Rotation and Translation Selective Pareto Optimal Solution to the Box-Pushing Problem by Mobile Robots Using NSGA-II

Jayasree Chakraborty, Amit Konar, Atulya Nagar, and Swagatam Das

Abstract—**The paper proposes a novel formulation of the classical box-pushing problem by mobile robots as a multi-objective optimization problem, and presents Pareto optimal solution to the problem using Non-dominated Sorting Genetic Algorithm-II (NSGA-II). The proposed method adopts local planning scheme, and allows both turning and translation of the box in the robots' workspace in order to minimize the consumption of both energy and time. The planning scheme introduced here determines the magnitude of the forces applied by two mobile robots at specific location on the box in order to align and translate it along the time- and energy- optimal trajectory in each distinct step of motion of the box. The merit of the proposed work lies in autonomous selection of translation distance and other important parameters of the robot motion model using NSGA-II. The suggested scheme, to the best of the authors' knowledge, is a first successful realization of a communication-free, centralized cooperation between two robots used in box shifting problem satisfying both time and energy minimization objectives simultaneously, presuming no additional user-defined constraint on the selection of linear distance traversal**.

I. Introduction

COOPERATION of mobile robots is an interesting area of modern research in multi-agent robotics [13], [14], [15]. Since 1990's researchers took active interest in formulating and solving the box-pushing problem by different techniques. Some of the well known works in this regard include adaptive action selection by the robots without communication [10], mutual cooperation by intension inference [9], cooperative conveyance by velocity adaptation of robots [11], and role of perceptual cues in multi-robot box-pushing [5]. The principle of subsumption architecture proposed by Brooks [16] was realized in a recent work [17] of cooperative box-pushing problem by mobile robots. The architecture proposed in this work combines the coordination principle of subsumption with motor schemas to obtain an efficient controlled movement of the box.

The work proposed in this paper, however, is different from the existing works on multi-agent robotics, as it attempts to satisfy multiple objectives concerning minimization of both time and energy in local trajectory planning of the box by employing an evolutionary algorithm. The NSGA-II algorithm used here provides a Pareto optimal solution, concerning different parameters of the box-pushing problem in each distinct step of local planning.

In this paper, we consider a special version of the box-pushing problem, called box-shifting, where two similar robots have to locally plan the trajectory of motion of the box from a predefined starting position to a fixed goal position in a complex terrain with non-liner boundary, containing one or more static obstacles. We presume that the robots do not have any background knowledge about their environment; consequently, the problem of box-shifting is solved here heuristically. Here, the robots jointly attempt to shift (both push and pull) a large box by applying forces at specific locations perpendicular to the edge of the box. The shifting of the box is performed by turning and translating the box in each step of local trajectory planning. The turning involves both push and pull operations, while translation requires only push operation by the robots [12]. In both the operations, the robots stand by one side of the box, and apply force perpendicularly to an edge of the box. A centralized local planning scheme has been adopted to determine the necessary turning angle and displacement of the box, and the magnitude of forces to be applied by the robots on the box. Sufficient spacing between the box and the obstacle is maintained during turning and translation of the box.

The most interesting issue of this paper is the formulation of box-shifting as a multi-objective optimization problem. The primary objectives of the box-shifting problem in this context are to minimize the time consumed by the robots for complete traversal of the planned trajectory, and to minimize the exploitation of the robots. In other words, we expect the robots to apply forces efficiently, so that the box is shifted from a given position to the next position (sub- goal) in a time and energy optimal sense without colliding with obstacles or the boundary of the world-map (robot's environment). To ensure the objective of minimizing time consumption for traversal of the box, we require maximizing the forces applied by the two robots. On the contrary, for minimum energy consumption, the robots have to apply minimum forces. So, there is trade-off between these two objectives. Consequently, the problem of box-shifting here, has been formulated as a multi-objective optimization problem, and has been solved using the well known and most popular multi-objective optimization algorithm, called Non-dominated Sorting Genetic Algorithm-II (NSGA-II) proposed by Deb et al. [2].

Jayasree Chakraborty is with Department of Electronics and Tele-Communication Engineering, Jadavpur University, Kolkata 700032, India (phone: +91 33 25142396; e-mail: jayasree2@gmail.com).

Amit Konar is with Department of Electronics and Tele-Communication Engineering, Jadavpur University, Kolkata 700032, India (e-mail: konaramit@yahoo.co.in).

Atulya Nagar is with Intelligence and Distributed Systems Lab, Liverpool Hope University, Liverpool L169JD (email: nagara@hope.ac.uk).

Swagatam Das is with Department of Electronics and Tele-Communication Engineering, Jadavpur University, Kolkata 700032, India (email: swagatamdas19@yahoo.co.in).

978-1-4244-2959-2/09/$25.00 ©2009 IEEE

The rest of the paper is organized into 5 sections. In section II, we provide a formulation of the problem. In section III, we provide an overview of NSGA-II and its application in box-pushing. In section IV, we demonstrate the experimental issues and computer simulations for the said problem. Conclusions are listed in section V.

II. FORMULATION OF THE BOX SHIFTING PROBLEM

In this section we use the basic problem formulation undertaken in [12] with slight extension in the nomenclature. We attach a notion of time t to current two dimensional positions of the box and the linear distance and angular rotation selected for motion of the box.

Let ABCD be the initial position of a box, at time $(t-1)$, represented by solid line in Fig. 1. Suppose two robots R_1 and R_2 are applying forces perpendicularly at points E and F on the front edge BC of the box.

Let O be the centre of gravity of the box at time $(t-1)$ and the coordinates of the points E, F, and O be $(x_e(t-1), y_e(t-1))$, $(x_f(t-1), y_f(t-1))$ and $(x_c(t-1), y_c(t-1))$ respectively.

Suppose the robots R_1 and R_2 together maneuvered the box around the point $I(x_i(t-1), y_i(t-1))$ by an angle $\alpha(t)$ and centre of gravity after rotation becomes $(x_{cr}(t), y_{cr}(t))$, and the corresponding new position of the robots E and F become $(x_{er}(t), y_{er}(t))$, $(x_{fr}(t), y_{fr}(t))$ respectively.

By using the principle of static's we derive the new positions of the robots and centre of gravity of the box, the x- and y- coordinates of which are explicitly given in (1) at time t.

$$\left.\begin{aligned}
x_{cr}(t) &= x_i(t-1)(1-\cos\alpha(t-1)) + x_c(t-1)\cos\alpha(t-1) \\
&\quad -\sin\alpha(t-1)(y_c(t-1)-y_i(t-1)) \\
y_{cr}(t) &= y_i(t-1)(1-\cos\alpha(t-1)) + y_c(t-1)\cos\alpha(t-1) \\
&\quad -\sin\alpha(t-1)(x_c(t-1)-x_i(t-1)) \\
x_{er}(t) &= x_i(t-1)(1-\cos\alpha(t-1)) + x_e(t-1)\cos\alpha(t-1) \\
&\quad -\sin\alpha(t-1)(y_e(t-1)-y_i(t-1)) \\
y_{er}(t) &= y_i(t-1)(1-\cos\alpha(t-1)) + y_e(t-1)\cos\alpha(t-1) \\
&\quad -\sin\alpha(t-1)(x_e(t-1)-x_i(t-1)) \\
x_{fr}(t) &= x_i(t-1)(1-\cos\alpha(t-1)) + x_f(t-1)\cos\alpha(t-1) \\
&\quad -\sin\alpha(t-1)(y_f(t-1)-y_i(t-1)) \\
y_{fr}(t) &= y_i(t-1)(1-\cos\alpha(t-1)) + y_f(t-1)\cos\alpha(t-1)
\end{aligned}\right\} \quad (1)$$

Consider another position of the box ABCD with its edge BC at an angle $\theta(t)$ with respect to x-axis. The box is now displaced by a magnitude $d(t)$.The new position of the centre of gravity of the box is now given by,

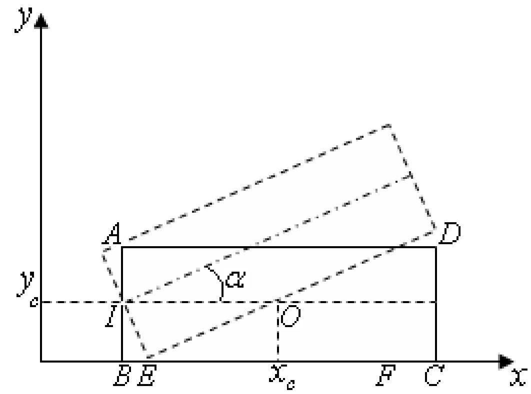

Fig. 1. Position of the box before (solid line) and after rotation (dashed line).

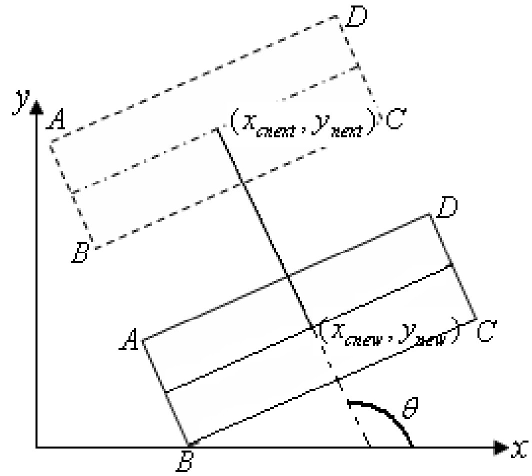

Fig. 2. Current (solid line) and next position (dashed line) of the box after translation.

$$\begin{aligned}
x_c(t) &= x_{cr}(t) + d(t)\cos\theta(t) \\
y_c(t) &= y_{cr}(t) + d(t)\sin\theta(t)
\end{aligned} \qquad (2)$$

We, now form an objective function concerning minimization of time, which has three components. The 1st component refers to the time required for rotation, denoted by t_1, where

$$t_1 = \sqrt{\frac{2\alpha(t)J}{T}} \qquad (3)$$

where, J=mass moment of inertia

T= Torque= $F_{1r}.d_1 + F_{2r}.d_2 = 2F_{1r}.d_1$,

since $F_{1r}.d_1 = F_{2r}.d_2$,

and, F_{1r} =force applied by R_1 to turn the box,

F_{2r} =force applied by R_2 to turn the box,

d_1 and d_2 are the perpendicular distance from the rotational axis to the line of action of the forces.

The 2nd time component refers to the time needed for translation of the box to the next position, while the 3rd time refers to the predicted time cost required for transportation of the box from the next position to the goal position. Let t_2 and t_3 be the respective times defined above. Evaluation of t_2 and t_3 follows from (4) and (5).

$$t_2 = \sqrt{\frac{2md(t)}{F_{1t} + F_{2t}}} \quad (4)$$

and $t_3 \propto \sqrt{S}$ or $t_3 = k_t \sqrt{S}$ (5)

where, m=mass of the box and K_t is a constant.

F_{1t} = force applied by R_1 to transport the box,

F_{2t} = force applied by R_2 to transport the box,

for translation only $F_{1t} = F_{2t}$

and S is the distance between the next centre of gravity and the goal position of the centre of gravity,

$$S = \sqrt{(x_c(t) - x_{cg})^2 + (y_c(t) - y_{cg})^2}$$

$$= \sqrt{\{x_{cr}(t) + d(t)\cos\theta(t) - x_{cg}\}^2 + \{y_{cr}(t) + d(t)\sin\theta(t) - y_{cg}\}^2}$$

Here (3), (4), and (5) are derived from the relations given below:

$T = J \times \omega$, where ω = angular acceleration,

$\theta(t) = \frac{1}{2}\omega t^2$, $S = \frac{1}{2}at^2$ and $F = ma$, where, a= linear acceleration.

So, the first objective function is,

$$f_1 = t_1 + t_2 + t_3 \quad (6)$$

Our 2nd objective function concerning minimization of energy consumption has also three components, energy consumption for rotation, and energy consumption for translation of the box to the next position and the predicted energy for transportation of the box from the next position to the goal position. If these energy consumptions are denoted by E_1, E_2, E_3 respectively, then the total energy consumption f_2 is obtained as

$$f_2 = E_1 + E_2 + E_3 \quad (7)$$

where, $E_1 = T\alpha(t) = 2F_{1r}d_1\alpha(t)$,

$E_2 = (F_{1t} + F_{2t})d(t) = 2F_{1t}d(t)$ and

$E_3 = k_e S$ where, k_e is a constant.

In our problem, it is also desired that the distance of the nearest obstacle in the direction of movement is as high as possible. For this, we introduce one penalty function. Thus, the 2nd objective function becomes,

$$f_2 = E_1 + E_2 + E_3 + f_{st} / dis_{obs} \quad (8)$$

Here, the objectives are the functions of $(x_i(t-1), y_i(t-1)), F_{1r}, F_{1t}$, $d(t)$ and $\alpha(t)$, which we have to determine to optimize the objective functions.

III. Solving The Box Shifting Problem Using NSGA-II

In this section, we first briefly outline NSGA-II for convenience of the readers, and then present the pseudo code for the entire scheme.

A. *Non-Dominated Sorting Genetic Algorithm (NSGA-II)*

In a multi-objective optimization problem, we usually need to optimize more than one *conflicting* objectives [1], [18], [19]. Naturally, finding a *true* optimal solution satisfying all the objective functions is not feasible. The general trend of solving multi-objective optimization is to determine a *Pareto optimal solution* set [1] to the problem. Several formulations for determining Pareto-optimal solutions to multi-objective optimization problem, employing evolutionary/swarm optimization algorithms are addressed in the current literature [1], [2], [3], [4], [6], [20], [21]. One of such evolutionary algorithms was proposed by Deb et al. in [2], which is well known as *Non-dominated Sorting GA-II* (NSGA-II). Due to its good spread of solutions and convergence near the true *Pareto-optimal front*, low computational requirements, elitism, and parameter less-niching, simple constraint handling strategy, it is widely used.

Like many other evolutionary algorithms, in NSGA-II also an initial population called parent population P_0 (at time t=0) of size N is randomly generated. Then, the population is sorted according to non-domination. Subsequent generations can be represented by discrete time steps: t = 1, 2, ... etc. After initialization, an iterative optimization process begins, where at the first step, using genetic operations i.e. binary tournament selection, recombination, and mutation operations child population Q_t of the same size N, is generated from the parent population P_t. Next, the parent and the child populations are combined to form the merged population R_t i.e. $R_t = P_t \cup Q_t$, which is of size 2N. Then, the next population P_{t+1} is constructed by choosing the best N solutions from the merged population R_t. Each solution is evaluated by using its rank as primary criterion and crowding distance as secondary.

The ranking is done based on the non-domination. All the non-dominated solutions in the merged population are assigned rank 1. The rank 1 solution set is called front set F_1. We now remove these solutions from the merged population, and again look for non-dominated solutions, if any, from the reduced merged population, and then assign rank 2 to these non-dominated solutions. The list of non-dominated solutions thus obtained is called front set F_2. In this way, rank is assigned to all the solutions. The members of the population P_{t+1} are chosen from subsequent non-dominated fronts in order of their ranking. Let F_l be the set, beyond which no other set can be accommodated. If by adding set F_l to P_{t+1}, size of P_{t+1} exceeds the population size then to select some solutions (N-$|P_{t+1}|$) from F_l, the set will be sorted based on the crowding distance, and the solutions with higher crowding distance are chosen. For maintaining good spread of solutions in the obtained set of solutions, the crowding distance concept has been introduced instead of choosing random solutions from F_l. Crowding distance of a solution is the sum of the difference between the function values of two adjacent solutions for all objectives i.e., to determine crowding distance of a solution, we have to sort the population according to each of the objective function value. Then, for each objective function, the solutions with maximum and minimum objective values are assigned infinite distance value, and the other intermediate solutions are assigned a distance value by taking the difference of the function value of their adjacent solutions, as shown in Fig. 3. Fig. 3 shows the solutions for two objective functions f_1 and f_2. Then from the above discussion, the distance value of the *i*-th solution, for the 1st and the 2nd objective functions will be,

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

$CR_{2distance}[i] = f_2[i-1] - f_2[i+1]$ respectively.

After calculating all the distance values for a solution, the crowding distance for the solution is obtained by taking the sum of the distance values corresponding to each objective functions. In this way, the crowding distances for all the solutions are obtained and according to the crowding distance, solutions from F_l are selected.

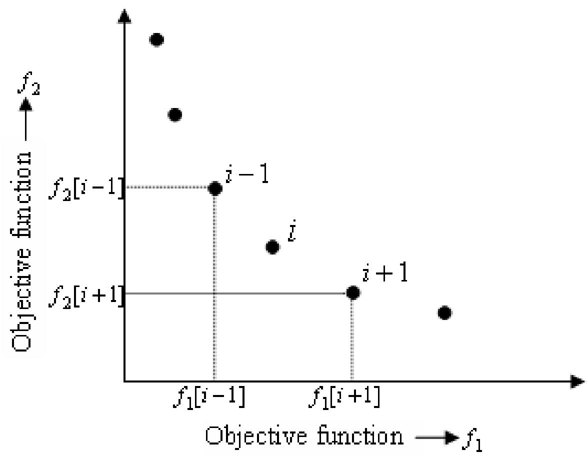

Fig. 3. The crowding distance calculation

Thus, by using ranking and crowding distance concept next population P_{t+1} is generated. This process is repeated for a certain number of time steps, or until some acceptable solution has been found by the algorithm.

Now, we propose a solution to the multi-objective box-shifting problem, which presumes current centre of gravity (CG) of the box, and determines the forces to be applied by the two robots to the box to shift it to the next position of the CG of the box. The pseudo code for the algorithm is presented below:

B. Pseudo code

Input: Initial CG of the box (x_c, y_c), final CG of the box (x_{cg}, y_{cg}), called goal position, points of application of the two forces on the box by the two robots are (x_e, y_e) and (x_f, y_f) and a threshold value ε (=6 selected in our experiment).

Output: forces applied by the two robots to move the box from (x_c, y_c) to (x_{cg}, y_{cg}),Euclidean distance 'd' between rotated CG and the next CG, and the rotational angle to align the box to the desired alignment in the next position

Begin
Set: $x_{curr} \leftarrow x_c$; $y_{curr} \leftarrow y_c$;
Repeat
Call NSGA-II ($x_{curr}, y_{curr}, x_e, y_e, x_f, y_f; \alpha, x_l, F_{1r}, F_{1t}, d$);
Move-to (x_{curr}, y_{curr});
Until $\| curr - G \| \le \varepsilon$
// $curr = (x_{curr}, y_{curr})$, G= ($x_{cg}, y_{cg}$)//
End.

Procedure NSGA-II($x_{curr}, y_{curr}, x_e, y_e, x_f, y_f; \alpha, x_l, F_{1r}, F_{1t}$)
Begin
Initialize a random parent population of size N=100;
Sort initial population based on the non-domination.
For
K< Maximum_Iterations do
Begin
Create child population (Q_t) using following three genetic operations: 1) Binary tournament selection, 2) recombination, and 3) mutation;
Combine parent and child population to form merged population i.e. $R_t = P_t \cup Q_t$;
Construct all non-dominated front sets ($F_1, F_2 ....$);
Set: $P_{t+1} = \phi$; i=0
Repeat
$P_{t+1} = P_{t+1} \cup F_i$;
i=i+1;
Until $|P_{t+1}| + |F_i| \le N$
End.
Calculate crowding distance in F_i;
Based on the crowding distance sort F_i in descending order;
$P_{t+1} = P_{t+1} \cup$ First ($N - |P_{t+1}|$) elements of F_i;
End.
End For.
End.
Update
1. (x_{cr}, y_{cr}), (x_{er}, y_{er}), (x_{fr}, y_{fr}) using (1).
2. $x_{curr} \rightarrow x_{curr} + d\cos\theta$; $y_{curr} \rightarrow y_{curr} + d\sin\theta$;
3. $x_e \rightarrow x_e + d\cos\theta$; $y_e \rightarrow y_e + d\sin\theta$;
4. $x_f \rightarrow x_f + d\cos\theta$; $y_f \rightarrow y_f + d\sin\theta$;
Return.
End.

During the selection process of the NSGA-II pseudo code, collision of the box with the obstacles is taken care of as constraints. In case the side edge of the new position of the box attempts to hit the obstacle, it is excluded from the trial solution list, and thus is never brought forward in the Pareto front. Consequently, the solution obtained from the NSGA-II based box-shifting problem never hits any obstacles.

IV. Computer Simulations and Experimental Result

The experiment was undertaken on a simulated environment on Intel Core-2-duo processor architecture with a clock speed of 3 MHz. The experiment includes construction of an obstacle map for the robots to transfer the box from a given starting position to a fixed goal position. A C-code program was developed for both time and energy optimization for the planned local trajectory of motion of the box using NSGA-II, the pseudo code of which is given in section III. Chromosomes used in the present context include five fields namely $x_i(t-1), F_{1r}$, F_{1t}, $d(t)$ and $\alpha(t)$. In order to move the robot to the next best optimal position, satisfying both time and energy optimization criteria [(6) and (8)], we determine the best chromosome in a single step of movement of the box. The best chromosome is obtained from the Pareto front after the NSGA-II converges. Since all chromosomes in the Pareto front are equally good, to select the one among many possible solutions, we normalize both time and energy for the individual chromosome in the Pareto front. Let ${}^i f_1(.)$ and ${}^i f_2(.)$ be the measure of time and energy fitness of the *i*-th chromosome (ch_i) in the Pareto front. The normalization here has been accomplished by using the following operation. Let ${}^i f_1^*(.)$ and ${}^i f_2^*(.)$ be the respective normalized value of the fitness measures for ch_i. Then, we define

$${}^i f_1^*(.) = \frac{{}^i f_1}{\sum_{i=1}^{n} {}^i f_1}$$

$${}^i f_2^*(.) = \frac{{}^i f_2}{\sum_{i=1}^{n} {}^i f_2}$$

The above process is repeated for all chromosomes in the Pareto front. Now to determine the effective chromosome to be used for one step of movement of the box, we take a new measure by taking the product of normalized time and energy, for the chromosomes present in the Pareto front. Let

$$P_i = ({}^i f_1^*(.) \times {}^i f_2^*(.)) \tag{9}$$

be a composite measure of time and energy optimality of the *i*-th chromosome. The effective chromosome ch_j having the smallest P_j for *j=1* to *n* is now identified for one step of transportation problem of the box by the robots.

The significance of the present work with reference to our previous work [12] lies in selection of $d(t)$ by the NSGA-II pseudo code and variation in K_e and K_t.

The entire trajectory of motion for the box-shifting problem was obtained for different settings of K_e and K_t in [10, 120]. A part of the experimental result is given in Table I. It is apparent from this table that the optimal performance for the given environment is obtained, when $K_e = 100$ and $K_t = 90$.

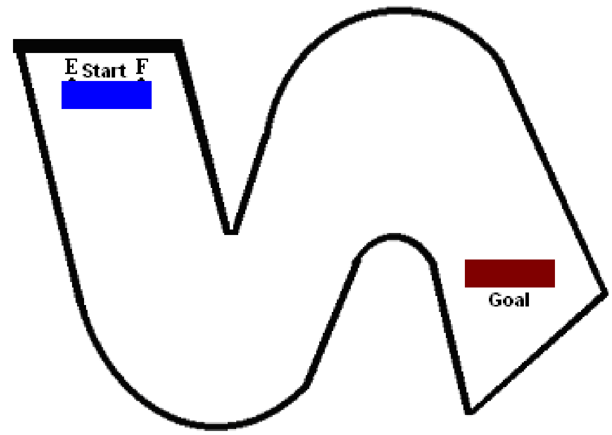

Fig. 4(a). Initial configuration.

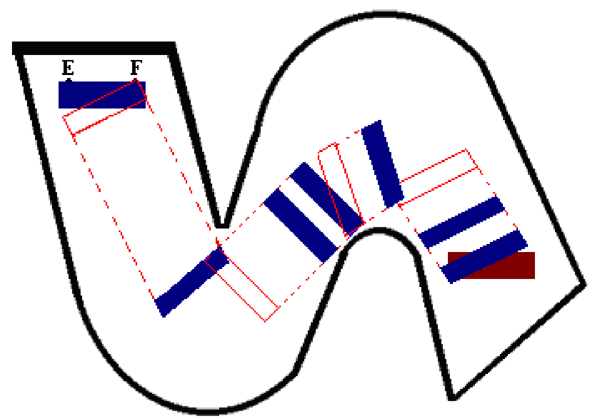

Fig. 4(b). Final configuration for the current work with time varying distance of traversal d(t).

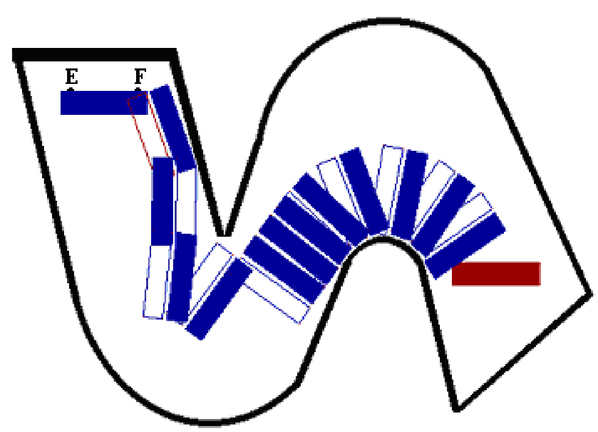

Fig. 4(c). Final configuration for the previous work [12] with constant distance of traversal, i.e. dt =d= constant.

The initial configuration of the world map for the box-pushing problem is shown in Fig. 4(a). The solution to the problem with demonstration of the motion of the box by the current and the previous approaches are given in Fig. 4(b) and 4(c) respectively.

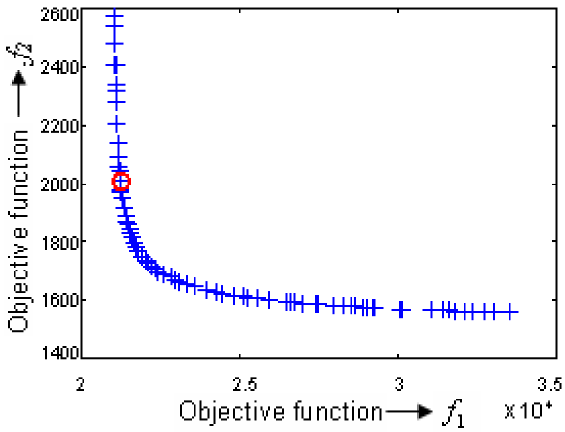

Fig. 5(a). Non-dominated solutions with NSGA-II for step 1.

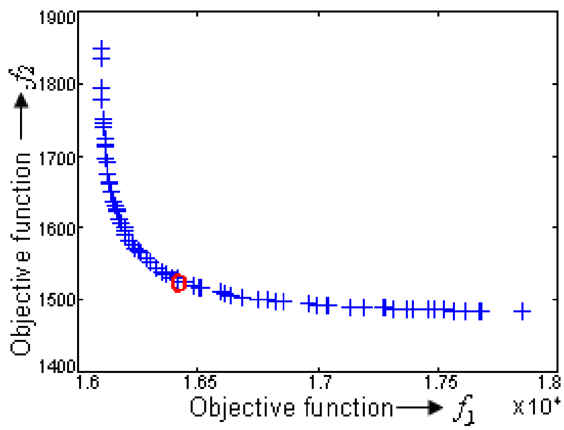

Fig. 5(b). Non-dominated solutions with NSGA-II for step 3.

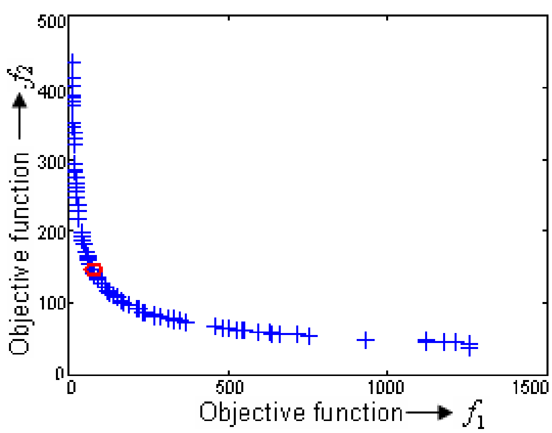

Fig. 5(a). Non-dominated solutions with NSGA-II for step 6.

The Pareto optimal fronts obtained in few successive movement of the box are shown in Fig. 5(a)-5(c).The effective solution on each Pareto front is marked in Fig. 5(a)-5(c). The effective solution indicates the chromosome with minimum normalized product of time and energy, as given in expression (9).

The experimental simulations for the environment (Fig. 4) require 6 steps to complete the motion of the box. Summary of the results for these 6 steps are presented in Table II and Table III. In Table II, we provide 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. The forces applied by the robot for translation, next centre of gravity position and required time and energy consumptions are given in Table III.

TABLE II

TOTAL ENERGY AND TIME FOR DIFFERENT K_T AND K_E

K_t	K_e	Energy	Time
10	50	3145.090	705.310
10	90	4871.879	688.548
10	100	3852.114	657.916
30	50	2598.746	685.515
30	90	2568.860	771.688
50	10	2544.319	957.267
50	30	2645.770	794.369
50	100	3228.275	656.399
70	30	2926.668	845.216
70	50	2894.794	792.266
70	90	2658.669	741.614
90	30	2587.274	809.698
90	70	2827.852	689.072
90	100	2565.012	657.098
100	30	2578.779	871.573
100	70	2924.186	669.029
100	90	2801.205	692.335
120	10	2546.995	1007.507
120	100	5531.977	648.359
120	120	2566.97	805.547

TABLE II

TURNING FORCES AND ANGLE OBTAINED BY USING NSGA-II

Step	F_{1r}	F_{2r}	α	x_i	y_i
1	3.215	24.716	0.5325	220.40	308.0
2	0.090	0.389	1.876	300.91	162.37
3	5.324	17.469	-0.0076	357.75	221.222
4	31.67	0.3218	-0.4289	410.79	205.28
5	9.287	0.2306	-1.5203	449.81	222.89
6	2.185	6.5338	-0.0053	524.29	179.35

TABLE III

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

Step	F_{1t}	x_c	y_c	Time	Energy
1	2.819	283.24	151.96	172.96	1137.7
2	2.362	370.82	209.45	145.43	404.59
3	3.198	386.9	227.09	157.76	61.389
4	3.949	437.5	251.79	76.797	356.01
5	2.479	509.49	172.16	127.39	400.76
6	2.241	519.8	150.73	73.121	108.15

A comparison of our present and previous work is given in table IV. In this table we provide the cumulative energy and cumulative time taken by the box for transportation from a given starting point to the fixed goal point in the prescribed world map of Fig. 4(a). It is apparent from Table IV that both time and energy consumption have been reduced by a margin of 1.64% and 15.523% respectively due to automatic selection of $d(t)$ using NSGA-II.

TABLE IV
COMPARISON OF OUR PRESENT AND PREVIOUS WORK

Method used	Total time	Total energy
Current	657.098	2565.012
Previous	668.097	3036.37

V. CONCLUSIONS

This paper provides a novel approach to handling box-shifting as a multi-objective optimization problem, and offers Pareto-optimal solutions in real time by utilizing the power of optimization of the NSGA-II program. The approach to the solution for the problem is unique and is different from the classical behaviour based [9] and perceptual cues based [5] multi-robot box-pushing problems. The merit of the work lies in online optimization of the cost functions, with an ultimate objective to minimize the traversal time of the box, and energy consumption by the robots using NSGA-II program. The solution obtained by the proposed technique is better than our previous realization of the problem with constant linear traversal of the box at each step of local planning. The optimal setting of K_t =90 and K_e=100 also improves the qualitative performance of the problem.

REFERENCES

[1] C. A. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with Particle Swarm Optimization," *IEEE Transactions on Evolutionary Computation*, 2004, vol. 3, pp. 256-279.

[2] K. Deb, A. P. S. Agarwal, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: NSGA II," *IEEE Transactions on Evolutionary Computation*, 2002, vol. 2, pp 162-197.

[3] 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.

[4] C. M. Fonseca, and P. J. Flaming, "Multi-objective optimization and multiple constraint handling with evolutionary algorithms-Part II: Application example," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 1998, pp. 38-47.

[5] 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.

[6] 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.

[7] M. J. Mataric, M. Nilsson, and K. T. Simsarian, "Cooperative multi-robot box-pushing," *In IEEE International Conference on Intelligent Robots and Systems*, 1995, vol. 3, pp.556-561.

[8] G. B. Parker and H. J. Blumenthal, "Competing sample sizes for the co-evolution of heterogeneous agents," in *IEEE International Conference on Intelligent Robots and Systems*, 2004, pp.1438-1443.

[9] H. Sugie, Y. Inagaki, S. Ono, H. Aisu, and T.Unemi, "Placing objects with multiple mobile robots- mutual help with intension inference," in *IEEE International Conference on Robotics and Automation*, 1995, pp.2181-2186.

[10] S. Yamada, and J. Saito, "Adaptive action selection without explicit communication for multi-robot box-pushing," in *IEEE International Conference on Intelligent Robots and Systems*, 1999, pp. 1444 -1449.

[11] Y. Yamauchi, S. Ishikawa, N. Uemura, and K. Kato, "On cooperative conveyance by two mobile robots," *in IEEE International Conference on Robotics and Systems*, pp. 1993, 1478-1481.

[12] J. Chakraborty, A. Konar, A. Nagar, H. Tawfik, "A multi-objective Pareto-optimal solution to the box-pushing problem by mobile robots," *Second UKSIM European Symposium on Computer Modeling and Simulation*, pp.70-75, 2008.

[13] 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.

[14] T. Langle, and H. Worn, "Human-robot cooperation using multi-agent systems," *Journal of Intelligent and Robotic system*, vol. 32, pp. 143-160, 2001.

[15] B. Innocenti, B. Lopez, and J. Salvi, "A multi-agent architecture with cooperative fuzzy control for a mobile robot," *Robotics and Autonomous Systems,* vol. 55, pp 881-891, 2007.

[16] R. A. Brooks, "A robust layered control system for a mobile robot," *Journal of Robotics and Automation,* pp. 14-23, 1986.

[17] L. Adouane, and N. L. Fort-Piat, "Evolutionary parameters optimization for an hybrid control architecture of multicriteria tasks," in *IEEE International Conference on Robotics and Biomimetics*, pp. 356-361, 2004.

[18] M .A. Abido, and J.M. Bakhashwain, "A novel multiobjective evolutionary algorithm for optimal reactive power dispatch problem," in *Proceedings of the 2003 10th IEEE International Conference on Electronics, Circuits and Systems, 2003 (ICECS 2003)*, vol. 3, pp. 1054--1057, December 2003.

[19] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: empirical results," *Evolutionary Computation*, vol. 8, pp. 173--195, 2000.

[20] M.A. Abido, "Two-Level of Nondominated Solutions Approach to Multiobjective Particle Swarm Optimization," in *Genetic and Evolutionary Computation Conference (GECCO 2007)*, London, UK, July 2007, pp. 726-733, vol. 1.

[21] A. Zamuda, J.. Brest, B. Boskovic and V. Zumer, "Differential Evolution for Multiobjective Optimization with Self Adaptation," in *2007 IEEE Congress on Evolutionary Computation (CEC'2007)*, IEEE Press, Singapore, September 2007,pp. 3617--3624.