



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA

DO RIO GRANDE DO NORTE

DIRETORIA ACADÊMICA

CURSO TÉCNICO INTEGRADO EM INFORMÁTICA

TRADUÇÃO – ARTIGO

**REDESIGN OF THE SUPPLY OF MOBILE MECHANICS BASED ON A NOVEL GENETIC
OPTIMIZATION ALGORITHM USING GOOGLE MAPS API**

ANDRÁS KIRÁLY, JÁNOS ABONYI

DOCENTE (S): ROMERITO CAMPOS DE ANDRADE

DISCENTE (S):

ANA SANTANA DE SOUZA FÉLIX

BRENDA DANTAS BEZERRA

MIRELE DA SILVA COSTA

VERÔNICA MEDEIROS PEREIRA

CAICÓ/RN

2017

Redesign do fornecimento de mecânica móvel com base em um novo algoritmo de otimização genética usando a API do Google Maps

András Király e János Abonyi

University of Pannonia, Department of Process Engineering, P.O. Box 158. Veszprém H-8200, Hungary

Resumo

Se um mecânico móvel tiver que viajar por material, o tempo produtivo é perdido. Este artigo apresenta um novo método para reduzir atividades relacionadas ao manuseio de materiais com a extensão de locais de atendimento. O design do sistema de suprimento pode ser considerado um problema complexo de otimização combinatória, onde o objetivo é encontrar um plano de rota com custo mínimo de rota, que presta serviços a todas as demandas dos armazéns centrais, ao mesmo tempo em que satisfaz a capacidade e outras restrições. Apresentamos uma técnica multi-cromossomo para resolver o Problema do Múltiplo Caixeiro Viajante (mTSP). Os novos operadores, com base em uma representação específica do problema, mostraram-se mais eficazes em termos de flexibilidade, complexidade e transparência e também em eficiência do que os métodos anteriores. O algoritmo de otimização proposto foi implementado no MATLAB e integrado ao Google Maps para fornecer uma estrutura completa para cálculo de distância, definição das rotas iniciais e visualização. Esta estrutura integrada foi aplicada com sucesso na solução de um verdadeiro problema logístico, no fornecimento de mecânica móvel em um dos maiores fornecedores de energia da Hungria.

1- Introdução

O objetivo da logística é obter os materiais certos no lugar certo no momento certo, ao mesmo tempo em que otimiza uma determinada medida de desempenho (p. Minimizando o custo operacional total) e satisfazendo um determinado conjunto de restrições (por exemplo, restrições de tempo e capacidade) (Christopher, 2005). O gerenciamento da cadeia de suprimentos inclui o planejamento e gerenciamento de todas as atividades envolvidas em abastecimento, aquisição, conversão e gestão da logística, bem como componentes cruciais da coordenação e colaboração. Trata-se de vários problemas como: distribuição, configuração da rede, trocas em atividades logísticas, estratégia de gerenciamento de estoque ou distribuição

(Bowersox et al., 2002). Na maioria dos sistemas de distribuição, os bens são transportados de várias origens para vários destinos. Por exemplo, muitas cadeias de varejo gerenciam sistemas de distribuição em que os bens são transportados de vários fornecedores para várias lojas de varejo. O desafio é encontrar o plano de transporte ideal, geralmente o mais barato, de acordo com algumas funções objetivas e considerando restrições adicionais (Beamon, 1998). Este é um problema complexo de otimização combinatória, comumente uma tarefa NP-hard. Na logística, vários tipos de problemas poderiam surgir; Um dos mais notáveis é o conjunto de problemas de planejamento de rotas.

A principal motivação deste trabalho é derivada de um problema industrial real, que surgiu na E.ON Hungária Zrt., o principal fornecedor de energia da Hungria. E.ON Network Services Kft. presta serviços principalmente às empresas de fornecimento de energia elétrica e gás do grupo E.ON Hungária. Este serviço inclui toda a gama de atividades de gerenciamento de operações, como garantir o fornecimento de energia ininterrupta, manutenção regular de objetos de rede e a eliminação de interrupções associadas a avarias. Além disso, a empresa instala medidores de consumo, enquanto também renova sites de consumo e elimina avarias informadas pelos clientes.

A empresa possui uma topologia estelar, o que significa que um caminhão transporta os materiais necessários do Depósito Central para os Armazéns, enquanto a mecânica nas Bases transporta os materiais do Armazém para as Bases. Esta topologia produz altos custos operacionais, portanto, a empresa queria substituir este sistema de abastecimento por um caminhão, que pode servir o Armazém e todas as Bases com os materiais necessários. Esse problema pode ser modelado como um problema de vendedores ambulantes múltiplos (mTSP) com janelas de tempo e restrições adicionais.

No mTSP, um conjunto de n nós (locais ou cidades) são fornecidos e m vendedores estão localizados em um único nó de depósito. Os nós restantes ou as cidades que devem ser visitadas são os nós intermediários. O objetivo é encontrar passeios para todos os vendedores, que todos começam e terminam sua rota no depósito central, de modo que cada cidade intermediária seja visitada exatamente uma vez e o custo total de viagem é minimizado. O mTSP é uma generalização do Problema de vendedores ambulantes (TSP), onde os vendedores de $m \geq 1$ estão preocupados. A métrica de custo pode ser definida em termos de distância, tempo, etc. As possíveis variações do problema podem ser encontradas em Bektas (2006) e Gutin e Punnen (2002). No nosso caso, são dadas restrições adicionais, p. Para o



comprimento máximo do percurso por vendedor e janelas de horário que representam os tempos de carregamento dos caminhões.

2- Formulação do problema

No nosso caso, as condições de tráfego devem ser consideradas e os tempos de carregamento também são apresentados em cada local. Temos restrições adicionais, como o tempo / distância de viagem máximos de cada vendedor; janela de tempo em cada local, e a empresa tem que investir em cada caminhão novo. Portanto, o problema pode ser definido como um Problema do Caixeiro Viajante Múltiplo Assimétrico com Janela de Tempo (mTSPTW) e restrições especiais adicionais, onde o número de vendedores é uma variável limitada superior.

Geralmente, o mTSP é especificado como uma formulação de programação de números inteiros. Antes de apresentar o modelo do mTSP modificado mencionado acima, serão dadas algumas definições técnicas. O mTSP é definido em um gráfico $G = (V, A)$, onde V é o conjunto de n nós (vértices) e A é o conjunto de arcos (bordas). Seja $C = (c_{ij})$ uma matriz de custo (distância ou duração) associada a A . A matriz C é simétrica quando $c_{ij} = c_{ji}$; 8 ÷i; JPAA e assimétrico

O problema que é analisado neste artigo é mais complexo do que o problema tradicional do mTSP. É um chamado mTSPTW (Bektas, 2006) com restrições adicionais, que podem ser formuladas da seguinte forma. Vamos definir a seguinte variável binária:

OBS: Variável presente no artigo original

Vamos definir M como o número máximo de vendedores que podem ser incluídos na solução, ou seja, M é o limite superior de m , o número real de vendedor usado. Seja S o comprimento máximo de qualquer turno na solução. Além disso, vamos definir o custo (distância ou duração) associada a A como $C_t = (c_{t,ij})$, onde $c_{t,ij} = c_{ij} + p_{t,j}$, e c_{ij} é o custo ordinário (por exemplo, o salário do motorista, que é proporcional à distância, uma vez que são pagos por hora) do arci j e $c_{t,j}$ é o custo da janela de tempo. A janela de tempo significa que todo vendedor deve aguardar em cada local, o que pode ser, por exemplo, a duração do carregamento das mercadorias. Obviamente, C_t não pode ser uma matriz simétrica, já que em uma aplicação da vida real $c_{ij} \neq c_{ji}$; 8 ÷i; JPAA, por causa disso pode existir, por exemplo, estradas unidirecionais. Assim, o problema de otimização pode ser dado da seguinte maneira:

OBS: Fórmula presente no artigo original

Se usarmos a variável binária recém-introduzida, a função de objetivo baseada em atribuição usual é alterada na Eq. (1), onde o custo do envolvimento de um vendedor aparece também (cm). (2) - (5) são as restrições de atribuição usuais, usando a variável binária x_{ijk} , e (7) garante que o comprimento do percurso de cada vendedor esteja sob o limite especificado, S.

Na maioria das aplicações reais, o custo de uma entrega deve incluir vários fatores, portanto, a função de custo em (1) torna-se mais complicada. Então, podemos expressar o custo de um transporte da seguinte maneira:

OBS: Fórmula presente no artigo original

onde w_s são pesos. Estes fatores podem ser, por exemplo: o salário do motorista, o consumo do caminhão ou o pedágio em uma rodovia. Todos os componentes da função objetivo devem ser transformados em custos reais. Isso é feito por pesos, representando, por exemplo, o custo do combustível ou o salário do motorista por km. Esta abordagem pode ser associada a um modelo multi-objetivo, como pode ser visto em várias obras da literatura, como em Samanlioglu e Kurz (2008) e em Jozefowicz et al. (2008). No entanto, a agregação desses fatores de custo é bastante evidente, porque cada parte da função de custo denotado pode ser expressa em moeda, exemplos: em USD ou em HUF. Assim, a tarefa principal continua sendo um problema de otimização de objetivo único, que pode ser resolvido com a ajuda de uma abordagem inovadora, que será proposta nas próximas seções.

Se quisermos adicionar uma penalidade para o vendedor que atinge o comprimento máximo da turnê, o formalismo acima pode mudar ligeiramente.

OBS: Fórmula presente no artigo original

Assim, a Eq. (1) é alterada da seguinte maneira:

OBS: Fórmula presente no artigo original

Em (10), a penalidade é proporcional ao comprimento do percurso de um vendedor acima do limite superior S, enquanto o grau da punição é determinado pela constante λ , cujo valor depende muito do alcance de c_{ij} . Note que outro tipo de penalidade pode ser um ponto de corte da rota de um vendedor que chega ao limite superior.

3- Revisão da literatura

Nas últimas décadas, o TSP recebeu grande interesse e várias abordagens foram propostas para resolver o problema (Kara e Bektas, 2003, 2006). Alguns desses métodos são algoritmos exatos, enquanto outros são algoritmos quase ótimos ou aproximados. Os algoritmos exatos normalmente usam abordagens de programação linear inteira com restrições adicionais. Recentemente, uma estrutura de software de código aberto para benchmarking e teste do TSP foi disponibilizada, o que fornece uma maneira conveniente de implementar solucionadores TSP em Java (Weise et al., 2014).

A pesquisa do mTSP é muito mais estreita. Bektas (2006) dá uma revisão abrangente dos métodos conhecidos. Pesquisas recentes podem ser encontradas em Nallusamy et al. (2009), onde os autores usaram a combinação de diferentes abordagens para resolver o problema mTSP. Usando K-Means Clustering, o problema é reduzido em TSP m, onde m é o número de vendedores. As soluções TSP iniciais são formadas pelo Algoritmo de encolhimento de encolhimento (Shrink Wrap Algorithm) e, finalmente, aplicam Tabu Search e Simulated Annealing para resolver o mTSP. O número de vendedores é fixo, e apenas dados sintéticos foram usados para testar o algoritmo com 60 cidades e 8 vendedores. A comparação com outros métodos não faz parte do artigo, e basicamente os TSP múltiplos são otimizados e as soluções são combinadas para formar a solução final. A otimização da colônia de formigas para o mTSP é usada muito recentemente como em Ghafurian e Javadian (2011), onde os autores fornecem uma solução para o mTSP multi-depósito. Após a revisão das soluções AC para o mMTSP, eles apresentam o destino fixo mMTSP considerando os limites de viagem para cada vendedor. Eles usam listas de tabu para manter destinos visitados e aplicar restrições para garantir soluções válidas. Seus testes são baseados apenas em dados sintéticos, e o número de depósitos e caminhões são corrigidos. Em Shi e Weise (2013), um problema bastante semelhante é apresentado ao que discutimos na seção anterior, no entanto, os autores usam terminologia diferente e a função objetivo é ligeiramente diferente. Nesse artigo, um algoritmo ACO inicializado é apresentado para resolver o problema do roteamento do veículo com o Time Windows (VRPTW). O algoritmo otimiza o problema de uma forma inovadora, onde o número de veículos necessários é minimizado como principal objetivo, bem como a distância geral de viagem como objetivo de segunda prioridade. Os objetivos são otimizados simultaneamente, mas não em uma moda Pareto. Uma boa revisão sobre Ant Colony Optimization para problemas de agendamento é apresentada em Neto e Filho (2013).

Recentemente, algoritmos genéticos são implementados com sucesso para resolver o TSP e suas variantes. Os GAs são algoritmos de busca aleatórios diretos, baseados no modelo evolutivo (Goldberg, 1989), relacionados com a teoria evolutiva de Darwin. Potvin apresenta um levantamento das abordagens GA para o TSP geral (Potvin, 1996). Para resolver o mTSP, devido à sua complexidade combinatória, é necessário aplicar alguma heurística na solução, especialmente em aplicações de tamanho real.

Na literatura, existem vários exemplos de que uma boa representação específica de problemas pode melhorar drasticamente a eficiência de algoritmos genéticos. Um design individual específico do problema pode reduzir o espaço de busca e, usando operadores especiais para lidar com essa representação, o problema pode ser modelado de forma mais eficaz. Essas propriedades tornam o algoritmo genético específico do problema mais apropriado para a tarefa determinada e torna-se mais facilmente interpretável. A maior parte do trabalho na resolução de mTSPs usando AGs tem focado no Problema de Programação de Veículo (VSP). O VSP geralmente inclui restrições adicionais, como a capacidade de um veículo (também determina o número de cidades que cada veículo pode visitar), ou janelas de tempo para a duração das cargas. Em Malmberg (1996) é apresentada a minimização de períodos unitários de espera (UPW). O autor demonstra o agendamento de um único caminhão para entrega de correios entre destinos durante um período de operação fixa. O veículo tem capacidade ilimitada e cada destino pode ser alcançado a partir de qualquer local. É uma solução em duas fases, onde rotas aleatórias são geradas. Em primeiro lugar, trata os subprocessos como TSPs e, na segunda fase, estas soluções relativamente boas são combinadas pelo GA. Uma GA híbrida é submetida em Park (2001), onde cada etapa do GA gera uma solução viável, que é garantida por um processo de reparo. O Park aplicou um mecanismo de 4 passos, onde a população é dividida em subpopulações e os melhores indivíduos são selecionados em cada etapa do GA para formar as novas subpopulações. A função de fitness é a combinação das três funções objetivas: o tempo de viagem total normalizado, o atraso total ponderado normalizado e o tamanho normalizado da frota. Outra aplicação recente pode ser encontrada em Tang et al. (2000), onde os GAs são desenvolvidos para o agendamento rápido. Não há restrições sobre os comprimentos de rotas dos vendedores, e introduz muitos nós falsos e algumas variáveis binárias adicionais, portanto, eles podem converter o mTSP em um único TSP e aplicar um GA modificado para resolver o problema.

Na próxima seção, propomos um algoritmo genético baseado em uma nova representação multi-cromossômica para mTSP, que é semelhante à representação usada para

programação de veículos em Tavares et al. (2003). No entanto, os operadores de cruzamento sugeridos por Tavares et al. podem produzir crianças inviáveis, portanto, etapas adicionais de melhoria devem ser realizadas. Em contraste, nossos operadores sempre geram soluções viáveis, ou seja, a correção adicional não é necessária. Portanto, no seguinte, apresentaremos uma breve descrição da representação usada, e a descrição dos novos operadores cruzados receberá o foco principal.

A principal motivação para o uso de representações genéticas multi-cromossômicas origina-se do reconhecimento de que, embora os vendedores no mTSP sejam separados um do outro "fisicamente", quase todas as soluções anteriores do mTSP com GA usaram um único cromossomo para representar uma solução inteira. Essas abordagens são a técnica cromossômica (Zhang et al., 1999), a técnica do cromossomo dois (Malmberg, 1996; Park, 2001) e o cromossomo único mais eficaz até agora, a chamada técnica cromossômica de duas partes (Carter e Ragsdale, 2006), o que reduz o tamanho do espaço de busca pela eliminação de soluções redundantes. Um recente agrupamento de novidades GA usa uma representação muito próxima à abordagem multi-cromossômica (Singh e Baghel, 2009) eo algoritmo proposto minimiza a redundância durante a reprodução. O GA apresentado por Singh e Baghel aplica o cruzamento ou o operador de mutação nos indivíduos em cada passo. O operador de cruzamento é altamente complexo, onde os novos indivíduos são gerados usando a melhor rota possível dos pais, iterativamente. Eles usam duas funções de fitness diferentes, uma é para minimizar a distância total percorrida por todos os vendedores e o outro é para minimizar a distância máxima percorrida por qualquer vendedor. Outras restrições ou custos não são apresentados. Uma vez que Singh e Baghel propõem os melhores resultados computacionais até agora, discutiremos uma comparação com nosso método inovador na próxima seção.

Como pode ser visto na nossa revisão, a maioria das soluções anteriores para o mTSP reduz o problema em vários TSPs e aplica algum método para resolver sua combinação. Por outro lado, analisamos várias abordagens que são capazes de resolver o mTSP sem esta separação, no entanto, o número de vendedores é fixado ou não são apresentadas restrições adicionais, como o custo máximo por vendedor ou ambos. Portanto, introduzimos um método inovador, que atende a todos os nossos requisitos, capaz de resolver o mTSP considerando as restrições adicionais necessárias sem a transformação para vários TSPs.

4- Desenvolvimento do algoritmo

Como mencionamos anteriormente, nosso trabalho é derivado de um problema industrial, onde uma aplicação efetiva, fácil de usar e rápida é necessária para oferecer uma solução viável e quase ótima para o redesenho de fornecimento de mecânica móvel. A principal motivação de nossa pesquisa foi a falta de um algoritmo que seja "inteligente" o bastante para lidar com restrições no comprimento do percurso, distâncias assimétricas ou onde o número de vendedores não está predefinido e pode variar durante a evolução das soluções individuais. Para satisfazer essas condições, é necessária uma representação matemática refinada, que reflete o caráter composto da função de custo. **Nossas principais expectativas foram investigar um método genético, que pode suportar não apenas a implementação, mas a inicialização e o ajuste fino heurístico das rotas individuais facilmente.** Para este propósito, desenvolvemos **um novo algoritmo genético** usando uma representação diferente para resolver o mTSP. Com base nessa representação, **um conjunto de novos operadores genéticos foi definido para modificar os indivíduos com precisão suficiente.** Para melhorar a convergência do algoritmo, **desenvolvemos operadores complexos**, que combinam outros simples. Para provar a necessidade e precisão da representação do romance, será apresentada uma análise abrangente, comparando nosso método com as melhores abordagens publicadas, e recursos adicionais são publicados em nosso site, detalhando mais testes. Além disso, será discutida uma nova ferramenta automatizada, que fornece uma solução completa para o redesenho do fornecimento de mecânica móvel no principal fornecedor de energia da Hungria. A ferramenta implementada é capaz de otimizar um problema logístico que requer apenas um mapa definido na interface da web do Google Maps. Utilizamos a ferramenta automatizada para apoiar nossos testes detalhados também.

No restante desta seção, apresentaremos como a representação multi-cromossômica pode ser desenvolvida por novos operadores complexos. Portanto, uma breve visão geral da representação genética utilizada será seguida pela descrição dos operadores e a eficiência do nosso algoritmo será ilustrada por resultados de simulação.

4.1- Uma nova maneira de resolver MTSP com GA

Na Fig. 2, representamos como um algoritmo genético multi-cromossômico pode ser usado na solução de um mTSP com vinte cidades ($n=20$) e com cinco vendedores ($m=5$). Mais discussões podem ser encontradas em Tavares et al. (2003).

Como vimos anteriormente, muitos exemplos podem ser encontrados na literatura para operadores genéticos. A maioria desses operadores pode ser derivada de outros operadores,

por exemplo, uma mutação multi cromossômica pode ser construída a partir de **uma sequência de mutações cromossômicas únicas**. Os operadores descritos abaixo também podem ser criados a partir de outros operadores simples, mas a nova representação requer a introdução de novos operadores genéticos também. Existem dois conjuntos de operadores de mutação, as chamadas mutações na rota e as mutações da rota cruzada (Crossroute). Implementamos vários operadores para a nova representação, mas apenas uma visão geral deles será dada aqui.

Os operadores de mutação na rota trabalham dentro de um cromossomo, como a inversão da sequência do gene ou o giro, que apenas troca 2 genes dentro de um cromossomo. Podem encontrar-se boas referências para essas representações e operadores "clássicos" em Larranaga et al. (1999).

Um operador de mutação de rota cruzada modifica múltiplos cromossomos ao mesmo tempo. Note-se que usando a nomenclatura clássica e considerando os cromossomos como indivíduos, este operador pode ser muito parecido com o operador de cruzamento regular. FIG. 3 ilustra o operador quando são transpostas as sequências escolhidas aleatoriamente de genes de dois cromossomos, isto é, o "Swap". Se uma das sequências do gene estiver vazia, o operador será realizado como a inserção da sequência não vazia em um local escolhido aleatoriamente no outro cromossomo. O operador de cruzamento também é uma mutação de rota cruzada, que faz um cruzamento de um ponto entre dois vendedores. Também aplicamos o chamado operador de "Otimização local", que é um simples solucionador TSP usando o algoritmo genético. Este operador opera em cada vendedor e otimiza suas rotas separadamente. Seus benefícios serão discutidos na próxima seção.

Combinando operadores simples como o da Fig. 3, criamos operadores complexos, que podem acelerar o processo evolutivo, melhorando assim a eficiência do GA. FIG. 4 ilustra o método quando dois operadores de rotas cruzadas são aplicados um após o outro, compondo uma mutação complexa. Em primeiro lugar, é aplicado um operador de slide, que move o último gene de cada cromossomo para o início de outro; Posteriormente, uma Swap é aplicada produzindo o novo deslocamento.

Usando mutações simples para produzir complexos, uma hierarquia dos operadores pode ser construída. FIG. 5 mostra uma árvore, que representa os nossos operadores utilizados para testar a nova representação e algoritmo. Usando esses e outros operadores simples, podem ser gerados mais complexos.

OBS: Imagens presentes no artigo original

4.2- Análise numérica da representação proposta

Como nos apresentamos anteriormente em Király e Abonyi (2011), o uso da técnica multi cromossoma para o mTSP reduz o tamanho do espaço da solução, que é:

OBS: Fórmula presente no artigo original

é igual ao espaço da solução em Carter e Ragsdale (2006), mas esta abordagem é mais adequada para modelar um mTSP, por isso é mais específico do problema e, portanto, mais efetivo, já que será comprovado mais tarde no documento

Para analisar a nova representação, foi desenvolvido um GA usando essa técnica multicanal em MATLAB e o novo método foi comparado com a abordagem cromossômica mais conhecida (técnica cromossômica em duas partes). Para fazer uma comparação justa, desenvolvemos dois algoritmos diferentes, ambos baseados na implementação disponível no MATLAB Central.¹ O código real completo dos algoritmos está disponível em nosso site.

Os algoritmos usam duas matrizes como entradas, o conjunto de coordenadas dos locais (para visualização) e a matriz de distância que contém as distâncias de viagem entre duas cidades (em quilômetros ou em minutos). Claro, os parâmetros genéticos também devem ser especificados, como o tamanho da população, o número de iterações ou as restrições para o novo algoritmo. Primeiro, a população inicial é gerada, que consiste em indivíduos com genes permutados aleatoriamente. Em ambos os algoritmos, a função de aptidão simplesmente calcula a soma do comprimento total da rota (ou duração) de cada vendedor dentro de um indivíduo. Para comparação, os custos de *persalesmen*, salários, assimetrias ou penalidades não são incorporados nestes testes, apenas o número mínimo de cidades por rota é fixado em 3. Estes parâmetros adicionais serão considerados em nossa nova AG, que será discutida na Seção 5.

A seleção é seleção de torneios, onde, no nosso caso, o tamanho do torneio (ou seja, o número de indivíduos que competem pela sobrevivência) é 8. O indivíduo com o menor valor de aptidão ganha o torneio, portanto será selecionado para geração de novos indivíduos, e esse membro será transferido para a nova população sem qualquer modificação.

Para analisar a eficácia da nova representação, foi testado usando vários exemplos, apenas um contendo 1 depósito e 100 locais adicionais é apresentado aqui em detalhes.

Os resultados são apresentados na Fig. 6. A nova representação foi comparada com a chamada abordagem cromossômica de duas partes (Carter e Ragsdale, 2006), que é a melhor técnica para otimizar o mTSP usando um único cromossomo até agora. É evidente que dois algoritmos genéticos diferentes não podem ser comparados de forma bastante justa, porque o desempenho desses métodos estocásticos depende muito de seus parâmetros, portanto, é impossível encontrar um conjunto de parâmetros que resulte em um ótimo desempenho para ambos os algoritmos. No entanto, aspiramos a fazer as representações bastante comparáveis, portanto, dois algoritmos diferentes foram desenvolvidos, mas ambos baseados na mesma implementação disponível na web. Os dois métodos são quase iguais, apenas a representação genética e os operadores aplicados são diferentes.

Em cada caso, o tamanho da população era 80 e o número da iteração era de 20.000. FIG. 6 mostra um resultado médio de 100 execuções dos algoritmos. Durante uma única corrida, a população inicial das duas variedades do algoritmo era a mesma. A figura mostra inequivocamente que a nova abordagem produz melhores resultados nesses casos. O mínimo estabelecido é melhor no caso de múltiplos cromossomos, e esta técnica pode convergir para a ótima mais rapidamente. No exemplo acima, a abordagem multi-cromossomo precisava apenas de iterações 7411 para encontrar o valor 996, enquanto o cromossomo de duas partes técnicas requeria 18.861 iterações. Os tempos de execução dos algoritmos foram quase iguais, o que é devido à mesma complexidade das representações. Assim, as representações podem ser comparáveis somente no caso de melhor valor de função objetivo por iteração. Esses resultados do teste confirmam a eficiência da representação proposta para resolver problemas de mTSP usando algoritmos genéticos. Testamos o novo método com vários exemplos sintéticos gerados aleatoriamente, variando o número de locais de entrada, o tamanho da população e o número de iterações. Alguns deles são apresentados na Tabela 1. Os resultados na tabela apresentam uma comparação justa, enquanto os algoritmos eram muito semelhantes, ambos foram executados no mesmo ambiente. Uma lista quase completa dos casos de teste e os algoritmos estão disponíveis em nosso site.

A Tabela 1 resume os resultados de nossos testes usando conjuntos de dados sintéticos. Todos os resultados apresentados durante o trabalho foram gerados em um PC com processador Core i5, 2,66 GHz com 3 GB de RAM. A Tabela 1 mostra claramente que a abordagem inovadora pode encontrar soluções com menor distância geral, e pode encontrar esta solução durante menos iterações. O tempo necessário para encontrar o ótimo era quase idêntico, portanto, a abordagem inovadora pode ser considerada mais eficaz nestes casos de teste.

OBS: Tabelas e gráficos presentes no artigo original

A Tabela 2 representa os resultados de nossos testes usando inicialização na população inicial e usando operadores de mutação complexos (ver Seção 4.1 e Figuras 4 e 5). A inicialização foi feita por um processo de otimização local, a saber, um solucionador TSP em cada vendedor em cada indivíduo da população. Após a inicialização, os operadores simples foram usados (veja a 2ª coluna). Operadores complexos usando o Opt local. Ou L.O. Na Fig. 5 apliquem o mesmo método. A coluna 3 indica o caso quando a otimização foi feita sem a etapa de inicialização, usando os operadores complexos (veja o 3º nível da hierarquia dos operadores na Fig. 5). Nossos testes destacam que o uso da otimização local inicial pode melhorar a precisão e a velocidade do processo, enquanto a aplicação de operadores complexos resulta em convergência muito mais forte, mas torna a abordagem mais lenta e a precisão também é um pouco pior. Esse resultado implica a necessidade de inicialização por otimização local (por exemplo, usando o TSP solver), mas indica que devemos ter cuidado com o uso de operadores muito complexos. No entanto, como esses operadores podem produzir uma variabilidade muito maior na população do que os simples, eles podem produzir melhores resultados em espaços de pesquisa altamente complexos. Claro, a seleção de operadores adequados pode diferir de problema a problema.

Para analisar o desempenho e a escalabilidade do nosso método, outros testes foram realizados utilizando problemas de teste bem conhecidos da TSPLIB (Reinelt, 1991) e os dados de teste de Carter e Ragsdale (Carter e Ragsdale, 2006) que foram utilizados em Brown Et al. (2007) e Singh e Baghel (2009). Como as linguagens de programação e os ambientes em execução diferem muito, ignoramos a apresentação dos tempos de execução nas tabelas a seguir. A Tabela 3 apresenta nossos testes com cinco casos de TSPLIB com tamanho crescente, usando 5 vendedores. Nossos resultados são comparados com o algoritmo Ant The Colony Optimization, que mostrou-se melhor do que os GAs em Junjie e Dingwei (2006). Os resultados mostram inequivocamente que o nosso novo método funciona melhor especialmente em enormes problemas.

Singh e Baghel (2009) propuseram um algoritmo genético de agrupamento muito poderoso para mTSP, nomeado GGA-SS. Este método foi analisado utilizando os dados de Carter e Ragsdale (2006) em comparação com seu algoritmo, GGA2PC (usando representação cromossômica de duas partes). Nossos testes de desempenho são apresentados na Tabela 4 usando os mesmos dados que Singh e Bagel e nossas implementações em MATLAB. Embora

o GGA-SS seja melhor nos casos de maior número de vendedores, nossa abordagem pode encontrar melhores soluções na maioria dos casos.

Para testar os diferentes métodos em situação real, nós os usamos em nosso problema industrial (veja a próxima seção), onde os custos por trabalhador, os salários dos motoristas, os custos por km, as distâncias de condução reais, as janelas de tempo e as penalidades são apresentados. Utilizamos o serviço de planejamento de rotas do Google para calcular as distâncias de condução e as despesas reais foram fornecidas pelo nosso parceiro industrial:

- Reparar o custo por vendedor (preço de um caminhão novo): 4 M HUF

~ 18,000 USD;

- Salário do motorista: 1550 HUF / h ~ 7 USD / h;
- Custo por km: 150 HUF / km ~ 0,7 USD / km;
- Janela de tempo: 7 min.

As funções de fitness foram modificadas para lidar com as diferentes despesas. Após a otimização, os resultados são concluídos na Tabela 5. O problema contém 30 cidades no Transdanúbio, que são iguais aos locais de base reais de nosso parceiro industrial (ver Seção 5.2). O número máximo de vendedores é 6 e o comprimento mínimo do passeio é 3. Usando custos reais, o método multi cromossomo serve uma solução mais barata, que é principalmente baseada em sua capacidade de minimizar o número de vendedores também.

5- Estudo de aplicação

Apresentamos uma metodologia completa nesta seção, que demonstra uma nova estrutura baseada em componentes usando tecnologias da web e MATLAB. Também será apresentado um estudo de aplicação que orientará o leitor através de um projeto industrial a partir da definição do problema para a visualização de resultados. O framework baseado na web apresentado nesta seção depende muito da API do Google Maps, que é única na literatura até agora.

5.1- Desenvolvimento do framework

Com base na nova representação genética, desenvolvemos um novo algoritmo, que difere dos algoritmos mencionados na seção anterior, uma vez que é capaz de lidar com restrições adicionais e janelas de tempo, que foram discutidas na Seção 2. O algoritmo pode minimizar o número de vendedores incluídos na solução também, considerando seu custo inicial. O documento passa sobre a apresentação dos códigos-fonte, que são acessíveis em nosso site. Observe que a penalização de rotas que excedem as restrições é realizada como uma divisão, usando o operador de partição cromossômica em vez da atribuição de um valor de fitness proporcionalmente alto. Uma vez que o algoritmo minimiza o número de vendedores envolvidos, esta penalidade tem um efeito notável no processo de otimização. Além disso, o algoritmo pode lidar com as restrições para as rotas e as janelas de tempo para os locais também, e julgamos que a representação aplicada é mais semelhante para a característica do problema, do que as até agora, portanto, pode ser mais facilmente compreensível e realizável.

O problema em si, que motivou essa pesquisa, pode ser abordado como um problema de mTSP, que deve ser resolvido por um sistema automatizado. Um diagrama de fluxo esquemático de tal sistema é representado na Fig. 7. A linha de base deve ser um mapa (por exemplo, definido no Google Maps) e as distâncias entre os locais podem ser calculadas por um serviço baseado na web, como o Google Maps. Uma vez que na aplicação real a quantidade de mercadorias entregues é muito inferior à capacidade dos veículos, as restrições de volume ou de massa podem ser ignoradas (o que define o problema como um mTSP em vez de um VRP). A saída do sistema deve ser um plano de rota que pode ser definido em um mapa baseado na web ou em um dispositivo GPS. O cálculo dos custos, do tempo e dos fluxos de materiais também são necessários. Nós decidimos usar a API do Google Maps gratuita e publicamente disponível, pois oferece um serviço web rápido e confiável para definir de maneira amigável mapas, informando distâncias de viagem e tempo, e visualizando rotas. Além disso, é o serviço de mapeamento mais popular hoje em dia.

Com base nos serviços do Google, desenvolvemos uma estrutura completa e automatizada para fornecer um sistema automatizado como o apresentado na Fig. 7. Com a ajuda deste programa, os usuários conseguem otimizar um problema mTSP definido em um mapa do Google Maps em alguns minutos, e o resultado da computação é visualizado de forma realmente facilmente interpretável. Um exemplo concreto esclarecerá estas declarações na próxima seção.

O método apresentado fornece uma ferramenta muito útil para resolver problemas de transporte, bem como um sistema útil de suporte a decisões. A recuperação de coordenadas

de um Google Maps, o cálculo da tabela de distância, a otimização e a visualização são todos processos automatizados, que tornam os usuários capazes de otimizar um problema complexo de forma interativa, gratuita e fácil através de aplicativos web.

5.1- Solução de um problema industrial

O sistema a ser otimizado contém mais de 500 mecânicos móveis, cujo suprimento é gerenciado na chamada topologia em estrela (veja a Fig. 1). Neste sistema, a própria mecânica deve coletar as ferramentas / recursos dos armazéns. É uma solução muito ineficiente, uma vez que os salários da mecânica móvel são relativamente altos, e durante a coleta de ferramentas e dispositivos, eles são incapazes de fazer sua principal tarefa: fixação e instalação. Isso causa não só perda financeira, mas também reduz o nível de serviço. Portanto, esta topologia deve ser substituído por um sistema de rotas, onde os produtos são transportados por caminhões, considerando restrições adicionais como o janelas-tempo.

As atividades da mecânica móvel são controladas por um Sistema de Gerenciamento de Serviço de Campo (FSMS) com o objetivo de otimizar dinamicamente as tarefas de manutenção planejadas / não planejadas na rede de energia / gás. Uma vez que a viagem para o material da mecânica móvel é uma tarefa não produtiva no FSMS, o fornecimento de 20 armazéns (Regional, central) representa um problema complexo e economicamente importante para o E.ON Network Services Kft. A nova abordagem dinâmica apresentada neste artigo tem como objetivo reduzir significativamente as atividades, no que diz respeito ao manuseio de materiais para as equipes móveis, com o aumento de locais de atendimento de 20 a 100. O design do sistema de suprimento pode ser considerado como um problema complexo de otimização combinatória, onde o objetivo é encontrar um plano de rota com custo mínimo de rota, que presta serviços a todas as demandas de três armazéns centrais, ao mesmo tempo em que satisfaz outras restrições, como as janelas de horário.

Desenvolvemos um pacote de software completo para resolver esse tipo de problemas de otimização. Os dados de entrada são fornecidos por um mapa do Google Maps, que contém os locais (incluindo o depósito) e o resultado final é um sistema de rotas definido por um mapa do Google Maps também. Uma solução completa e automatizada está disponível gratuitamente em nosso site como um serviço baseado na web. No entanto, deve ser mencionado que por uma questão de reduzir a carga em nossos servidores, este aplicativo fornece apenas uma demonstração, onde o número de locais de entrada é Limitado a 10. Após um planejado

desenvolvimento futuro de TI em nosso departamento, essas restrições serão removidas. Portanto, o projeto real foi otimizado por um algoritmo genético offline. Na seção a seguir, vamos demonstrar a otimização do problema industrial, no entanto, apenas locais representativos são mostrados.

Em primeiro lugar, o usuário deve definir um mapa com o serviço do Google. Neste exemplo, contém 30 locais. Necessita de planejador de rotas do Google, portanto, usamos um método simples de mineração de texto para recuperá-los deste mapa. O resultado desta etapa é uma planilha do Excel, que serve como entrada para o próximo estágio, que é a determinação da matriz de distância. É calculado por chamadas iterativas da API do Google Maps, que fornece distâncias de viagem reais entre cada par de locais, que são definidos por longitude e coordenadas de latitude, usando dados de tráfego. Uma porção da tabela de distância gerada, que contém 30 locais (incluindo o depósito), é mostrada na Tabela 6. A tarefa é determinar as rotas ótimas para esses locais com as seguintes restrições: o número máximo de vendedores é 5; A distância máxima de viagem de cada vendedor é de 450 km; A duração máxima de viagem de cada vendedor é de 8 h.

Após a determinação da matriz de distância, o algoritmo calcula a solução com a nova representação. Nesse caso, nossa GA leva em consideração os custos e as assimetrias por vendas, e inclui salários e multas na função objetiva. Uma vez que todos os salários e as distâncias de viagem são proporcionais à duração total da condução, e a restrição de tempo (8 h, incluindo janelas de tempo para embalagem) é geralmente mais rigorosa do que a restrição de distância (450 km), fizemos o nosso algoritmo minimizar a duração. Devido à maior complexidade do problema, o tamanho da população deve ser aumentado para garantir uma convergência rápida. Durante o nosso ajuste manual dos parâmetros, descobrimos que é eficiente executar o GA com o tamanho da população 320 e fazer 200 iterações.

Descobriu-se que quatro vendedores são suficientes para satisfazer as restrições. Com o componente visualizador, podemos visualizar os resultados, como é mostrado na Fig. 8. Também é capaz de calcular os custos reais para cada rotas, considerando o custo por quilômetro, o salário por hora dos motoristas e a equipe de empacotamento (incluindo janelas de tempo), dada por uma tabela do Excel. Dá a oportunidade de atribuir diferentes salários a diferentes funcionários e custos diferentes para caminhões diferentes. O comprimento das rotas calculadas em nosso exemplo é de 261 km, 422 km, 333 km e 384 km, respectivamente, e a duração das rotas considerando 5 min para embalagem é de 304, 503, 368 e 471 min respectivamente. Os resultados satisfazem as restrições, pelo que o algoritmo forneceu uma

solução viável do problema. O algoritmo requer os dados de entrada como tabelas do Excel e fornece o sistema de rota resultante como uma tabela do Excel também. O número de mecânica para cada base é fornecido pela empresa, bem como a duração das janelas (geralmente 5-7 min).

Tanto quanto sabemos, nossa solução automatizada baseada na web, incluindo a recuperação de informações de um mapa do Google Maps, a determinação da matriz de distância e os componentes do visualizador que podem traçar as rotas finais uma após a outra são todas ferramentas exclusivas.

6- Conclusões

A viagem pelo material da mecânica móvel é uma tarefa não produtiva. Neste artigo, apresentamos uma nova abordagem para a otimização de locais de atendimento para reduzir essas atividades. Um mTSP modificado com restrições adicionais foi introduzido e resolvido por este método. A complexidade do problema motivador implicou a introdução de um algoritmo genético melhorado usando novos operadores cruzados para a representação individual multi-cromossomo, onde um cromossomo separado é atribuído a cada vendedor. A abordagem apresentada aqui é inovadora na reprodução de indivíduos, no tratamento das restrições, e dá toda uma metodologia e uma nova estrutura completa para resolver um problema NPhard, o mTSPTW. Além da metodologia proposta, o documento apresentado a ferramenta desenvolvida utiliza o Google Maps para visualizar a estrutura da oferta e coletar dados brutos utilizados para otimização. A nova abordagem dinâmica resulta em redução significativa de atividades em relação ao manuseio de materiais para as equipes móveis, estendendo os locais de atendimento de 20 a 100.

A aplicação da nova ferramenta contendo o processo de otimização e a estrutura baseada na web na solução de um problema industrial real justificou a necessidade da pesquisa. E.ON Hungaria Zrt. já aplicou a ferramenta proposta. Cálculos econômicos preliminares e experiências mostram que a implementação aumentou as economias significativas, enquanto a qualidade do serviço também melhorou.

OBS: Fórmulas, imagens e tabelas presentes no artigo original