

Algoritmo Genético com Interação Social Aplicado ao Problema de Cinemática Inversa

Janderson Tomé dos Santos Souza¹ e Otávio Noura Teixeira¹

¹ Faculdade de Engenharia de Computação
Campus Universitário de Tucuruí
Universidade Federal do Pará, PA, Brasil
{janderson.ts.souza, onoura}@gmail.com

Resumo. O uso de Algoritmos Genéticos (AG) ocorre em várias áreas do conhecimento e a robótica é uma delas. A fim de otimizar esses algoritmos, são realizados estudos de suas características e funcionalidades. Outras características podem ser inseridas, como é o caso da Interação Social nos Algoritmos Genéticos. A Interação Social é uma característica fenotípica complementar à característica genotípica do AG Clássico. Assim, este trabalho tem como objetivo realizar a execução do AG Clássico e do SIGA (Algoritmo Genético com Interação Social), baseado no Dilema do Prisioneiro, para comparar a eficiência dos dois algoritmos. Para isso, o AG Clássico e o SIGA foram executados com o Problema da Cinemática inversa, da Robótica, para identificar as características de cada um. O AG Clássico demonstrou maior velocidade de convergência e maior aproximação do resultado ótimo, enquanto o SIGA demonstrou maiores valores de variância e desvio padrão, indicando maior variabilidade da população.

Palavras-chave: Algoritmos Genéticos, Interação Social, Problema de Cinemática Inversa.

1 Introdução

Melanie Mitchell afirma que no contexto de Algoritmo Genético não há noção de fenótipo e que a Teoria Clássica dos Algoritmos Genéticos diz que cada indivíduo tem seu valor adaptativo fixo e imutável, mas na natureza não acontece assim [8]. Entretanto, Teixeira explica que a Teoria dos Jogos poderia ser usada como fundamento teórico para caracterizar uma nova abordagem fenotípica, porque formaliza com precisão situações de Conflitos de Interesses [14].

Vários trabalhos têm sido desenvolvidos a fim de aplicar conceitos da Teoria dos Jogos Evolucionária e Teoria dos Jogos a Algoritmos Genéticos através de aplicação de jogos Hawk-Dove, Dilema do prisioneiro e suas variações [1], [3], [6], [14], [15], e outros jogos. Essas aproximações foram estabelecidas baseadas em observações de como a natureza funciona, como colônias de formigas, enxame de abelhas ou mesmo na sociedade humana. Esses estudos possuem seu principal foco na regra de comportamento biológico simulado ou mecanismo biológico [18].

Para que isso seja possível, é necessário modificar a estrutura básica dos Algoritmos Genéticos, a fim de possibilitar que um indivíduo mude sua aptidão e, consequentemente, aumente suas chances de sobrevivência e de gerar descendência [14]. Então, os indivíduos são pareados e colocados em situação de Interação Social, permitindo que os comportamentos sejam expressados [1][14]. Esta fase é inserida antes da fase de reprodução e as interações são modeladas de acordo com a Teoria dos Jogos.

Os Algoritmos Evolucionários estão contidos no conjunto das técnicas existentes que são empregadas para resolver problemas práticos de robótica [9][12]. As redes neurais têm se tornado um dos tópicos mais populares em robótica por serem facilmente aplicáveis a diversos problemas dessa área [12]. Mesmo com a existência deste método e outros métodos de inteligência artificial, AGs têm recebido bastante atenção dos pesquisadores [9][19].

2 Estrutura do SIGA

O Algoritmo Genético com Interação Social (SIGA) tem como principal ideia a inserção de uma nova etapa nos Algoritmos Genéticos denominada Interação Social como característica fenotípica dos indivíduos através da formalização de situações de conflitos e interesses. Esta nova etapa ocorre antes da etapa de reprodução, permitindo que os indivíduos da população disputem por recursos disponíveis no meio ambiente [16].

A proposta de estrutura do SIGA é conforme a lógica do pseudocódigo a seguir:

```
// Algoritmo Genético com Interação Social
1. Início
2. Gerar População Inicial
3. Avaliar todos os indivíduos da população inicial
4. Repetir
    // Fase de Interação Social
    4.1. Repetir
        4.1.1. Selecionar dois indivíduos aleatoriamente
        4.1.2. Obter o comportamento de cada indivíduo
        4.1.3. Alterar o valor de fitness de acordo com a tabela de pagamentos do jogo
    Até que a Condição de Parada 2 seja satisfeita
    // Fase de Reprodução
    4.2. Repetir
        4.2.1. Selecionar dois indivíduos de acordo com o método de seleção estabelecido
        4.2.2. Operação de Cruzamento
        4.2.3. Operação de Mutação aplicada na geração
    Até que a Condição de Parada 3 seja satisfeita
    4.3. Substituir os indivíduos atuais da população pelos indivíduos mais novos gerados na fase de reprodução
```

- Até que a Condição de Parada 1 seja satisfeita
5. O melhor indivíduo da população é a solução do problema
 6. Fim

O comportamento utilizado na etapa de Interação Social possibilita que o indivíduo tenha sua aptidão alterada através da sua conduta, em situações de conflitos e interesses, que pode ser trair ou cooperar, de acordo com uma tabela de pagamentos baseada no Paradigma do Dilema do Prisioneiro [16]. Essa alteração da aptidão permite que indivíduos menos aptos se tornem mais aptos e que indivíduos mais aptos se tornem menos aptos, dependendo do que ocorra durante as disputas [13].

Algumas estratégias de comportamento podem ser adotadas, conforme mostrado no trabalho de Teixeira [16]. Desse modo, os indivíduos deixam de ter apenas a característica genotípica e passam a ter a característica fenotípica, com a inserção dos elementos “estratégia” e “aptidão” no cromossomo, conforme ocorrem as interações entre eles. O valor da aptidão do indivíduo é alterado conforme sua estratégia e pode ser acrescido ou subtraído do valor de fitness total [13].

Conforme pode ser observado na Tabela 1, o comportamento do indivíduo possui um valor de recompensa, que segue a ordem $T > R > P > S$ [18][19][20].

Tabela 1. Tabela de pagamentos dos indivíduos. O primeiro valor de cada parêntese pertence ao Indivíduo A e o segundo valor pertence ao Indivíduo B

		Indivíduo B	
		Coopera	Trai
Indivíduo A	Coopera	(R, R)	(S, T)
	Trai	(T, S)	(P, P)

A estrutura do SIGA apresentada, além de permitir as configurações básicas do AG clássico, como métodos de seleção, crossover e mutação, oferece outras possibilidades de combinação devido à etapa de Interação Social. Isso possibilita que o SIGA seja bom para algumas categorias de problemas e ruim para outras categorias [14].

Além de introduzir característica fenotípica no AG, Teixeira buscou tornar a Interação Social como principal motor de busca, ao invés de apenas auxiliar no processo de evolução [13]. Desse modo, o algoritmo possui um genótipo, que será denominado *Alpha*, e um fenótipo (característica de Interação Social). Para efeitos deste estudo, serão considerados os valores 0 e 1 para *Alpha*, onde $\alpha = 1$ indica presença de genótipo e $\alpha = 0$ indica apenas interação social.

3 Problema da Cinemática Inversa

Manipuladores robóticos são tipicamente construídos com junções ligadas a outras junções através de elos [12]. Para movimentar todos esses elos e junções, é comum utilizar a Cinemática Inversa [11].

Por ser utilizado para controlar a postura de um corpo articulado, a solução do problema da Cinemática Inversa tornou-se fundamental para a robótica e vários são métodos propostos [9][12][19]. A maioria desses métodos inclui abordagens geométrica, iterativa e algébrica. Entretanto, esses métodos são relatados como inadequados para robôs redundantes[9][11].

Infelizmente, não foi encontrada uma solução analítica para o problema da Cinemática Inversa de uma forma geral [12], possibilitando o crescimento dos Algoritmos Evolucionários e, em caso mais particular, dos Algoritmos Genéticos.

Considerando um robô, ou braço robótico com M graus de liberdade e N pontos nas coordenadas espaciais e dado um caminho desejado (P_{dc}), o objetivo é gerar um caminho (P_{gc}) em que o desvio padrão cumulativo entre P_{dc} e P_{gc} seja o mínimo. Nk é o número de conectores i (pontos) que especificam a precisão em relação ao caminho desejado. Tanto o P_{dc} quanto o P_{gc} são matrizes de dimensões $N \times Nk$. O cálculo do P_{gc} é calculado de uma matriz P_0 formada por valores que foram gerados através do Algoritmo Genético de tamanho $M \times Nk$. Para calcular o desvio padrão $E(i)$ de qualquer ponto i , aplica-se a Equação (1):

$$E(i) = \sum_{k=1}^N |Pdc(k, i) - Pgc(k, i)| \quad (1)$$

O desvio padrão cumulativo E (função fitness) entre o P_{dc} e o P_{gc} é dado pela Equação (2):

$$E = \sum_{i=1}^{Nk} \sum_{k=1}^N |Pdc(k, i) - Pgc(k, i)| \quad (2)$$

Considerando que o P_{dc} é um caminho definido, para este problema foi considerado um caminho desejado com 20 pontos/conectores ($Nk = 20$), mostrados na Equação (3), enquanto a Fig 1 demonstra de forma gráfica o formato do caminho desejado (P_{dc}):

$$\begin{aligned} Pdc(1, i) &= Xdc(i) = x_{initial} + \frac{x_{final} + x_{initial}}{Nk - 1} \times (i - 1) \\ Pdc(2, i) &= Ydc(i) = 0.25 \\ x_{initial} &= 0; x_{final} = 0.25 \end{aligned} \quad (3)$$

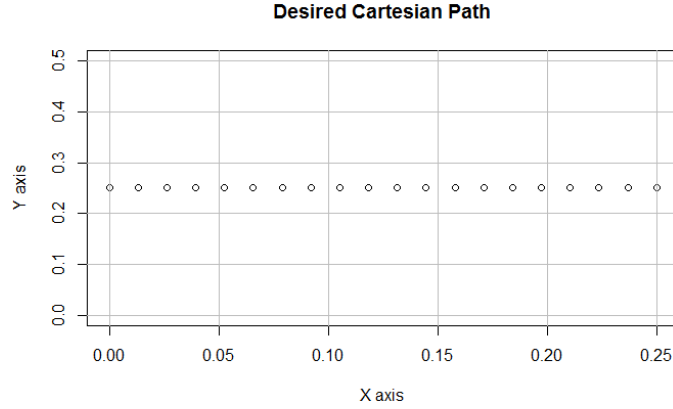


Fig. 1. Pontos desejados para o P_{dc}

O modelo cinemático do manipulador que gera o P_{gc} é dado pela Equação (4):

$$\begin{aligned} Xgc(i) &= \sum_{j=1}^M \left(L_j \times \cos \left[\sum_{k=1}^j \theta_{k,i} \right] \right) \\ Ygc(i) &= \sum_{j=1}^M \left(L_j \times \sin \left[\sum_{k=1}^j \theta_{k,i} \right] \right) \end{aligned} \quad (4)$$

Os parâmetros do manipulador ficaram definidos como:

- $L1 = L2 = L3 = 0.5$
- $N = 2$
- $M = 3$
- $0 \leq \theta(h) \leq 180$ para $h = 1, 2, 3$
- $Nk = 20$
- $1 \leq i \leq Nk$

4 Resultados

Para realização dos testes, foram seleccionadas 5 configurações distintas para o algoritmo genético: AG clássico e SIGA com duas configurações da tabela de pagamento sendo uma com $Alpha = 1$ e $Alpha = 0$. Os valores da tabela de pagamento (ver Tabela 1) ficaram definidos como $T=5$, $R=3$, $P=1$ e $S=0$ para a primeira configuração e $T=30$, $R=25$, $P=15$ e $S=10$ para a segunda configuração. Os valores da tabela de pagamento [17] e de $Alpha$ [13] foram baseados em trabalhos de Teixeira. Para o AG clássico e para cada configuração do SIGA, foram realizadas 50 execuções, totalizando 250 execuções. Todas as execuções foram realizadas com a mesma população

inicial, de tamanho 100, gerada aleatoriamente. Foi definido o valor de 85% de taxa de cruzamento e 5% de taxa de mutação para os AGs.

Cada execução ocorreu por 500 gerações e, a partir da obtenção dos resultados, foram extraídas as médias do melhor e pior indivíduo por geração das 50 execuções de cada algoritmo. Os valores médios dos melhores e piores indivíduos foram plotados, inserindo o valor médio das médias dos indivíduos por geração e são mostrados nas figuras Fig. 2, Fig. 3 e Fig. 4.

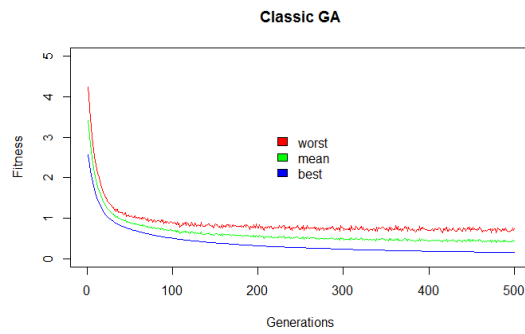


Fig. 2. Média dos melhores e piores indivíduos, por geração, das 50 execuções do AG Clássico, com a média desses valores.

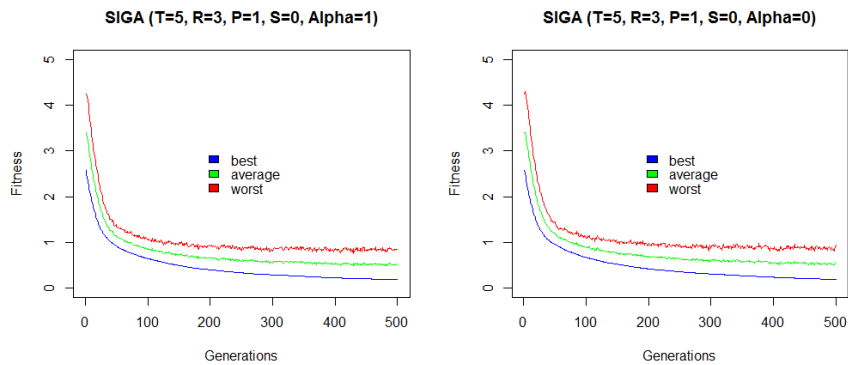


Fig. 3. Média dos melhores e piores indivíduos, por geração, das 50 execuções do SIGA, de acordo com a primeira configuração da tabela de pagamentos, com a média desses valores. $\alpha = 1$ à esquerda e $\alpha = 0$ à direita.

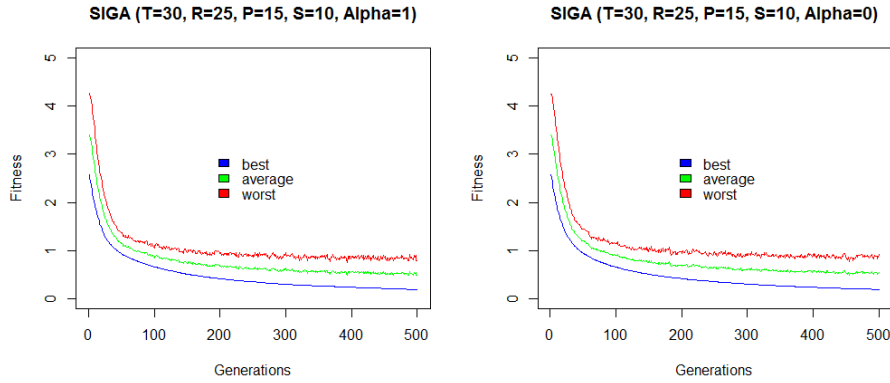


Fig. 4. Média dos melhores e piores indivíduos, por geração, das 50 execuções do SIGA, de acordo com a segunda configuração da tabela de pagamentos, com a média desses valores. $Alpha = 1$ à esquerda e $Alpha = 0$ à direita.

A visualização dos gráficos demonstra que o AG Clássico tem curva mais acentuada se comparado com as execuções do SIGA. Isso indica que a convergência desse algoritmo é mais rápida que o SIGA, que denota curvas mais suaves. Entretanto, pode ser difícil perceber o quanto essas curvas divergem.

Conforme mostrado na Fig. 5, os valores médios dos melhores indivíduos, quando traçados em gráfico juntos, denotam diferenças entre os resultados obtidos com os algoritmos. É notável que os valores iniciais e finais são próximos, mas a velocidade de convergência é diferente entre o AG Clássico e o SIGA. Nota-se também que são semelhantes as curvas do SIGA, independente dos valores das tabelas de pagamento utilizadas e dos valores de $Alpha$.

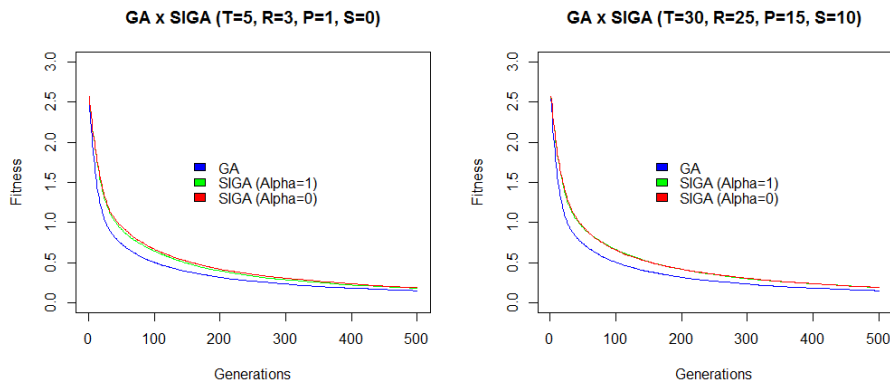


Fig. 5. Valores médios dos melhores indivíduos do AG Clássico confrontados com os do SIGA

Não obstante, os valores de desvio padrão e variância confirmam que a variabilidade de indivíduos é maior em todas as execuções do SIGA, apesar de os valores médios delas não terem se mostrado melhor do que o AG Clássico (ver Tabela 2). Entretanto,

as curvas dos gráficos convergem para valores próximos em todas as execuções de todos os algoritmos testados, demonstrando a eficiência.

Os valores de desvio padrão do SIGA indicam que a população possui maior variabilidade que as populações do AG Clássico. A variância, por sua vez, constata que as populações do SIGA alcançam valores mais distantes da média e que as populações do AG Clássico. Manter a variabilidade nas populações é importante, pois, possibilita a estagnação em ótimos locais e amplia o alcance, permitindo o alcance do ótimo global.

Tabela 2. Valores de desvio padrão e variância dos algoritmos

Algoritmo	Desvio padrão	Variância
AG Clássico	0.1313562	0.01725444
SIGA (T = 5, R = 3, P = 1, S = 0, Alpha = 1)	0.197874	0.03915412
SIGA (T = 5, R = 3, P = 1, S = 0, Alpha = 0)	0.2086938	0.04355311
SIGA (T = 30, R = 25, P = 15, S = 10, Alpha = 1)	0.1967533	0.03871187
SIGA (T = 30, R = 25, P = 15, S = 10, Alpha = 1)	0.1907599	0.03638933

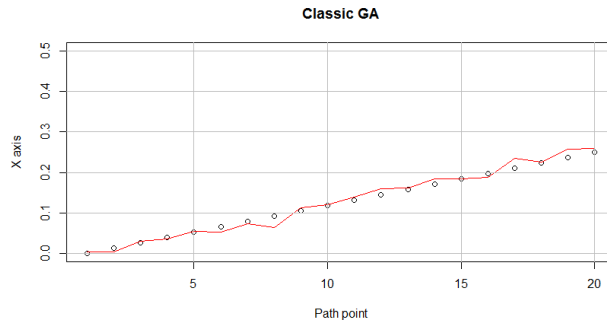


Fig. 6. Valores de Xgc encontrados pelo AG Clássico

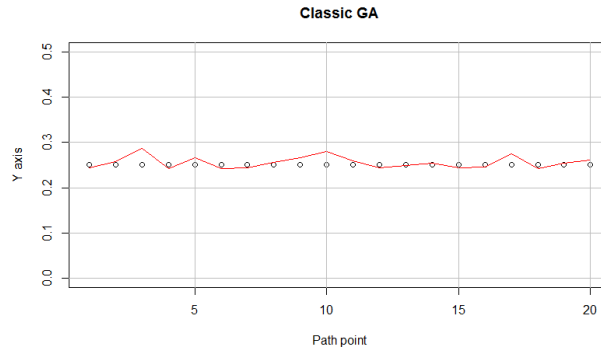


Fig. 7. Valores de Ygc encontrados pelo AG Clássico

As figuras Fig. 6, Fig. 7, Fig. 8 e Fig. 9 representam os melhores valores encontrados de cada algoritmo, enquanto a Tabela 3 ilustra o melhor indivíduo encontrado dentre todos os testes executados.

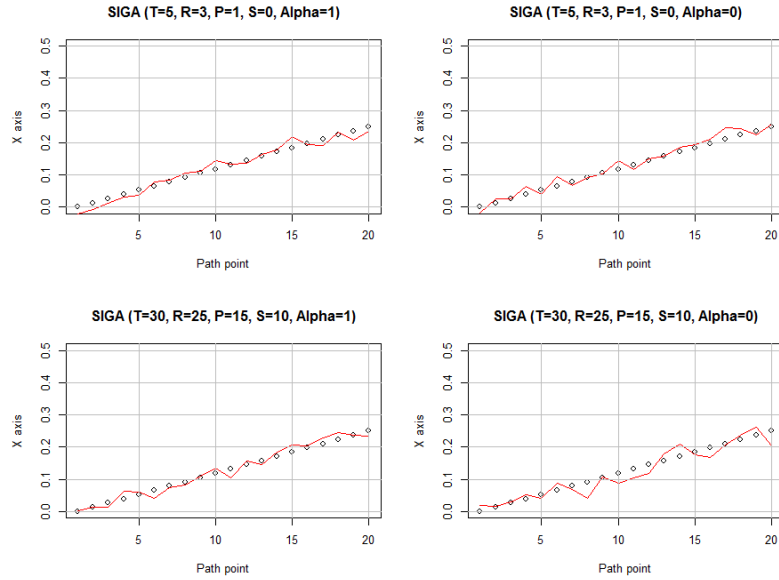


Fig. 8. Valores de X_{gc} encontrados pelo SIGA

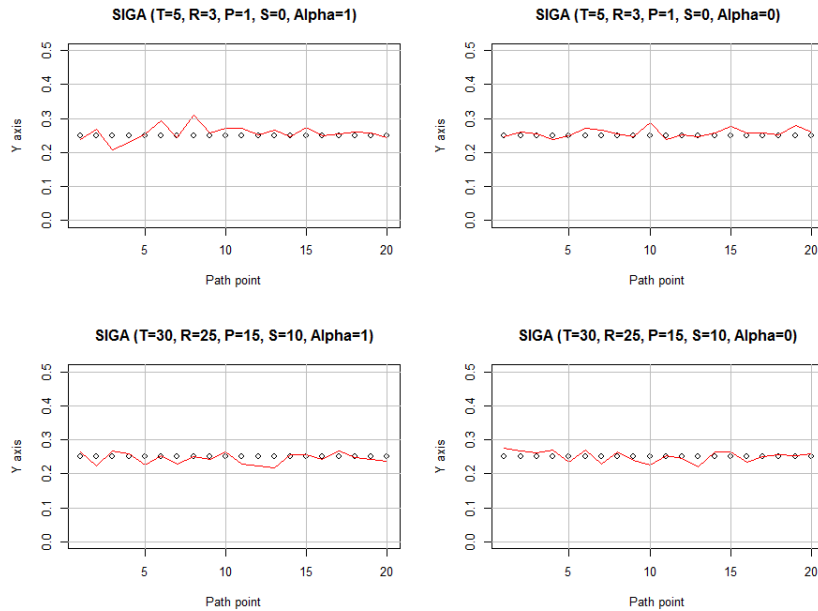


Fig. 9. Valores de Y_{gc} encontrados pelo SIGA

Tabela 3. Ângulos do melhor indivíduo encontrado

Nk (Path point)	P θ (M = 3)			Pgc (N = 2)	
	θ_1	θ_2	θ_3	Xgc	Ygc
1	170.1350246	149.0426475	117.1610151	0.0038541027293	0.244660070244
2	113.4360641	128.4420598	146.8173915	0.0040548079952	0.25785617017
3	131.5940627	146.6414391	136.3469795	0.0294619272158	0.28656492931
4	92.26452229	115.1721421	149.4685873	0.035753913574	0.24223197468
5	71.73026406	95.46601345	153.0395078	0.0535195641068	0.265786661994
6	11.22181161	87.99238168	124.9792563	0.051882559618	0.24231063196
7	91.49129409	122.9449487	148.5318577	0.073938752434	0.242976214880
8	102.7087587	130.8438302	146.4020914	0.062941405267	0.25618760279
9	177.7458277	155.4225519	97.44838392	0.112497914205	0.265641154180
10	116.6555687	150.170346	134.9368544	0.12095648494	0.280650781086
11	113.8692075	149.256625	133.56221	0.138780469788	0.259558978825
12	170.1763649	155.681955	95.70989128	0.15921342827	0.244379652342
13	50.72518717	92.80627493	155.9616079	0.160587775502	0.249042302580
14	46.55455069	88.89851581	157.8106636	0.184975799252	0.25441305595
15	113.7286928	153.3635115	127.7757524	0.183668312633	0.244217915156
16	106.4048387	151.8241808	132.4202514	0.186940624869	0.245050351463
17	15.14742739	61.42205955	146.5768742	0.233956712973	0.27504578049
18	36.2807221	80.98620571	158.5407254	0.224590030770	0.242878302279
19	48.40761306	107.4943255	163.7584393	0.256599860405	0.254434419397
20	168.6387469	163.5370447	80.02218241	0.25845710760	0.260183683443

5 Conclusões

Os Algoritmos Genéticos têm se mostrado eficientes para solucionar diversos problemas de diversas áreas do conhecimento, inclusive o de Cinemática Inversa da robótica. Entretanto, esses algoritmos possuem a tendência de estagnar em ótimos locais, o que pode distanciar as soluções aproximativas de soluções ótimas.

O controle da variabilidade populacional pode proporcionar maior exploração no espaço de busca, aumentando as chances de encontrar o ótimo global. Os trabalhos realizados com o SIGA têm demonstrado que é possível realizar esta manutenção na população sem perda de desempenho no algoritmo, encontrando valores bem próximos ou tão bons quanto o AG Clássico.

Nos testes realizados neste trabalho, nenhuma das configurações do SIGA encontrou melhores indivíduos que o AG Clássico. Entretanto, é considerável a quantidade de configurações que podem ser utilizadas para ajustar o AG, permitindo que sejam encontrados valores mais satisfatórios.

Todos os algoritmos testados possuem características em comum, tais como taxa de cruzamento, taxa de mutação e método de seleção. Por isso, vale ressaltar que o SIGA possui características próprias, pois se trata do caráter fenotípico, o que inclui uma estratégia e um valor de aptidão dos indivíduos.

Para a realização dos testes foram utilizados valores de referência empregados em outros trabalhos. Todavia, modelar os valores T, R, P e S da tabela de pagamento podem contribuir para que valores mais próximos de um ótimo global sejam encon-

trados. Além disso, o valor de *Alpha* pode ser modificado para ser tratado como valor ponderado, isto é, dar maior significância às características genotípicas ou maior significância às características fenotípicas.

Além de variar os valores para encontrar melhores soluções, a tabela de pagamentos e o *Alpha* podem ser associados aos valores intrínsecos do AG (taxa de mutação, taxa de cruzamento, método de seleção) para viabilizar controle sobre a variabilidade dos indivíduos.

Em relação ao problema de Cinemática Inversa, pode-se observar que o espaço de busca é considerável, pois são 60 valores numéricos pertencentes aos conjuntos dos números.

Portanto, em trabalhos futuros, podem ser realizados novos testes no mesmo Problema da Cinemática Inversa, alterando os valores característicos de Algoritmos Genéticos e os valores inerentes à Interação Social empregada neles.

É viável, também, fazer a normalização dos valores que compõem a tabela de pagamento. Isso é promissor, pois, viabiliza a utilização do SIGA com maior eficiência, culminando no alcance de melhores resultados do algoritmo.

Referências

1. Beltra, R. L.; Ochoa, G.; Aickelin, U. Cheating for problem solving: a genetic algorithm with social interactions. In: Proceedings of the 10th annual Conference on Genetic and Evolutionary Computation, 811-818, ACM, Montreal, Quebec, Canada, 2009.
2. Borges, P. S. S.; Pacheco, R. C. S.; Barcia, R. M.; Khator, S. K. A fuzzy approach to the prisoner's dilemma. Elsevier: Biosystems, Volume 41, Issue 2, January 1997, pages 127-137.
3. Brito, F. H.; Teixeira, O. N.; Oliveira, R. C. L. A Introdução da Interação Fenotípica em Algoritmos Genéticos Através dos Jogos Evolucionários e da Codificação e Transmissão Genética do Comportamento. In: Proceedings of VII SBAI / II IEEE-LARS - Simpósio Brasileiro de Automação Inteligente / Latin American IEEE Robotics Symposium, 2005, São Luís.
4. Goldberg, D. Genetic Algorithms in search, optimization and machine learning. Addison Wesley Longman, Inc., 1989.
5. Kaveh, A. and Mahdavi, V.R. Colliding bodies optimization: A novel meta-heuristic method. Computers and Structures, 139, pp. 18–27, 2014.
6. Lehrer, C. Operador de Seleção para Algoritmos Genéticos Baseado no Jogo Hawk-Dove. 2000. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Santa Catarina, Florianópolis.
7. Luce, D. R.; Raiffa, H. Games and decision: introduction and critical survey. New York: Dover, 1957.
8. Mitchell, M. An introduction to genetic algorithms. MIT Press, 1999.
9. Momani, Shaher & Abo-Hammour, Zaer & Alsmadi, Othman. Solution of Inverse Kinematics Problem using Genetic Algorithms. Applied Mathematics & Information Sciences. (2015) 10. 1-9.
10. Poundstone, W. Prisoner's Dilemma: John Von Neumann, Game Theory, and the Puzzle of the Bomb. Anchor Books, 1992.

11. Samuel R. Buss, Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods, "<http://www.math.ucsd.edu/sbuss/ResearchWeb/ikmethods/iksurvey.pdf>", 2009.
12. Rait Koker, Tark akar, Yavuz Sari, A neural-network committee machine approach to the inverse kinematics problem solution of robotic manipulators. *Engineering with Computers*, DOI:10.1007/s00366-013-0313-2, 2013.
13. Teixeira, O. N. Algoritmo Genético com Interação Social Nebulosa. Tese de Doutorado (Programa de Pós-graduação em Engenharia Elétrica – Computação Aplicada) – Universidade Federal do Pará, Belém, 2012.
14. Teixeira, O. N.; Brito, F. H.; Teixeira, A. N.; Oliveira, R. C. L. Game Theory as a new paradigm for phenotype characterization of Genetic Algorithms. In: *Proceedings of the 8th annual Conference on Genetic and Evolutionary Computation*, 1431-1432, ACM, Seattle, Washington, USA, 2006.
15. Teixeira, O. N.; Brito, F. H.; Teixeira, A. N.; Oliveira, R. C. L. Genetic Algorithms with Social Interaction Phase as Phenotype Characterization. In: *Proceedings of Advance in Natural Computation and Data Mining*. Xi'an : Xidian University Press, 2006.
16. Teixeira, O. N. Proposta de um novo algoritmo genético baseado na teoria dos jogos. *Dissertação de Mestrado* (Programa de Pós-graduação em Engenharia Elétrica – Computação Aplicada) – Universidade Federal do Pará, Belém, 2005.
17. Teixeira, O. N., Yasojima, C. T. K, Teixeira, A. N., Lobato, W. A. L, Brito, F. H., Oliveira, R. C. L.: A New Hybrid Nature-Inspired Metaheuristic for Problem Solving Based on the Social Interaction Genetic Algorithm Employing Fuzzy Systems. In: *10th International Conference on Hybrid Intelligent Systems, HIS 2010*, pp. 31-36. IEEE, Atlanta (2010).
18. Yang G. Game Theory-Inspired Evolutionary Algorithm for Global Optimization. *Algorithms*. 10(4), 111 (2017).
19. Yasojima, E., De Oliveira, R., Teixeira, O., & Pereira, R. (n.d.). CAM-ADX: A New Genetic Algorithm with Increased Intensification and Diversification for Design Optimization Problems with Real Variables. *Robotica*, 1-46. doi:10.1017/S026357471900016X
20. Zhanga, G., Chenga, J., Gheorghe. M., Meng, Q. A hybrid approach based on differential evolution and tissue membrane systems for solving constrained manufacturing parameter optimization problems. *Applied Soft Computing*, 13, pp. 1528–1542, 2013.