

OPTIMAL SOLUTION TO BOX PUSHING PROBLEM BY USING BBO – NSGAII

By

¹Sudeshna Mukharjee and ²Sudipta Ghosh

¹Cognizant Technology Solution Pvt. Ltd., Salt lake, Sector V, Kolkata, West Bengal,
India

² Department of Electronics and Communication Engineering, Camellia School of
Engineering and Technology, Barasat, Kolkata, West Bengal, India

Abstract:

In this paper we have developed a new technique to determine optimal solution to box pushing problem by two robots . Non-Dominated sorting genetic algorithm and Biogeography-based optimization algorithm are combined to obtain optimal solution. A modified algorithm is developed to obtain better energy and time optimization to the box pushing problem.

Keywords and phrases : box pushing, robots, Non-Dominated sorting genetic algorithm, Biogeography-based algorithm

বিমূর্ত সার (Bengali version of the Abstract)

রোবটদ্বয় (two robots) কৃত বক্স পুশিং (box pushing) সমস্যায় প্রকৃষ্টতম সমাধান নির্ণয়ের জন্য এই পত্রে আমরা একটি নূতন কৃৎকৌশলের উদ্ভব করেছি । ইহার প্রকৃষ্ট সমাধান নির্ণয়ের জন্য অ-নিয়ন্ত্রক বর্গের বংশানুক্রম এ্যালগোরিদম এবং জীব - ভৌগোলিক নির্ভর প্রকৃষ্টতম এ্যালগোরিদমের সমন্বয় করেছি । বক্স পুশিং সমস্যার অধিকতর শক্তি এবং সময়ের প্রকৃষ্টতম মান নির্ণয়ের জন্য ঈষৎ পরিবর্তন কৃত এ্যালগোরিদমের উদ্ভব করা হয়েছে ।

1. Introduction:

The use of perceptual cues in multi robot box pushing was investigated by Kube, C.R and Zhang, H (9). Brooks, R.A (8) presented a robust layered control for a mobile robot. Genetic algorithm for multi-object optimal solution formulated and generalized by Fonseca, C.M and Flaming, P.J.(11). This work was further extended to Multi-objective programming using uniform design and genetic algorithm by Leung, Y.W. and Wang, Y.P (10). Langle, T and Worn, H.(6) investigated the problem of Human-robot co-operation using multi-agent system. Innocenti, B., Lopez, B and Salvi, J.(7) presented a multi-agent architecture with co-operative fuzzy control for a mobile robot. Deb, K.(4) developed a fast and Elitist Multi-objective Genetic algorithm: NSGAII. Department of Electrical Engineering, Shaoxing University, China(3) had presented an analysis of the equilibrium of Migration Models for Biogeography-based optimization. Biogeography-based optimization was also deduce by Simon, D.(2). Rotation and Translation selective pareto optimal solution to the box-pushing problem by mobile robot using NSGA-II was developed by Chakraborty, J., Konar, A., Nagar, A. and Das, S.(1).

However, optimal solution to box pushing problem by two robot by using BBO – NSGA-II algorithm has not been yet investigated by any researcher in a similar approach. The objective of the present paper is to find optimal solution to box pushing problem by developing combination of two algorithms: Non-Dominated sorting Genetic algorithm and Biogeography-based algorithm.

1. Statement of the problem and its solution.

Let us consider two robots participating in Solid lines represent initial and final positions and dashed lines represent other positions between initial and final. The stepwise performance is as follows-

Step1: A translation operation is carried out where one robot is pushing the box and the other is pulling it in order to move it along its width.

Step2: A rotational operation is carried out where two robots are applying equal and opposite forces to bring about rotation of the box about its centre.

M EXPRESSIONS FOR BOX PUSHING PROBLEM:

Let $G(X_g, Y_g)$ be the centre of gravity and $A(X_a, Y_a)$, $B(X_b, Y_b)$, $C(X_c, Y_c)$ and $D(X_d, Y_d)$ be the 4 vertices of the box.

1. for translatory motion the following equations are derived

$$X_g' = X_g + D \cos \alpha \quad Y_g' = Y_g + D \sin \alpha \quad (1)$$

$$X_m' = X_m + D \cos \alpha \quad Y_m' = Y_m + D \sin \alpha \quad (2)$$

for all $m \in P = \{a, b, c, d\}$

$$\alpha = \theta + 180^\circ$$

Box pushing as shown in fig-1, fig-2, and fig-3.

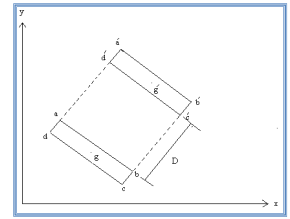


Fig.-1

2. For rotational motion the following equations are derived:

$$X_m' = X_g(1 - \cos \theta) + X_m \cos \theta - (Y_g - Y_m) \sin \theta \quad (3)$$

$$Y_m' = X_g(1 - \cos \theta) + Y_m \cos \theta - (Y_g - Y_m) \sin \theta \quad (4)$$

for all $m \in P = \{a, b, c, d\}$

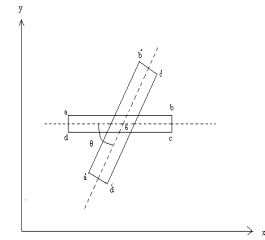


Fig.-2

3. For combined translator and rotational motion :

Co-ordinates after rotational movement we have,

$$X_m' = X_g(1 - \cos \theta) + X_m \cos \theta - (Y_g - Y_m) \sin \theta \quad (5)$$

$$Y_m' = X_g(1 - \cos \theta) + Y_m \cos \theta - (Y_g - Y_m) \sin \theta \quad (6)$$

for all $m \in P = \{a, b, c, d\}$

Co-ordinates after translatory movement we have

$$X_g' = X_g + S \cos \alpha \quad Y_g' = Y_g + S \sin \alpha \quad (7)$$

$$X_m'' = X_m' + S \cos \alpha \quad Y_m'' = Y_m' + S \sin \alpha \quad (8)$$

for all $m \in P = \{a, b, c, d\}$

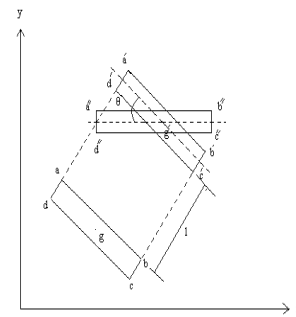


Fig.-3

4.. Algorithms:

In this paper, two algorithms are combined to obtain optimization:

- A. Non Dominated Sorting Genetic Algorithm.[3]
- B. Biogeography-Based Optimization Algorithm.

A. Non Dominated Sorting Genetic Algorithm:

NSGA-II is a non-dominated sorting based Multi Objective Evolutionary Algorithm(MOEA)[3] which overcomes 3 problems of common MOEA-

1. Computational complexity reduces from $O(MN^3)$ to $O(MN^2)$ where M is the number of objective and N is the population size.
2. Non elitism approach.
3. Need for a shared parameter.

Here a selection operator is introduced which selects the best solution with respect to fitness among the N solutions i.e. the combination of parent and child solutions. This process also introduces three new techniques-

- a. Fast non-dominated sorting procedure
- b. Fast crowding distance estimation procedure
- c. A simple crowded comparison operator

In order to determine the solution of crowding distance have to add the differences between the value of the solutions of two adjacent objective functions.

Solutions. Let us consider solutions for two objective functions be f_1 and f_2 . Now for the first and second objective functions, the value of the distance value for the i-th solution will be-

$$CR 1_{distance} [i] = f_1 [i + 1] - f_1 [i - 1]$$

$$CR 2_{distance}[i] = f_2 [i - 1] - f_2 [i + 1] \text{ respectively.}$$

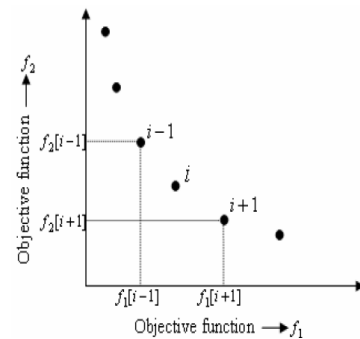
Calculation of crowding distance

Now, for the solution of crowding distance for the solution we have to obtain the sum of the distance values for each objective function.

Process:

NSGA-II procedure can be summarized as follows-

- 1) At first a random parent population P_0 is initialized.
- 2) Sorting of population is done based on non-domination.



- 3) Each solution is given a fitness or rank based on their nondomination level starting from 1 as the best level. In this way we achieve minimization of fitness.
- 4) A child population Q_0 of size N is created using binary tournament selection, recombination and mutation.
- 5) Due to the introduction of elitism a different procedure is followed after the initial generation-
 - A. Let in t th generation, parent population be denoted as P_t (size N) and child population as Q_t (size N).A combined population $R_t = P_t \cup Q_t$ of size $2N$ is formed.
 - B. R_t is sorted according to non-domination. Due to the presence of previous and current population members in R_t elitism is ensured.
 - C. Solutions belonging to best non-domination set (say F_1) are of best solutions in the combination population.
 - D. If size of $F_1 < N$, then all members of F_1 are selected for the new population P_{t+1} .
 - E. The remaining members of P_{t+1} are selected from the consecutive ranked non-dominated sets. The solutions from sets F_2, F_3 and so on are selected as shown in the fig.4
 - F. To select exactly N population members we sort the last set F_l using crowded comparison operator α_n in descending order and select best solutions needed to fill the population slots. α_n requires both rank and crowded distance of each solution in the population. These quantities are calculated while forming population P_{t+1} .
 - G. The new population P_{t+1} of size N is now used to create a new population Q_{t+1} of same size by the process of selection, crossover and mutation.

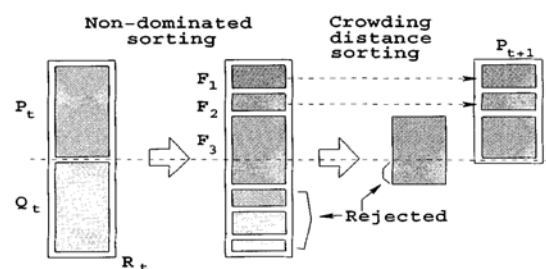


Fig.4

A. Biogeography-based optimization:

BBO algorithm deals with the migration of species from one 'island' to another. Here the word 'island' means a geographically isolated habitat. Different species migrate between islands. Some of the islands are more suitable to live for some species than

others depending on its topography ,climate etc.This suitability is expressed in the form of a variable called suitability index variable(SIV).Habitats with high suitability is said to have high habitat suitability index(HHSI) and those with low suitability is said to have low habitat suitability index(LHSI).High HSI indicates high emigration rate[2](more species leave the habitat) and low immigration rate[2](fewer species enter the habitat).It happens because in highly suited habitats density of species are near to the saturation and they tend to migrate to other habitats with more opportunities. Low HSI indicates high immigration rate[2](more species enter the habitat) and low emigration rate(fewer species leave the habitat).

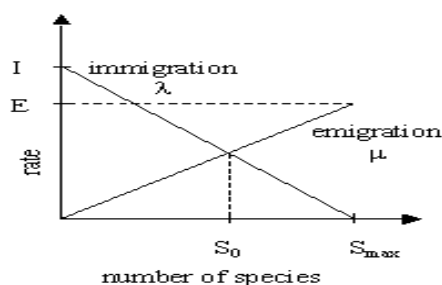


Fig.1

Now this fig.1 shows the migration curve, where λ is the immigration rate and μ is the emigration rate. Both λ and μ are functions of the number of species in a habitat.

From the immigration curve it is clear that immigration rate is maximum (I) when there is no species in a habitat. But when the habitat becomes more populated with species, possibility of successful survival of immigration to the habitat decreases i.e. λ decreases. When the number of species becomes S_{\max} and the habitat is saturated with species the immigration rate λ becomes zero.

Again from the emigration curve it is clear that emigration rate is zero when there is no species in a habitat. But when the habitat becomes more populated with species, possibility of leaving the habitat increase i.e. μ increases. When the number of species becomes S_{\max} and the habitat is saturated with species the emigration rate becomes maximum(E).

In the curve the intersection point S_0 shows the equilibrium state that means both the immigration rate and emigration rate are equal at this point. Now migration can occur from S_0 in two ways –positive excursion(immigration) and negative excursion(emigration).Positive excursion occurs due to a sudden arrival of a huge number of species from neighboring habitat by flotsam, flying, swimming or due to a sudden increase in speciation. Negative excursion occurs due to natural calamity, disease etc.

Now suppose probability to contain S species in a habitat is P_s .From time t to $(t+\Delta t)$ the probability P_s changes as follows:

$$P_s(t+\Delta t) = P_s(t)(1-\lambda_s\Delta t - \mu_s\Delta t) + P_{s-1}\lambda_{s-1}\Delta t + P_{s+1}\mu_{s+1}\Delta t \quad (9)[2]$$

Where λ_s and μ_s are the immigration and emigration rates for S species in the habitat. This equation is true if one of the following conditions must satisfy:

- 1) At time t there were S number of species in the habitat and no species arrived in the habitat or no species left the habitat between the time t and $(t+\Delta t)$;
- 2) At time t there were $(S-1)$ number of species, and one species arrived in the habitat.
- 3) At time t there were $(S+1)$ number of species, and one species left from the habitat.

Here Δt is so small that probability of migration of more than one species can be neglected.

Application of BBO to optimization problems :

I. Migration[2]:

In a problem , the population of possible solutions can be considered as vectors of integers. Each integer in the solution vector represents a SIV. Good solutions are considered to be habitats with high HSI and poor solutions are considered to be habitats with low HSI.HSI in BBO is equivalent to “fitness” function as in other population based genetic algorithms. For each solution(habitat) an identical species curve is assumed i.e. $E=I$ as shown in fig.2.

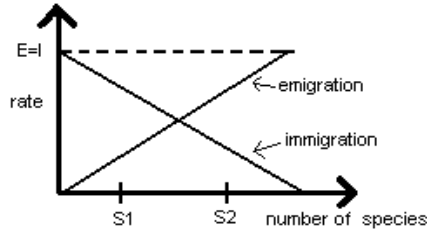


Fig.2

S1=habitat with few species(high HSI) and S2=habitat with many species(low HSI)

From here we can conclude immigration rate(λ) of S1 is greater than immigration rate of S2 and emigration rate (μ) of S1 is lower than emigration rate of S2.

II. Mutation[2]:

HSI of a habitat is prone to changes of environment. The species count suffers a deviation from its equivalent value under unusual circumstances (i.e.arrival of large number of migrants from other habitats).

Probabilities of each species is expressed by the differential equation-

$$\begin{aligned} \dot{P}_s &= -(\lambda_s + \mu_s)P_s + \mu_{s+1}P_{s+1}, & S=0 \\ &= -(\lambda_s + \mu_s)P_s + \mu_{s-1}P_{s-1} + \mu_{s+1}P_{s+1}, & 1 \leq S \leq S_{\max}-1 \\ &= -(\lambda_s + \mu_s)P_s + \mu_{s-1}P_{s-1}, & S=S_{\max} \end{aligned} \quad (10)[2]$$

From the fig.1 it can be concluded that species having low count and high count have relatively low probabilities compared to species with medium count. The reason behind this is that species with medium count are near to the equilibrium point(S_0). Thus it can be inferred that medium HSI solutions are relatively more probable than both high HSI and low HSI solutions.

Solutions with low probability is more prone to mutate to some other solutions than solutions with high probability. The implementation can be expressed as

$$m(S) = m_{\max}(1 - P_s)/P_{\max} \quad (11)[2]$$

Here $m(S)$ is the mutation rate which is inversely proportional to solution probability and m_{\max} is an user defined parameter.

Mutation is applied both on low HSI and high HSI solutions but not on medium HSI solutions. This modification helps the low HSI solutions to improve and decrease the dominance of high HSI solutions(which are also improving).If mutation destroys HSI of the best solution we use elitism approach to restore it.

III.Modifications:

To obtain better energy and time optimization of the box pushing problem we combine the NSGA-II and BBO algorithms. The modified algorithm is as follows-

1. At first a random population(habitat) P_0 with N number of species(solution) is initialized. Each solution is a D -dimensional vector. Box parameters are initialized as $x_{\text{curnt}} = x_c$ and $y_{\text{curnt}} = y_c$
2. Calculate HSI of each solution of population P_0 , then sorting of solutions in population P_0 is done based on non-domination using those HSI values.
3. Calculate immigration rate λ and emigration rate μ for each solution in population P_0 .
4. Child solutions of size D is created from randomly selected parent solution exchanging the SIV between two chromosomes (solution) keeping other fields unchanged, called migration. Calculate HSI of each child. Then non-dominated sorting is performed on child population based on HSI.
5. After migration, mutation takes place. Calculate the probability of population using equation (9). Calculate mutation rate for each species in population. Mutation chooses equiprobable solution and increases diversity among population. If mutation destroys HSI of the best solution we use elitism approach to restore it.
6. Compare HSI of parent & child solution.
i.e. $\| \text{HSI}_{\text{PARENT}} - \text{HSI}_{\text{CHILD}} \| < \epsilon$ or not.
7. If $\| \text{HSI}_{\text{PARENT}} - \text{HSI}_{\text{CHILD}} \| < \epsilon$, then parent population P_0 is selected.
Otherwise check for three conditions-

a) If $\text{HSI}_{\text{PARENT}} > \text{HSI}_{\text{CHILD}}$, then
select parent population P_0 .

b) If $\text{HSI}_{\text{PARENT}} < \text{HSI}_{\text{CHILD}}$, then

select childpopulation Q_0 .

c) If $HSI_{PARENT} = HSI_{CHILD}$, then
we have to select the best

solution using crowding

distance[1].

8. Hence, x_{curnt} and y_{curnt} are updated using equations 1-8.

9. Above mentioned steps (step 1 - 9) are repeated until the termination criteria i.e. $(|x_{cg} - x_{curnt}| \text{ and } |y_{cg} - y_{curnt}|) < \beta$ is met. Here β is an arbitrarily small number and (x_{cg}, y_{cg}) is the co-ordinate of the centre of gravity of the box.

IV. Pseudo code:

Input: Initial centre of gravity(cg) of the box(x_c, y_c), final centre of gravity(cg) of the box(x_{cg}, y_{cg}).

Output: Forces(F_A, F_B) applied by the two robots, Total energy consumed , Total time taken.

Begin:

Initialize parameters for the box:

$x_{curnt} = x_c$;

$y_{curnt} = y_c$;

Repeat

Call **BBO-NSGAI**(x_{curnt}, y_{curnt})

Update(x_{curnt}, y_{curnt}) in each step using eqns.1 to 8.

Until $(|x_{cg} - x_{curnt}| \text{ and } |y_{cg} - y_{curnt}|) < \beta$

/* β is an arbitrarily small no.*/

End

Procedure **BBO-NSGAI**(x_{curnt}, y_{curnt})

Begin:

- Initialize a random population P_0 Set $t=0$;
- Determine HSI of each solution of population P_0 ,
apply Non-dominated sort on P_0 based on those HSI.
- Calculate immigration and emigration rate for each solution in the population.
- Using Habitat modification, BBO migration creates a child solution and calculate their corresponding HSI.
- Then non-dominated sorting is performed on child population based on HSI.
- Mutation takes place for each child solution.
- Check $\| HSI_{PARENT} - HSI_{CHILD} \| < \epsilon$ or not.

If $\| HSI_{PARENT} - HSI_{CHILD} \| < \epsilon$

Select parent population.

else

If $HSI_{PARENT} > HSI_{CHILD}$

Select parent population.

Else If $HSI_{PARENT} < HSI_{CHILD}$

Select child population.

Else Call **Crowding_distance_determination**(P_{t+1} , F_i , N)

End

End

End

Procedure **Crowding_distance_determination**(P_{t+1} , F_i , N)

Begin :

- Initialize $P_{t+1} = \text{null}$, $i=1$, $F_i = N - (P_0 + Q_0)$;

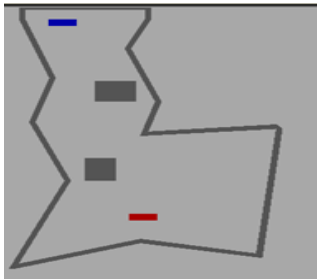
Repeat:

1. Calculate Crowding distance for non-dominated solution.
2. $P_{t+1} = P_t + F_i$;
 $i = i+1$;

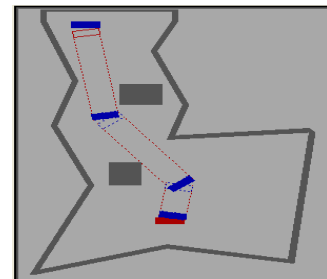
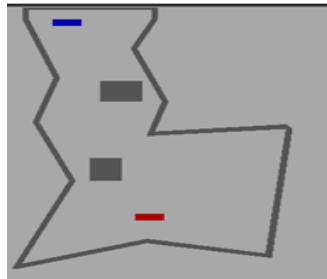
Until $(P_{t+1} + F_i) \leq N$

V. COMPUTER SIMULATION AND EXPERIMENTAL RESULT:

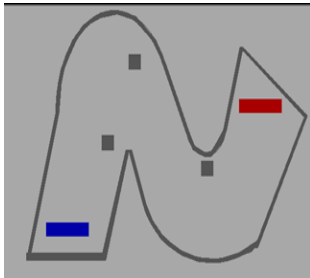
ARENA1 USING NSGAII



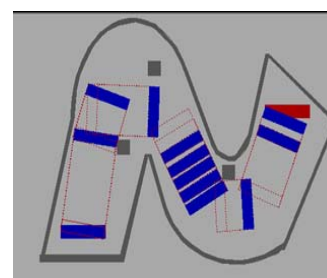
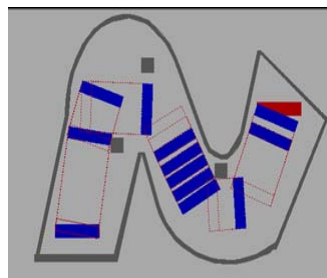
ARENA1 USING BBO-NSGAII



ARENA2 USING ABC-NSGA II



ARENA2 USING ABC-NSGA II



The experimental simulations for the environment ARENA1 and ARENA2 need 4 steps and 11 steps respectively to move the box to the goal position.

A brief description of the results for ARENA1 is presented in Table I and Table II. In Table I, we have entered the forces applied by two robots to turn the box, the turning angle, and the x, y co-ordinate of the point on the box around which turning is to take place and in table II we have provided the forces applied by the robot for translation, next position of centre of gravity, required time, energy consumptions for

the motion of the box. In table III we have compared the results of BBO-NSGA II and NSGA II and found that BBO-NSGAI gives better optimization for requiredtime and consumed energy.

A summary of the results for ARENA2 is presented in table IV and table V and the compared results are given in table VI.

TABLE-1FOR ARENA1

TURNING FORCES AND ANGLE OBTAINED IN BBO-NSGAI

Step	F_{1r}	F_{2r}	α	x_i	y_i
1	6.086127	19.881029	-0.165768	85.132693	30.000000
2	5.100903	14.156248	-0.503272	103.317528	140.444849
3	1.298498	31.535676	0.829523	152.410879	238.722542
4	6.086127	19.881029	-0.165768	85.132693	30.000000

TABLE-2FOR ARENA1

FORCES FOR TRANSLATION, NEXT CENTRE OF GRAVITY POSITION, TIMEAND ENERGY CONSUMPTION

Step	F_{1T}	X_c	Y_c	Time	Energy
1	2.051166	108.333239	139.605707	52.454930	493.840212
2	3.316401	173.953148	221.689258	40.686218	804.120242
3	1.533589	173.361907	281.005647	37.781158	173.874330
4	2.051166	108.333239	139.605707	52.454930	493.840212

TABLE-3FOR ARENA1

COMPARISON BETWEEN CURRENT AND PREVIOUS WORK

Method used	Total Time	Total Energy
BBO-NSGA II	130.922306	1471.834784
NSGA II	170.184352	1698.317755

TABLE-4 FOR ARENA2

TURNING FORCES AND ANGLE OBTAINED IN BBO-NSGA II

Step	F_{1r}	F_{2r}	α	x_i	y_i
1	42.418473	13.436211	0.153859	216.072142	308.0000
2	14.365552	48.334855	0.167977	202.059193	175.462524
3	1.266601	59.841370	1.286237	210.7697	112.098426
4	36.173891	8.883624	1.04288	307.988367	176.040632
5	59.343749	12.197916	-0.006447	328.527211	203.840576
6	35.914730	21.294524	-0.011449	340.152857	223.404154
7	37.796076	20.787651	0.001463	344.730793	243.813752
8	59.228602	5.912156	-0.027469	363.39229	256.354302
9	0.829737	58.993003	-1.101164	415.846034	248.545959
10	7.621267	53.757188	-1.145045	459.5831	246.207835
11	37.637947	6.865995	-0.003819	512.692990	179.098199

TABLE-5 FOR ARENA2

FORCES FOR TRANSLATION, NEXT CENTRE OF GRAVITY POSITION, TIME
AND ENERGY CONSUMPTION

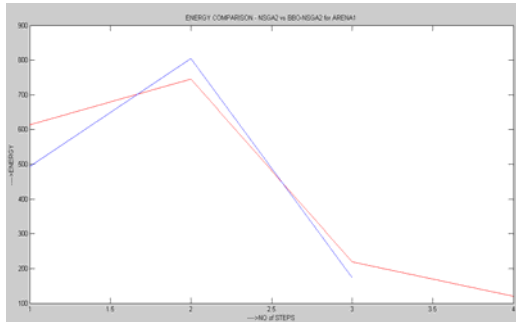
Step	F_{IT}	X_c	Y_c	Time	Energy
1	1.58306	215.232992	177.505572	64.006596	440.851599
2	6.125331	234.704148	120.078836	22.825549	440.320356
3	4.995232	309.292051	141.082992	34.120373	584.752027
4	3.703075	354.938309	189.731481	23.209369	342.535399
5	3.994147	365.678029	209.556059	16.964976	182.298378
6	6.006211	375.271957	226.788897	13.003685	242.596087
7	6.1863	385.010327	244.344984	12.764322	247.920433
8	3.256636	394.852086	260.975109	17.692334	133.215907
9	5.724099	461.138324	270.133974	22.336556	321.760559
10	18.025634	510.948578	178.439921	15.951547	1613.777044
11	2.488391	520.496553	152.990165	23.403304	136.263191

TABLE-6 FOR ARENA2

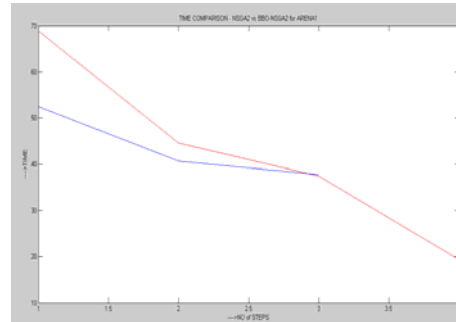
COMPARISON BETWEEN CURRENT AND PREVIOUS WORK

Method used	Total Time(sec)	Total energy(KJ)
BBO-NSGA II	266.278610	4686.290980
NSGA II	542.046899	6188.945924

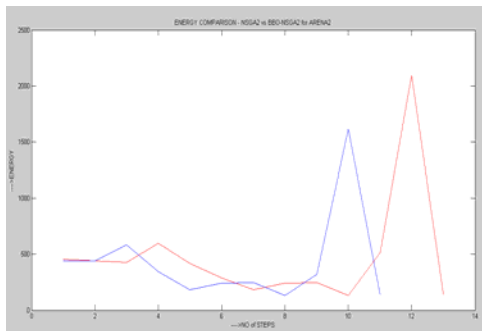
ENERGY COMPARISON GRAPH FOR ARENA1



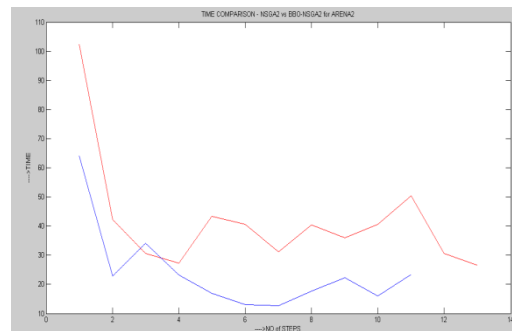
TIME COMPARISON GRAPH FOR ARENA1



ENERGY COMPARISON GRAPH FOR ARENA2



TIME COMPARISON GRAPH FOR ARENA2



VI.Conclusion:

We have successfully implimented the combination of BBO NSGAI to find optional solution to solve box pushing problem by two robots. It revealed that combination of BBO-NSGAI gives luttter optimization for required time and consumed energy in case of box pushing problem than that of NSGA-II alone.

Acknowledgement :

The authors are grateful to Prof. Amit Kumar, Dept. of Electronics and Telecommunication, Jadavpur University, Kolkata, and Prof. Prabir Chandra Bhattacharyya, Dept. of Mathematics, Camellia School of Engineering & Technology, Barasat, India for their kind help and valuable suggestions towards the preparation of this article

References

- 1) J. Chakrabarty, A.Konar,A.nagar,S.das, “Rotation and translation selective Pareto optimal solution to the box-pushing problem by mobile robots using NSGA-II” IEEE CEC 2009
- 2) Biogeography-Based Optimization Dan Simon, Senior Member, IEEE
- 3) An analysis of the equilibrium of migration models for biogeography-based optimization, Department of Electrical Engineering, Shaoxing University, Shaoxing, Zhejiang 312000, China
- 4) A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II Kalyanmoy Deb, Associate Member, IEEE, Amrit Pratap, Sameer Agarwal, and T. Meyarivan
- 5) F. C. Lin, and J. Y. -J. Hsu, “Cost-balanced Cooperation protocols in multi-agent robotic systems,” in International Conference on Parallel and Distributed Systems, pp.72, 1996.
- 6) T. Langle, and H. worn, “Human-robot cooperation using multi-agent systems,” Journal of Intelligent and Robotic system, vol. 32, pp. 143- 160, 2001.
- 7) B. Innocenti, B. Lopez, and J. Salvi, “A multi-agent architecture with cooperative fuzzy control for a mobile robot,” Robotics and
- 8) Autonomous Systems, vol. 55, pp 881-891, 2007.
- 9) R. A. Brooks, “A robust layered control system for a mobile robot,” Journal of Robotics and Automation, pp. 14-23, 1986.
- 10) C. R. Kube, and H. Zhang, “The use of perceptual cues in multi-robot box pushing,” in IEEE International Conference on Robotics and Automation, 1996, vol. 3, pp. 2085-2090.
- 11) Y. W. Leung, and Y. P. Wang, “Multiobjective programming using uniform design and Genetic Algorithm,” IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, 2000, vol. 3, pp. 293-304.
- 12) C. M. Fonseca, and P. J. Flaming, “Genetic algorithm for multi-objective optimization: Formulation, discussion, and generalization,” in Proceedings of the 5th International Conference on Genetic Algorithms, 1993, pp. 416-423.