

Algoritmo Genético com Coordenação Fuzzy para Resolução do Problema de Roteamento de Veículos com Janelas de Tempo

Vitor Marques , Fernando Gomide (Orientador)

Departamento de Engenharia de Computação e Automação Industrial (DCA)

Faculdade de Engenharia Elétrica e de Computação (FEEC)

Universidade Estadual de Campinas (Unicamp)

Caixa Postal 6101, 13083-970 – Campinas, SP, Brasil

{vmarques, gomide}@dca.fee.unicamp.br

Abstract – This paper presents a genetic algorithm coordinated by fuzzy rule models to solve the vehicle routing problem with time windows. The fuzzy rule-based coordinators play distinct roles during the genetic algorithm execution. The aim is to trade-off exploration and exploitation behavior for route and distance minimization. Experimental results using classic benchmark test instances suggest that the fuzzy coordinated genetic approach is competitive against classic genetic algorithm.

Keywords – GFS, VRPTW, Genetic Algorithm

1. Introdução

O problema de roteamento de veículo com janelas de tempo (PRVJT) é um problema combinatorial e tem sua origem nos problemas clássicos do caixeiro viajante (PCV) e no problema do empacotamento (PE), dois respeitados e conhecidos problemas *np-difíceis*. Aplicações para PRVJT são inúmeras, especialmente em transportes e sistemas de logística como serviços de entrega, postagem e distribuição.

Já foi gasto um grande esforço para resolver PRVJT até 100 clientes e enfatizando minimização de distância [7],[16]. Instâncias mais práticas, todavia, necessitam meta-heurísticas para obtenção de solução com tempo de computação razoável.

Este trabalho propõe o uso de um sistema baseado em regras fuzzy (SBRF) para coordenar um algoritmo genético (AG) para resolver PRVJT. O SBRF coordenador é um conjunto de modelos baseados em regras cujo propósito é manter o equilíbrio entre intensificação e diversificação no AG. A idéia principal é tentar evitar paradas precipitadas em ótimos locais; e como é bem sabido, boas soluções para PRVJT está em pequenas partes de todo espaço de busca [20]. Ainda, uma dificuldade adicional para AGs quando resolvem PRVJT é que pequenas alterações na geração de um AG geralmente direciona o algoritmo para uma parte do espaço de busca onde ele não consegue evoluir. A solução adotada para superar tal dificuldade tem sido especializar os operadores do GA [20],[26]. Claramente, a natureza do PRVJT requer um controle cuidadoso das características de intensificação e diversificação e sistema baseados em regras fuzzy [21] proporciona

um mecanismo poderoso para resolver isto.

O uso de SBRF para controle dinâmico de algoritmos genéticos, foi inicialmente proposto em [18], onde taxa de recombinação e tamanho da população eram dinamicamente escolhidos durante execução do AG. [17] usa controlador fuzzy dinâmico para resolver o PRV multi-objetivo com múltiplos depósitos e produtos. Aqui o controle é sobre a variação da taxa de recombinação e mutação, usando fitness médio ao longo das gerações e uma medida de diversidade média da população. Mais especificamente, o controle é realizado em um procedimento de dois passos. O primeiro passo coleta valores do fitness médio, diversidade da população e tamanho médio das rotas dentro da população. O segundo passo define taxa de recombinação, mutação e da busca local. O tamanho da população não está sujeito a controle.

Entre as várias meta-heurísticas existentes, uma de grande sucesso para PRVJT é a busca tabu (BT) [8]. Uma outra importante meta-heurística é sistema de colônia de formigas. [10] apresenta um algoritmo com múltiplas colônias, uma para minimização de veículos e outra para minimização de distância. Uma outra abordagem efetiva é a combinação de meta-heurísticas. Por exemplo, [5] propõe um busca reativa em vizinhança variável, uma variação da BVV com quatro fases: inicialização, redução de rotas, minimização da distância total usando quatro novas buscas locais e uma modificação da função objetivo para escapar do máximo local.

Muitas das mais bem sucedidas meta-heurísticas para instâncias grandes do PRVJT usam

alguma forma de computação paralela. Em [14], busca com estratégias cooperativas e paralela são criadas usando meta-heurísticas em duas fases; a primeira fase tenta minimizar o número de veículos usando uma meta-heurística evolucionária, a segunda fase tenta minimizar a distância total percorrida com uma busca tabu. Como uma alternativa, [4] apresenta um método de múltiplas buscas cooperativas e paralelas baseado na estratégia de armazém (“warehouse”), na qual várias “threads” de busca cooperam trocando informações assincronamente a respeito das melhores soluções identificadas. A pesquisa [9] exhibe mais detalhes e realiza comparações de resultados.

Diferentemente das abordagens correntes visitadas na literatura, este trabalho propõe modelos baseados em regras fuzzy para coordenar algoritmos genéticos na resolução do PRVJT. O propósito dos coordenadores com SBRF é dinamicamente escolher taxa de reprodução e mutação, estruturas de vizinhança local e o comportamento de busca do AG (intensificação, diversificação) usando uma taxa de melhoria, uma medida para diversidade, proporção média para incumbente e proporção média de rotas pequenas como entradas.

Após esta introdução o artigo segue da seguinte maneira. Seção 2. apresenta a formulação do PRVJT e a seção 3. detalha o AG coordenado com fuzzy. Resultados experimentais são resumidos na seção 4.. Seção 5. conclui o artigo e sugere itens para sequência de trabalhos.

2. Formulação do PRVJT

Este trabalho trata do PRVJT com um depósito. Mais especificamente, existe um conjunto de clientes que possuem uma demanda de transporte específica e uma janela de tempo determinando quando a demanda deve ser atendida. Veículos com capacidades idênticas realizam o atendimento do serviço. O objetivo é achar o conjunto de rotas com custo mais baixo começando e terminando no depósito dentro do horário de trabalho. Factível significa que os veículos devem respeitar suas capacidades e atender a demanda do cliente dentro da janela de tempo especificada por este. Cada cliente é atendido uma vez. O serviço pode ser definido antes da janela de tempo (neste caso o veículo espera), mas não pode ultrapassar a janela. Essa é a abordagem rígida (“hard”) para o PRVJT.

O PRVJT pode ser associado a um grafo $G(V, A)$ onde $V = C \cup \{v_0, v_{n+1}\}$ e $C = \{v_1, \dots, v_n\}$ representam os n clientes e v_0, v_{n+1} o depósito. O conjunto $A = \{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$ define arestas entre nós clientes, e v_0 e v_{n+1} representam o depósito no início e fim da rota, respectivamente. Conseqüentemente, não existe aresta começando em v_{n+1} nem uma aresta terminando em v_0 . Cada aresta (v_i, v_j) têm um custo c_{ij} em um tempo de viagem associados t_{ij} . Um tempo de serviço s_i é associado ao cliente i que tem demanda d_i , $i \in C$. O depósito tem um conjunto K veículos com capacidade q para servir os clientes. Uma janela de tempo $[e_i, l_i]$ de ve ser obedecida quando inicia-se o serviço do cliente i . As variáveis de decisão são X^{kij} que assume o valor 1 se o veículo k atravessa aresta $(i, j) \forall k \in K, \forall (i, j) \in A$.

Deixe S^k_i ser o tempo que o veículo k começa atender o cliente i , $\forall k \in K, \forall i \in C$. O problema PRVJT pode ser assim descrito:

$$\min \sum_{k \in K} \sum_{(i,j) \in A} c_{ij} X^{kij} \quad (1)$$

sujeito a:

$$\sum_{k \in K} \sum_{j \in C \cup \{v_{n+1}\}, j \neq i} X^{kij} = 1, \quad \forall i \in C \quad (2)$$

$$\sum_{i \in C} \sum_{j \in C \cup \{v_{n+1}\}, j \neq i} d_i X^{kij} \leq q, \quad \forall k \in K \quad (3)$$

$$\sum_{j \in C \cup \{v_{n+1}\}} X^{k0j} = 1, \quad \forall k \in K \quad (4)$$

$$\sum_{i \in C} X^{kin+1} = 1, \quad \forall k \in K \quad (5)$$

$$\sum_{i \in C \cup \{v_0\}} X^{kih} - \sum_{j \in C \cup \{v_{n+1}\}} X^{hij} = 0, \quad \forall h \in C, \forall k \in K \quad (6)$$

$$X^{kij} (S^k_i + s_i + t_{ij} - S^k_j) \leq 0, \quad \forall (i, j) \in A, \forall k \in K \quad (7)$$

$$e_i \leq S^k_i \leq l_i, \quad \forall i \in C \cup \{v_{n+1}\}, \forall k \in K \quad (8)$$

$$X^{kij} \in \{0, 1\}, \quad \forall (i, j) \in A, \forall k \in K \quad (9)$$

Portanto, o objetivo é minimizar o custo geral de transporte considerando as seguintes restrições: (2) todos clientes devem ser atendidos por um único veículo, (3) toda demanda atendida por um veículo não pode ultrapassar sua capacidade de carga, (4) toda rota deve iniciar no nó v_0 , (5) todo

rata deve terminar no nó v_{n+1} , (6) um veículo deve atender e sair do cliente, (7) se um cliente j vêm depois do cliente i em uma rota, o tempo de serviço i e o tempo total de viagem (i, j) são considerados, (8) um veículo deve atender o cliente dentro de sua janela de tempo.

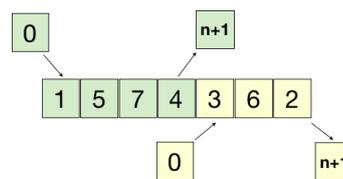


Figura 2. estrutura do cromossomo

3. Coordenação Fuzzy do AG

A coordenação fuzzy do algoritmo genético (CFAG) tem três grandes componentes: algoritmo genético, algoritmo de busca local e coordenação com base de regras fuzzy, respectivamente. Fig. 1 ilustra a abordagem CFAG. Segue os detalhes destes três componentes.

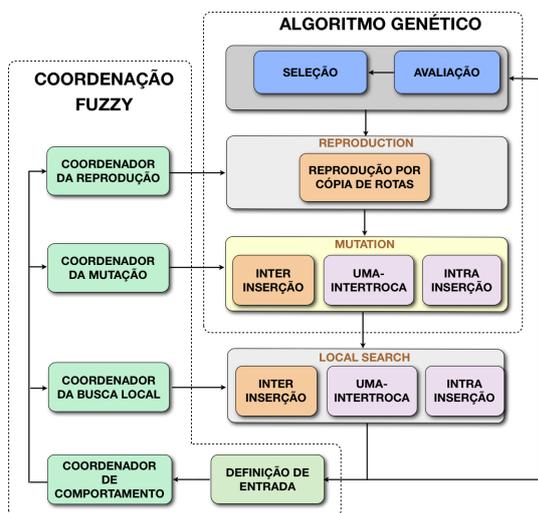


Figura 1. CFAG

3.1. Algoritmo Genético

Um cromossomo representa a ordem dos clientes a serem visitados. Os clientes são colocados em sequência. O depósito é omitido, mas é considerado implicitamente. Fig. 2 mostra a estrutura do cromossomo. Note que o cromossomo é composto por subsequências, cada uma correspondendo à uma rota factível. A população inicial é gerada da seguinte maneira. O primeiro indivíduo é gerado utilizando “*push forward insertion heuristics*”. Os indivíduos restantes são gerados das mesmas estruturas de vizinhança usadas na mutação. Diversidade populacional é garantida usando *medida de Jaccard* para evitar indivíduos similares.

3.1.1. Operadores do AG

A *reprodução* é baseada no operador de recombinação por cópia de rotas. Este operador copia rota dos dois parentes para formar os filhos. Se um cliente não está em alguma rota, ele é inserido em alguma rota que permaneça factível. Se isto não for possível, novas rotas são formadas com estes clientes. O procedimento de *mutação* usa três estruturas de vizinhança: *inter-inserção*, *intra inserção* e *uma-intertroca*, respectivamente. Vizinhanças são escolhidas usando uma distribuição uniforme. A *seleção* é executada usando uma variação da roleta russa, chamado de procedimento de amostragem universal estocástica [3], que busca reduzir a chance de repetir indivíduos com alto fitness, na nova população.

3.1.2. Fitness

A função de fitness considera tanto distância total (dt) quanto número de veículos (nv) na solução: $fitness(dt, nv) = \left(w_d \frac{1}{dt}\right) + \left(w_v \frac{1}{nv}\right)$, onde w_d e w_v são pesos com valores reais. O propósito do AG é maximizar o fitness, que acontece para os menores valores de dt e nv .

3.2. Busca Local

O procedimento de busca local usa as mesmas estruturas de vizinhança da mutação. Ela sempre tenta melhorar o incumbente. A taxa da busca local pode ficar mais alta ou mais baixa dependendo do coordenador fuzzy da busca local. Ainda, o coordenador pode enfatizar redução de rotas (aumentando quantidade de movimentos de inter-inserção), ou redução da distância (aumentando quantidade de movimentos intra-inserção e uma-intertroca) dependendo da fase da busca e da proporção média das rotas pequenas.

3.3. Coordenação por Base de Regras Fuzzy

O propósito dos coordenadores por base de regras fuzzy é ajudar o AG a diversificar a busca sempre que ele fica parado em ótimos locais, e intensificar a busca sempre que atingir uma nova região do espaço de busca. Eles decidem também quando minimização de veículo ou distância deve ser reforçada. Os coordenadores fuzzy são modelos de base de regras linguísticas. Por simplicidade, este trabalho usa o procedimento de inferência clássico max-min com defuzzificação usando centro de gravidade. As variáveis de entrada e saída e a base de regras de cada coordenador são explanadas na sequência.

Entradas: TG, taxa de ganho, melhoria. Mede a capacidade atual da busca local de melhorar uma solução; é a fração *número de todos movimentos factíveis que melhoram a solução* e o *número de todos movimentos factíveis*. DIV, diversidade da população. Mede quão diferente são os indivíduos da população. É a fração *números de arestas usadas da população* e *número total de arestas factíveis*. PMI, proporção média para incumbente. Mede quão distante estão os indivíduos da população do incumbente atual em termos da função de fitness. Proporção da população para incumbente (*PI*) é a fração entre *fitness de um indivíduo* e *fitness do incumbente*. Logo PMI é a média do *PI* da população. PRP, proporção média de rotas pequenas. Mede a proporção de rotas pequenas no conjunto de todas as rotas. Uma rota pequena é definida da seguinte maneira. Deixe *mcr* ser a média de clientes por rota, *cv* a capacidade do veículo, *md* a média de demanda entre todos clientes, *tfd* tempo de fechamento do depósito, *tms* o tempo médio de serviço dos clientes, e *tmv* o tempo médio de viagem entre todos clientes. Define-se *mcr* como

$$mcr = \min \left(\left\lceil \frac{cv}{md} \right\rceil, \left\lceil \frac{tfd}{(tmv + tms)} \right\rceil \right),$$

onde $\lceil x \rceil$ significa o maior inteiro mais perto de x . Uma rota pequena é uma rota com menos clientes que um percentual de *mcr*. Este artigo adota 20% em todos experimentos.

Saídas: FASE, indica se o algoritmo deve iniciar intensificação, diversificação ou uma fase intermediária de busca. TR, taxa de reprodução. TM, taxa de mutação. TB, taxa de busca. Proporção da população para a qual a busca local é aplicada. TMV, taxa de minimização de veículos, percentual da taxa de busca que reforça a minimização de veículos du-

rante a busca local. TMD, taxa de minimização de distância, percentual da taxa de busca que reforça a minimização de distância durante a busca local.

Todas variáveis linguísticas, exceto FASE, usa *P* para pequeno, *M* para médio, *G* para grande, *B* para baixo *A* para alto. Para FASE, *I* significa intensificação, *D* diversificação, e *IN* intermediário.

3.3.1. Coordenadores Fuzzy

O **coordenador do comportamento da busca** recebe a taxa de ganho e diversidade da população durante a execução do AG e processa a base de regras para escolher entre intensificação, diversificação ou comportamento intermediário de busca. Um exemplo de regra é: SE TG é *M* e DIV é *P* ENTÃO FASE é *D*. A base de regras especifica, se a busca local não está melhorando a solução satisfatoriamente, o algoritmo deve enfatizar a diversificação, entrando na fase de diversificação. Quando altas taxas de ganhos são observadas, o algoritmo entra em uma fase de intensificação. Com taxas de ganhos intermediárias, se a diversidade é baixa, a diversificação é incentivada, caso contrário a intensificação é favorecida. O **coordenador da reprodução** recebe a informação da fase do coordenador de comportamento de busca e a proporção média para incumbente para definir a taxa de reprodução. A base de regras diz, por exemplo, que a taxa de reprodução deve ser alta quando algoritmo está em fase de intensificação e baixa durante diversificação. A proporção média para incumbente (PMI) evita empates quando a fase não é de intensificação nem diversificação. O **coordenador da mutação** recebe a informação da fase do coordenador de comportamento de busca e a proporção média para incumbente para definir a taxa de mutação. O **coordenador da busca local** usa a informação de fase do coordenador de comportamento da busca e a proporção média de rotas pequenas para definir a taxa da busca, a taxa de minimização de veículos e taxa de minimização de distância. A base de regras determina, por exemplo, que quanto mais alta a diversificação mais baixa é a taxa de busca, quanto mais alta é a intensificação maior é a taxa de busca.

4. Resultados Experimentais

O algoritmo coordenado por regras fuzzy sugerido neste trabalho foi testado com as instâncias de Solo-

mon [23], um benchmark clássico para PRVJT. Ele usa três tipos de geração de clientes e suas localizações: randômica (instâncias R), de agrupamento (instâncias C), e combinação dos dois (instâncias RC). O resultados reportados são os melhores de 10 rodadas de execução, com 200 gerações cada, usando um Pentium Intel Core 2 duo, 2.4GHz with 2GB RAM. O algoritmo foi implementado usando Java e JFuzzyLogic [15]. Os resultados alcançados foram comparados contra o AG clássico com parâmetros fixados em TR=0.9, TM=0.1, TB=0.25, TMV=0.5, TMD = 0.5 para todos experimentos. O AG clássico levou em média 10 minutos em média para executar todo conjunto de instâncias. O algoritmo coordenado por base de regras fuzzy levou em média 12 minutos.

Tabela 1 resume os melhores resultados para PRVJT encontrados na literatura. O AG clássico com busca local e parâmetros fixados melhorou levemente a qualidade da solução, quando comparado com o melhor resultado [14], aproximadamente 1% para tanto, número de veículos (NV) e total de distância viajada (D). O AG coordenado por fuzzy se mostrou mais efetivo, apresentando ganho de 2.7% e 6.8% para número de veículos e distância total, respectivamente.

5. Conclusão

Este artigo introduziu um algoritmo genético com coordenação fuzzy para resolver o PRVJT. O propósito da coordenação é balancear diversificação e intensificação, e reforçar minimização de veículos ou minimização de distância durante a execução do algoritmo genético. Resultados experimentais revelam que AG com coordenação fuzzy é competitivo comparado aos melhores resultados da literatura para PRVJT por melhorar a qualidade da solução. Estamos trabalhando também com objetivo de construir um framework para automatização de parâmetros para meta-heurísticas.

Referências

- [1] R. Ah King, B. Radha, and H. Rughooputh. A fuzzy logic controlled genetic algorithm for optimal electrical distribution network reconfiguration. In *Networking, Sensing and Control, 2004 IEEE International Conference on*, volume 1, pages 577–582, March 2004.

Tabela 1. Resultados para Instâncias de Solomon

Autor	Resultados por grupo de instância		
	Instância	NV	D
AG	R1	11.25	1147.28
	C1	10.00	939.06
	RC1	11.75	1304.72
	R2	2.91	915.90
	C2	3.125	724.69
	RC2	3.125	1011.85
	Accum.	401.00	56624.15
AG + Fuzzy	R1	11.00	1100.88
	C1	10.00	874.85
	RC1	11.37	1264.62
	R2	2.91	843.91
	C2	3.00	690.30
	RC2	3.125	947.96
	Accum.	394.00	53590.52
[4]	R1	12.08	1209.19
	C1	10.00	828.38
	RC1	11.50	1386.38
	R2	2.73	960.95
	C2	3.00	589.86
	RC2	3.25	1133.30
	Accum.	407.00	57412.37
[13]	R1	11.91	1212.73
	C1	10.00	828.38
	RC1	11.50	1386.44
	R2	2.73	955.03
	C2	3.00	589.38
	RC2	3.25	1108.52
	Accum.	405.00	57192.00
[10]	R1	12.00	1217.73
	C1	10.00	828.38
	RC1	11.63	1382.42
	R2	2.73	967.75
	C2	3.00	589.86
	RC2	3.25	1129.19
	Accum.	407.00	57525.00

- [2] P. Badeau, F. Guertin, M. Gendreau, J. Potvin, and E. Taillard. A parallel tabu search heuristic for the vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies*, 5(2):109 – 122, 1997.
- [3] J. Baker. Reducing bias and inefficiency in the selection algorithm. In *Proceeding of the Second International Conference on Genetic Algorithms and their application*, pages 14–21. L. Erlbaum Associates Inc., 1987.
- [4] A. Bouthillier and T. Crainic. A cooperative parallel meta-heuristic for the vehicle routing problem with time windows. 32(7):1685–1708, 2005.

- [5] Olli Bräysy. A reactive variable neighborhood search for the vehicle-routing problem with time windows. *Inform. J. on Computing*, 15(4):347–368, 2003.
- [6] Olli Bräysy et al. A multi-start local search algorithm for the vehicle routing problem with time windows. *European Journal of Operational Research*, 159(3):586 – 605, 2004.
- [7] Alain Chabrier. Vehicle routing problem with elementary shortest path based column generation. *Computers Operations Research*, 33(10):2972 – 2990, 2006.
- [8] J. Cordeau and Gilbert Laporte. A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 52:928–936, 2001.
- [9] T. Crainic and M. Gendreau. Cooperative parallel tabu search for capacitated network design. *Journal of Heuristics*, 8(6):601–627, 2002.
- [10] L. Gambardella, E. Taillard, and G. Agazzi. Macs-vrptw: A multiple ant colony system for vehicle routing problems with time windows. Technical report, Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale, 1999.
- [11] H. Gehring and J. Homberger. Two evolutionary metaheuristics for the vehicle routing problem with time windows. *Infor.*, 37:297–318, 1999.
- [12] Michel Gendreau, Alain Hertz, and Gilbert Laporte. A tabu search heuristic for the vehicle routing problem. *Manage. Sci.*, 40(10):1276–1290, 1994.
- [13] J. Homberger and H. Gehring. A parallel two-phase metaheuristic for routing problems with time windows. *Asia-Pacific Journal of Operational Research*, 13(1):35–47, 2001.
- [14] J Homberger and H. Gehring. A two-phase hybrid metaheuristic for the vehicle routing problem with time windows. *European Journal of Operational Research*, 162(1):220–238, 2005.
- [15] Open source fuzzy logic library and fcl language implementation.
- [16] B. Kallehauge, J. Larsen, and Oli Madsen. Lagrangean duality applied on vehicle routing with time windows - experimental results. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, 2001.
- [17] H Lau, T. Chan, W. Tsui, F. Chan, G. Ho, and K. Choy. A fuzzy guided multi-objective evolutionary algorithm model for solving transportation problem. *Expert Systems with Applications*, 36(4):8255 – 8268, 2009.
- [18] M. Lee and Hideyuki Takagi. Dynamic control of genetic algorithms using fuzzy logic techniques. In *Proc. of the Fifth International Conference on Genetic Algorithms*, pages 76–83, 1993.
- [19] Oli Madsen, Niklas Kohl, J. Desrosiers, M. Solomon, and F. Soumis. 2-path cuts for the vehicle routing problem with time windows. *Transportation Science*, 33:101–116, 1999.
- [20] B. Ombuki, B. Ross, and F. Hanshar. Multi-objective genetic algorithms for vehicle routing problem with time windows. *Applied Intelligence*, 24:2006, 2006.
- [21] W. Pedrycz and F. Gomide. *Fuzzy Systems Engineering: Toward Human-Centric Computing*. Wiley Interscience/IEEE, Roboken-NJ, USA, first edition, 2007.
- [22] J. Potvin and S. Bengio. The vehicle routing problem with time windows - part ii: Genetic search. *Inform. J. on Computing*, 8(2):165–172, 1996.
- [23] M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.*, 35(2):254–265, 1987.
- [24] E. Taillard, P. Badeau, M. Gendreau, F. Guertin, and J. Potvin. A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*, 31(2):170–186, 1997.
- [25] K. Tan, L. Lee, and K. Ou. Artificial intelligence heuristics in solving vehicle routing problems with time window constraints. *Engineering Applications of Artificial Intelligence*, 14(6):825 – 837, 2001.
- [26] S. Thangiah. Vehicle routing with time windows using genetic algorithms. In *Application Handbook of Genetic Algorithms: New Frontiers, Volume II*, pages 253–277. Lance Chambers, CRC Press, 1995.