton University Press, vol. 1, ISBN 978-0-691-12993-8, 2006.

[6] J. Kocijan, R. Murray-Smith, C. E. Rasmussen, and B. Likar, "Predictive control with Gaussian process models, " Proceedings of IEEE Region 8 Eurocon 2003: Computer as a Tool, pp. 352-356, 2003.

[7] S. B. Liu, K. M. Ng, and H. L. Ong (2007), "A new heuristic algorithm for the classical symmetric traveling salesman problem", International Journal of Computational and Mathematical Sciences, 1(4), pp. 234-238.

[8] A. Blum , S. Chawla , D. R. Karger, T. Lane , Adam Meyerson , and M. Minkoff, "Approximation Algorithms for Orienteering and Discounted-Reward TSP" , Society for Industrial and Applied Mathematics, vol. 37, pp. 653-670, 2007.

[9] A. Varga, C. Chira, and a. D. Dumitrescu, "A Multi-agent Approach To Solving Dynamic Traveling Salesman Problem", Proceedings of the 1st International Conference on Bio-Inspired Computational Methods Used for Difficult Problems Solving: Development of Intelligent and Complex Systems. AIP Conference Proceedings, Volume 1117, pp. 189-197, 2009.

[10] M. Hasegawa, "Glassy Dynamics in Local Search by Metropolis Algorithm: Temperature-Cycling Experiments On Traveling Salesman Problem" ,Intelligent Interaction Technologies, University of Tsukuba, Tsukuba, 2006.

[11] C. Walder, K. Williams, and B. scholkopf, "Sparse Multiscale Gaussian Process Regression", In 25th International Conference on Machine Learning. ACM Press, New York, 2008.

[12] M. Ebden. (2008), Gaussian Process for Regression. Available: http://www.robots.ox.ac.uk/~mebden/reports/GPtutorial.pdf. , January 30, 2010.

[13] P. Sollich and C. K. I. Williams. (2005). Using the Equivalent Kernel to Understand Gaussian Process Regression. In Saul, L. K., Weiss, Y., and Bottou, L., editors, Advances in Neural InformationProcessing Systems 17. MIT Press.

[14] E. Meeds and S. Osindero, An alternative infinite mixture of Gaussian process experts. In Y. Weiss, B. Schölkopf, and J. Platt, editors, Advances in Neural Information Processing Systems 18, pp. 883-890. The MIT Press, Cambridge, MA, 2006.

[15] C. E. Rasmussen and Z. Ghahramani, "Infinite mixtures of Gaussian process experts" ,In T. G. Diettrich, S. Becker, and Z. Ghahramani, editors, Advances in Neural Information Processing Systems 14. The MIT Press, 2002.

[16] P. Boyle and M. Frean, "Dependent Gaussian Processes" ,In L. K. Saul, Y. Weiss, and L. Bottou, editors, Advances in Neural Information Processing Systems 17, pp. 217-224. The MIT Press, 2005.

[17] C. K. I. Williams, P. W.Goldberg, and C. K. I. Williams, "Regression with Input-dependent Noise:A Gaussian Proces Treatment", In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, Advances in Neural Information Processing Systems 10. The MIT Press, Cambridge, MA, 1998.

[18] C. E. Rasmussen and C. Williams (2006) "Gaussian Process for Machine Learning",The MIT Press.

[19] J. Chantapanich."Solving the Traveling Salesman Problem with Gaussian Process Regression", Kasetsart University, 2011.

[20] Rice Univeristy Software Distribution Center, Available: http://softlib.rice.edu/pub/tsplib/tsp/., January 30,2010,unpublished.