

baixar

PDF

Acesso fornecido por:

UNIVERSIDADE FEDERAL DO PIAUI

Sair

Squeaky toy

Minhas configurações

Obter ajuda

Conferências > IEEE International Workshop o ... IEEE International Workshop o ...IEEE International Workshop o ...

Análise Comparativa do Algoritmo Genético e Algoritmo de Colônia de Formigas na Solução do Problema do Vendedor em Viagem

4 Autor (es)

Kangshun Li ; Lanlan Kang ; Wensheng Zhang ; Bing Li

Visualizar todos os autores

3

Papel

Citações

504

Cheio

Exibições de texto

Exportar

para

Collabratec

Alertas

Gerenciar alertas

de conteúdo

Adicionar aos

Alertas de

Citação

Mais como isso

Um Algoritmo Genético Equilibrando Exploração e Exploração para o Problema do Vendedor Viajante

2008 Quarta Conferência Internacional sobre Computação Natural

Publicado em: 2008

Uma Nova Abordagem para Resolver o Problema do Vendedor Ambulante Usando Algoritmo Genético Baseado no Cruzamento Heurístico e Operador de Mutação

Conferência Internacional de Soft Computing e Reconhecimento de Padrões 2009

Publicado: 2009

Veja mais

Veja as principais organizações de patentes em tecnologias mencionadas neste artigo



Clique para expandir

Provided by: Innovation PLUS

POWERED BY IEEE AND IP.COM

A PATENT SEARCH AND ANALYTICS TOOL

Abstrato

Seções do documento

EU. Introdução

II. BREVE INTRODUÇÃO AO TSP

III O princípio da ACA e seu método para resolver o TSP

IV. GA o princípio e método para resolver TSP

V. Análise comparativa dos dois algoritmos

Mostrar esboço completo ▾

Autores

Figuras

Referências

Citações

Palavras-chave

Métricas

Mais como isso

Resumo:

Algoritmo de Colônia de Formigas e Algoritmo Genético (GA), dois algoritmos de otimização de inspiração biônica, têm grandes potencialidades para resolver os problemas de otimização de combinações, ... Ver mais

Metadados

Abstrato:

Algoritmo de Colônia de Formigas e Algoritmo Genético (GA), dois algoritmos de otimização de inspiração bionica, têm grandes potencialidades para resolver os problemas de otimização combinados, respectivamente usados na resolução de problemas de caixeiros viajantes, mas há algumas falhas se apenas um deles é usado para resolver TSP . Análise comparativa de desempenho foi feita usando ACA e GA, respectivamente, na resolução de TSP neste documento. Os experimentos mostram as vantagens e desvantagens utilizadas apenas ACA ou GA, podemos superar as deficiências se GA e ACA são combinados para resolver TSP e obter velocidade convergente mais rápida e resultados mais precisos em comparação com apenas usando ACA ou GA.

Publicado em:

IEEE International Workshop em Computação e Sistemas Semânticos

Data da Conferência:

14 a 15 de julho de 2008

Número de Acesso INSPEC :

10090202

DOI:

10.1109 / WSCS.2008.11

Editora:

IEEE

Data adicionada ao IEEE Xplore :

22 de julho de 2008

Local da Conferência:

Huangshan, China

CD-ROM ISBN:

978-0-7695-3316-2

Conteúdo

SEÇÃO I.

Introdução

Algoritmo de Colônia de Formigas (ACA) e Algoritmo Genético (GA) esclarecido pelo evolucionismo biológico são apresentados como uma espécie de algoritmos biônicos. Mas eles compartilham alguns bens comuns e têm diferenças ao mesmo tempo. O algoritmo genético simula o processo evolutivo do gene, enquanto o

Conteúdo

SEÇÃO I.

Introdução

Algoritmo de Colônia de Formigas (ACA) e Algoritmo Genético (GA) esclarecido pelo evolucionismo biológico são apresentados como uma espécie de algoritmos biônicos. Mas eles compartilham alguns bens comuns e têm diferenças ao mesmo tempo. O algoritmo genético simula o processo evolutivo do gene, enquanto o

Algoritmo Ant Colony difere muito dele. Em outras palavras, a ACA imita os comportamentos da colônia composta por um sistema simples de indivíduos-baixaciedade. Os dois algoritmos foram aplicados para resolver alguns problemas de PD otimização combinatória e conseguiram muitas conquistas.

Neste documento, tanto o GA quanto o ACA serão aplicados para resolver o problema do vendedor ambulante (TSP) [1]. Através da análise de suas vantagens e desvantagens são dadas por meio de um grande número de experimentos, respectivamente, alguns estudos futuros são fornecidos também.

SEÇÃO II

BREVE INTRODUÇÃO AO TSP

O Problema do Vendedor Viajante (TSP) [1] é o problema mais estudado na otimização combinatória. O objetivo é encontrar um ciclo Hamiltoniano de custo mínimo, isto é, um caminho através de um gráfico que começa e termina no mesmo vértice e inclui todos os outros vértices exatamente uma vez.

As razões dominantes para escolher o TSP para testar ACA e GA são ilustradas da seguinte forma:

1. O TSP é um problema de caminho mais curto; A ACA e a GA estão adaptadas para resolver esse tipo de problema.
2. Não é uma tarefa fácil entender o TSP devido à falta de muitos termos técnicos.
3. O TSP é um problema típico de otimização combinatória.

É frequentemente utilizado para validar um determinado algoritmo, tornando a comparação com outros algoritmos um trabalho fácil.

SEÇÃO III

O princípio da ACA e seu método para resolver o TSP

ACA [2], [3] foi primeiramente proposto por três estudiosos da Itália em 1991, Colomi A, Dorigo M e Maniezzo V., e tem recebido atenção crescente de pesquisadores, tendo sido usados para resolver muitos problemas difíceis na otimização de sistemas discretos. O processo utilizou principalmente a capacidade das formigas para procurar o melhor caminho de otimização enquanto procuravam por alimentos. Como se falou acima, a ACA fez uso total do mecanismo de comportamento das formigas reais na busca de comida. Enquanto procuram comida, as formigas reais usam um certo material chamado feromônio nos caminhos, que são deixados pelas formigas, para se comunicar com outras formigas. Nas fugas a falta de formigas de feromona mais ou menos executa um passeio fortuito. No entanto, assim que percebem que um feromônio trilha um caminho na vizinhança, é provável que sigam esse caminho, reforçando essa trilha. Mais especificamente, se as formigas perceberem mais de um rastro de feromônio, escolherão uma dessas trilhas com uma probabilidade relacionada às forças das trilhas existentes. Esta ideia foi aplicada primeiramente ao TSP, onde uma formiga localizada em uma cidade escolhe a cidade seguinte de acordo com a força das trilhas artificiais.

Os símbolos são definidos da seguinte forma:

E se m e n denota o número de formigas e o número de cidades respectivamente; d_{ij} ($i, j = 1, 2, \dots, n$) é referido como a distância da cidade i para a cidade j , $\tau_{ij}(t)$ a quantidade de feromônio entre a cidade i e a cidade j no tempo t ,

1. Na inicialização, os feromônios em todos os caminhos são iguais, suponha

$\tau_{ij}(0) = C$ (C é constante). No curso do movimento de uma formiga k (Onde $k = 1, 2, \dots, m$), a direção de transferência é determinada pela probabilidade de feromônio como uma variável em cada caminho, que pode ser escrito como:

$$p_{eu}^k(j|t) = \begin{cases} \frac{\tau_{eu}(t) \cdot \eta_{eu}^{\beta}}{\sum_{s \in \text{uma l l o w e d}} \tau_{is}^{\alpha} \cdot \eta_{is}^{\beta}} & \text{eu f j} \in \text{uma l l o w e d}_k \\ 0 & \text{O t h e r w i s d e e} \end{cases} \quad (1)$$

[Ver fonte](#)

Onde η_{eu} significa a visibilidade na borda (i, j) e pode ser calculada por uma certa função heurística, que é frequentemente descrita como:

$\eta = 1 / d_{eu}(j|t)$ a probabilidade de que a formiga k move posição de forma a posicionar j no tempo t . $\text{uma l l o w e d}_k = \{0, 1, \dots, m - 1\}$ denota a $ab u_k$ ant k tem sido permitido de escolha das cidades na próxima etapa ($t a b u_k(k = 1, 2, \dots, m)$ é a agregação que a formiga já atravessou as cidades. No começo há apenas um elemento que é a cidade inicial de ant k , com o desenvolvimento da evolução, elemento $not a b u$ será aumentado). α é a relativa ponderabilidade da pista. β é o parâmetro que determina a influência da função heurística.

2. Uma vez que todas as formigas tenham passado pela construção da solução e pela pesquisa local, o procedimento de atualização do feromônio é aplicado a esses valores de feromônio. A atualização utilizada para o algoritmo apresentado neste trabalho é uma variante da abordagem proposta em [8]. Pode ser escrito como:

$$\tau_{eu}(j|t + 1) = (1 - \rho) \tau_{eu}(j|t) + \Delta \tau_{eu}(j|t) \quad (2)$$

[Ver fonte](#)

Onde, $\rho \in (0, 1]$ significa grau de atenuação de $\tau_{eu}(j|t)$ com o processo do tempo, $\Delta \tau_{eu}$ é a quantidade de reforço na borda (i, j) que é definido como:

$$\Delta \tau_{eu}(j|t) = \sum_{k=1}^m \Delta \tau_{eu}^{(K)}(j|t) \quad (3)$$

[Ver fonte](#)

$\Delta \tau_{eu}^{(K)}(j|t)$ significa feromônios que a formiga se move da cidade i para a cidade j neste círculo. De acordo com diferentes formas de expressão de $\Delta \tau_{eu}^{(K)}(j|t)$, Dorigo M já apresentou diferentes modelos, conhecidos como Ant-ciclo System, Ant-quantidade System, sistema de densidade de formigas. Suas fórmulas são definidas como segue (4) ~ (6):

No sistema de ciclo de formigas:

$$\Delta \tau_{eu}^{(K)}(j|t) = \begin{cases} \frac{Q}{c_k}, & \text{eu f t h e u m a n t k u s e r e d g e (i j)} \\ 0, & \text{o t h e r w i s d e e} \end{cases} \quad (4)$$

[Ver fonte](#)

No sistema da quantidade da formiga:

$$\Delta \tau_{eu}^{(K)}(j|t) = \begin{cases} Q, & \text{eu f t h e u m a n t k u s e r e d g e (i j)} \\ 0, & \text{o t h e r w i s d e e} \end{cases} \quad (5)$$

[Ver fonte](#)

No sistema de densidade de formigas:

$$\Delta \tau_{eu}^{(K)}(j|t) = \begin{cases} \frac{Q}{d_{eu}^j}, & \text{eu f t h e u m a n t k u s e r e d g e (i j)} \\ 0, & \text{o t h e r w i s d e e} \end{cases} \quad (6)$$

[Ver fonte](#)

Onde Q é uma constante que denota a informação total de que cada formiga deixa

depois de viajar uma vez ao redor de todas as cidades; e_{ij} é o comprimento dos caminhos que a formiga tem percorrido nesse ciclo; d_{eu} significa distância de duas cidades i e j .

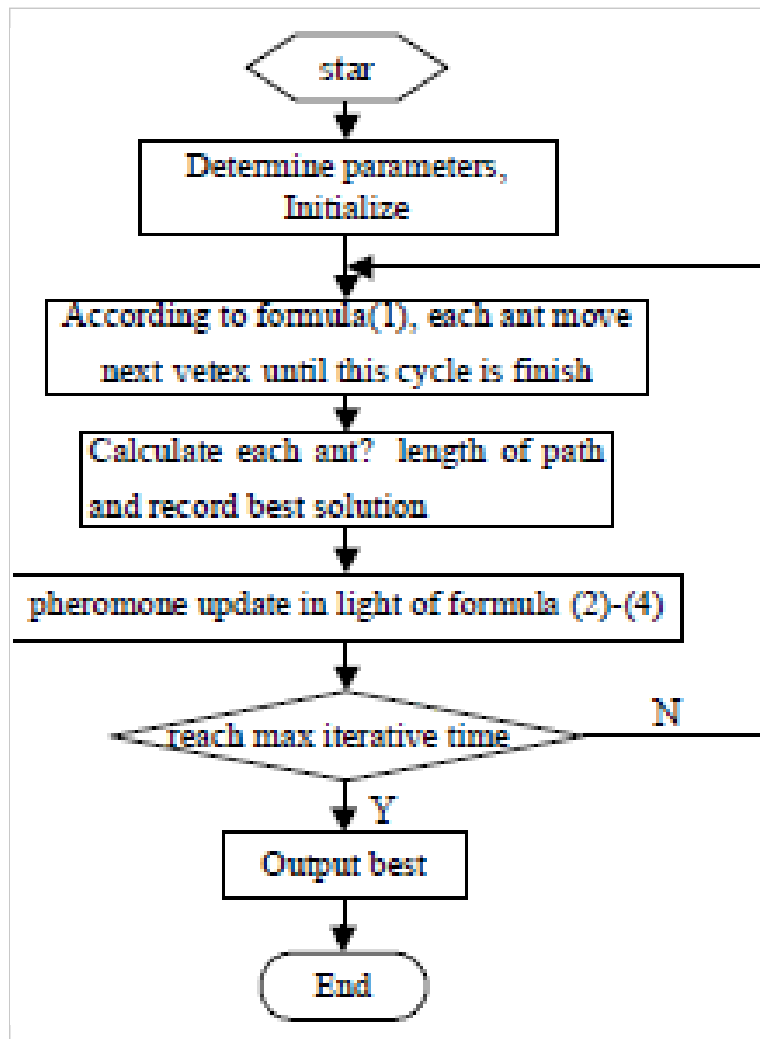


FIGURA 1 O fluxograma da ACA

A diferença mencionada acima é que o sistema Ant-cycle usa informações completas, no entanto, o sistema Ant-quantity e o sistema Ant-density usam informações locais. No processo de resolver o problema do vendedor ambulante (TSP), obviamente, a capacidade do sistema Ant-cycle é preferível, de modo que este artigo adota o sistema Ant-cycle.

1. Circulando os passos acima até o momento da viagem atingir o tempo designado ou não aparecer melhores soluções em um determinado período de tempo. O fluxograma do Algoritmo de Colônia de Formigas é mostrado na FIGURA 1.

SEÇÃO IV.

GA o princípio e método para resolver TSP

O Algoritmo Genético (GA) [4] é um algoritmo de busca heurística adaptativa baseado nas idéias evolucionárias de seleção natural e seleção genética. O conceito básico de GA é projetado para simular processos no sistema natural necessário para a evolução, especificamente aqueles que seguem os princípios estabelecidos pela primeira vez por Charles Darwin de sobrevivência do mais apto. Os GAs são algoritmos de busca com probabilidade aleatória e possuem processo iterativo de “sobrevivência + detecção”. Usando a tecnologia estocástica, os GAs realizam pesquisas eficientes em um espaço de parâmetro codificado.

Operação genética no GA incluindo três partes - seleção, cruzamento e mutação. Os

parâmetros de codificação, as configurações do grupo inicial e do parâmetro de controle, o design da função de adequação e a manipulação genética compõem o núcleo do GA.

Normalmente, o design de algoritmos genéticos inclui principalmente os seguintes passos:

1. Escolha os esquemas de codificação de dados e gerou aleatoriamente um grupo de indivíduos iniciais como a colônia inicial.

Este artigo adota um melhor esquema de codificação natural para viajar: representação do caminho. Por exemplo, no pressuposto de que (7, 4, 9, 5, 6, 1, 2, 8, 10) denota um caminho que partiu da cidade sete e passa pela cidade 4-9-5-6-1-2-8-10, finalmente, retorna para a cidade quatro.

2. Função de aptidão é fornecida para avaliar os indivíduos e avaliar o valor de aptidão de cada indivíduo.

Avaliar indivíduos é calcular o comprimento de cada caminho, que também é referido como a aptidão. Pode ser escrito como:

$$f(x) = \sum_{eu=1}^{n-1} d(C_{eu}, C_{i+1}) + d(C_1, C_n)$$

[Ver fonte](#)

O indivíduo com menor valor de aptidão significa que ele tem um caminho mais curto e o indivíduo pode ser melhor também.

3. O resultado da pesquisa é exportado se o algoritmo acima estiver satisfeito com critérios convergentes, caso contrário, execute as etapas seguidas.
4. De acordo com o valor da aptidão, copie os indivíduos para a próxima geração por um determinado método.

Após a produção da colônia inicial, a aptidão de cada indivíduo na colônia será calculada. De acordo com o valor de cada indivíduo, classifique os indivíduos de pequeno a grande porte. Defina a porcentagem de escolha P, a fim de manter a diversidade da colônia, selecione p / 2 como bom o número de indivíduos armazenados em mating pool e, em seguida, realizou operações genéticas, como crossover, mutação.

5. Executar operação de crossover de acordo com a probabilidade de crossover p_c .

Este artigo adota o operador OX apresentado por Davis. Nomeadamente, estruturando a descendência através da escolha de um subconjunto de sequências para viajar das sequências dos pais e salvando-a como sequência relativa das sequências dos pais

6. Executar operação de mutação de acordo com a probabilidade de mutação p_m .

Este artigo adota a mutação inversa. Ou seja, a subsequência entre os deuses é revertida após a escolha aleatória do deuce.

7. Retornar a etapa 3)

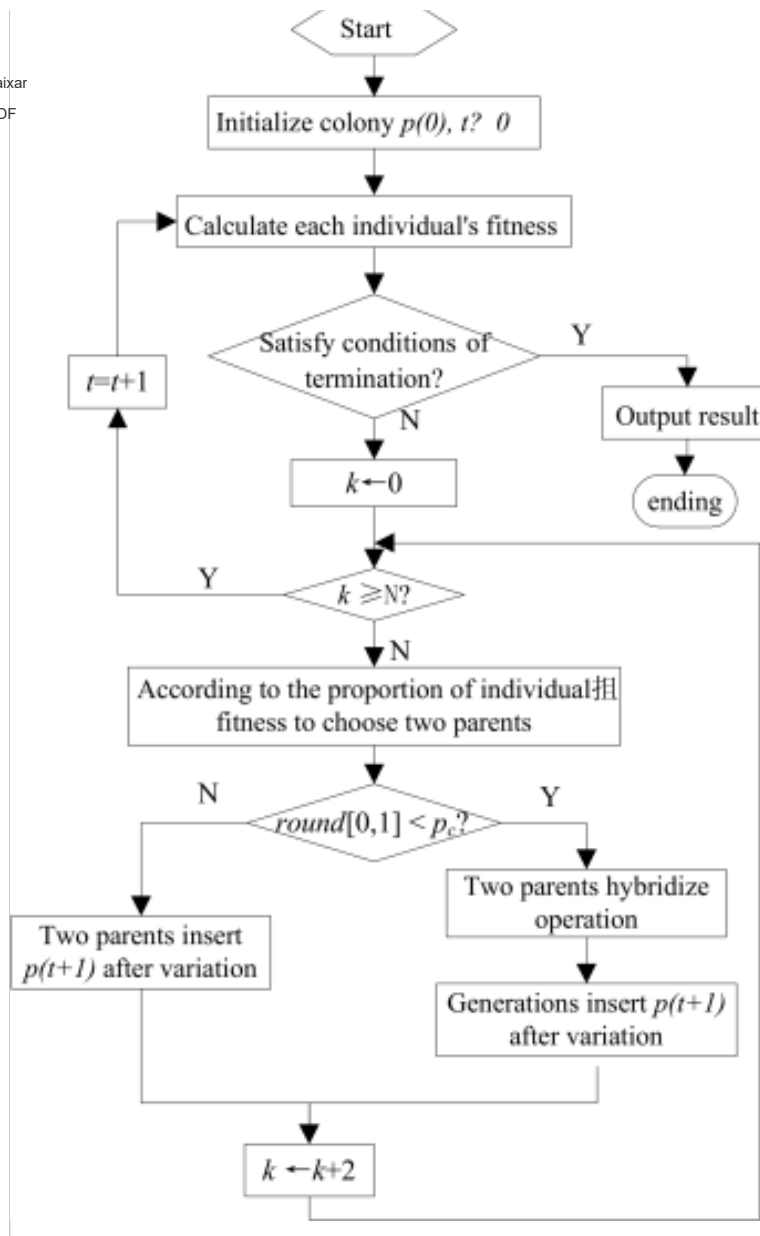


FIGURA 2 O FLUXOGRAMA DO GÁS

A FIGURA 2 mostra o fluxograma do GA.

SEÇÃO V. Análise comparativa dos dois algoritmos

Este artigo adota o sistema de colônia de formigas (ACS) como representação do ACA que é comparado com o GA básico. Resultados simulados de dois algoritmos são apresentados sobre o Problema do Vendedor em Viagem (Travelling Salesman Problem - TSP) com o número de cidades 30, 48 e 51, respectivamente. Dois algoritmos executam tempos de 20 e 50 respectivamente nas mesmas condições, ao mesmo tempo, os resultados de diferentes problemas em TSPLIB [6] são apresentados. Os parâmetros dos algoritmos acima são os seguintes:

$$\rho = 0,1, \alpha = 1, \beta = 5, m = 60, C = 10, Q = 1000, p_c = 0,75, p_m = 0,05$$

[Ver fonte](#)

A partir dos experimentos acima e da Tabela 1, sabemos que o ACA e o GA são todos algoritmos de busca probabilística biônicos que combinam com computação distribuída, tendem a computar paralelamente e são mais robustos. A ACA simula

comportamentos de grupo entre comunidades e ambiente compostos por

indivíduos simples, assim como indivíduos, viz. sistema social [5] , que pode causar um comportamento imprevisível do grupo, mas GA simula o processo evolutivo genético. Eles têm motivos comuns para a realização do Problema do Vendedor PDViajante (TSP):

1. Eles são todos propensos à convergência prematura, de modo a entrar em valor local ideal
2. Algoritmos se na escolha de parâmetros e seus 'valores não podem ser rigorosamente comprovados em teoria, apenas através do método experimental para determinar a combinação ótima.
3. Eles não têm requisitos especiais para a busca de espaço, como derivativos, continuidade, concavo-convexa e outras informações auxiliares, ao mesmo tempo em que suas faixas práticas são mais difundidas.

TABELA 1 DOIS RESULTADOS DOS ALGORITMOS DE EXPRESSÃO SOBRE O DIFERENTE PROBLEMA DE VENDEDOR DE VIAGEM

nome do problema	vezes	ACS			GA			TSPLIB
		melhor	média	pior	melhor	média	pior	
Oliver30	20	423,74	429,7	432,46	423,74	456,68	502,57	423,74
	50	423,74	428,76	432,45	423,74	425,7	439,84	
att48	20	33780	35595	36534	37880	38833	38894	33522
	50	33780	35533	36534	35633	38541	42458	
Ei151	20	426	428,4	431,42	495	512	536	426
	50	426	428	432,01	494	514	554	

Diferença:

O algoritmo de colônia de formigas converge para o caminho ideal através do acúmulo e atualização de informações, mas a falta de feromônio nos estágios iniciais leva a uma velocidade de convergência mais lenta. Algoritmos genéticos têm capacidade de busca global rápida, mas o feedback de informações no sistema não foi utilizado, às vezes levando a iteração redundante e ineficiente.

Portanto, se o tamanho das cidades for superior a 30, a capacidade de pesquisa dos algoritmos genéticos diminuirá gradualmente até certo ponto, quando o número das cidades for muito grande, não poderá obter a solução ótima em iteração finita, porque os tempos iterativos são muito longos e insuportáveis. Aqui o algoritmo de colônia de formigas é melhor que o algoritmo genético, e pode alcançar o valor ótimo. Quando o tamanho das cidades é muito grande, o algoritmo de colônia de formigas pode aparecer estagnação. Por meio disso não pode obter o valor ótimo.

SEÇÃO VI. Conclusão

A fim de superar as respectivas deficiências e ser forma de vantagens complementares, o uso de busca aleatória e rápida de algoritmos genéticos [1] , e mecanismo de feedback pros e alta eficiência característica do algoritmo de colônia de formigas, dois algoritmos serão amalgamados, e o núcleo idéias são as seguintes:

Em primeiro lugar, melhores soluções são produzidas e os feromônios são deixados para trás os caminhos de seus feromônios em outros caminhos não são alterados. E então, as formigas realizam operações de cruzamento e mutação de acordo com GA depois que cada formiga completa uma vez viajando à luz da ACA. Esta operação pode não melhorar as soluções de caminhos de formigas, portanto somente caminhos de formigas aprimorados são substituídos.

O algoritmo de mistura de ACA e GA é aplicado ao TSP e testado, o que se espera que obtenha um novo algoritmo heurístico melhor, esta é a direção futura para futuras pesquisas. Ao mesmo tempo, algumas melhorias no GA ou no ACA podem ser trazidas, como o sistema Max-MIN Ant System (MMAS) [7] , o sistema Best-Worst Ant (BWAS) e assim por diante.

AGRADECIMENTOS

Este trabalho foi financiado pela National Science Foundation Natural da China

(90612010,60703018), o Programa de Pesquisa e Desenvolvimento 973 Nacional Básico (2004CB318103, 2007CB310800), o Programa de Alta Tecnologia Nacional de Pesquisa e Desenvolvimento da China (2008AA01Z208) e do Projeto de Pesquisa de Departamento de Ciência e Tecnologia da Educação da Província de Jiangxi ([2007] 205).

Autores	▼
Figuras	▼
Referências	▼
Citações	▼
Palavras-chave	▼
Métricas	▼

IEEE Account

Profile Information

Purchase Details

Need Help?

Other

A not-for-profit organization, IEEE is the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity.
© Copyright 2019 IEEE - All rights reserved. Use of this web site signifies your agreement to the terms and conditions.

US & Canada: +1 800 678 4333

Worldwide: +1 732 981 0060

Conta IEEE

- » Alterar nome de usuário / senha
- » Atualizar endereço

Detalhes da compra

- » Opções de pagamento
- » Histórico de pedidos
- » Visualizar documentos comprados

Informação do Perfil

- » Preferências de Comunicações
- » Profissão e Educação
- » Interesses técnicos

Preciso de ajuda?

- » **EUA e Canadá:** +1 800 678 4333
- » **Em todo o mundo:** +1 732 981 0060
- » Contato e Suporte

Sobre o IEEE *Xplore* | Contate-Nos | Socorro | Acessibilidade | Termos de uso | Política de Não Discriminação | Mapa do Site | Privacidade e exclusão de cookies

Uma organização sem fins lucrativos, o IEEE é a maior organização profissional técnica do mundo dedicada ao avanço da tecnologia para o benefício da humanidade.
© Copyright 2019 IEEE - Todos os direitos reservados. O uso deste site significa sua concordância com os termos e condições.