A Multi-Objective Pareto-Optimal Solution to the Box-Pushing Problem by Mobile Robots

Jayasree Chakraborty
Artificial Intelligence Lab
ETCE Dept
Jadavpur University
Kolkata 700032, India
Jayasreechak2004@yahoo.co.in

Amit Konar
Artificial Intelligence Lab
ETCE Dept
Jadavpur University
Kolkata 700032, India
konaramit@yahoo.co.in

Atulya Nagar
Intelligence and Distributed Systems Lab
Deanery of Business and Computer Sciences,
Liverpool Hope University
Liverpool L169JD
nagara@hope.ac.uk

Hissam Tawfik
Intelligence and Distributed Systems Lab
Deanery of Business and Computer Sciences,
Liverpool Hope University
Liverpool L169JD
tawfikh@hope.ac.uk

Abstract

The paper provides a new formulation of the well-known box-pushing problem by robots as a multi-objective optimization problem, and presents Pareto-optimal solutions to the problem. The proposed method allows both turning and translation of the box, while shifting it to a desired goal position. Local planning scheme is employed here to determine the magnitude of the forces applied by two mobile robots at specific locations on the box to align and translate it in each distinct step of motion of the box, so as to minimize the consumption of both time and energy. This is realized using Non-dominated Sorting Genetic Algorithm-II (NSGA-II). The proposed scheme, to the best of the authors' knowledge, is a first successful communication-free, centralized co-operation between two robots applied in box-shifting, satisfying multiple objectives simultaneously using evolutionary algorithm.

1. Introduction

Since 1990's co-operation among robots remained a virgin area of modern research in multi-agent robotics. Box-pushing problem [5], [7], [8] is an interesting application of co-operation among robots. Various works on box-pushing have been reported in the literature. The works 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] among many others. The work presented in this paper, however, is unique and different from the above as it attempts to satisfy multiple objectives concerning minimization of both time and energy in real time, and employs evolutionary algorithms to obtain Pareto-optimal solution in each distinct step of local planning.

In this paper, we consider a specific version of the box-pushing problem (hereafter, box-shifting), where two similar robots have to plan the trajectory of motion of the box from a predefined starting position to a fixed goal position in a given environment, containing a number of static obstacles. The robots are capable of shifting (both pushing and pulling) a large box at specific location on the box. The shifting includes two basic operations: turning and translation. The turning involves both push and pull operations, while translation requires only push operation by the robots. In either operation, the robots stand in one side of the box, and apply force perpendicularly to the box. A centralized local planning scheme has been adopted to determine the next position/ turning angle of the box. Sufficient spacing between the box and the obstacle needs to be maintained during turning and translation of the box.

The most interesting issue of this paper is the formulation of the box-shifting as a multi-objective optimization problem. Here, the primary objectives of the box-shifting problem refer to a minimum time and minimum 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. To ensure minimum time constraint, the forces applied by the two robots should be maximized and on the contrary, for minimum energy consumption, the robots have to apply minimum forces. So, there is trade-off between these two objectives. The minimum time and minimum energy constraints, here, have been represented as a

978-0-7695-3325-4/08 $25.00 © 2008 IEEE
DOI 10.1109/EMS.2008.68

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].

The rest of the paper is organized into 5 sections. In section 2, we provide a formulation of the problem. In section 3, we provide a brief review on NSGA-II, followed by a discussion on the scope of NSGA-II in handling our problem. In section 4, we demonstrate the experimental issues and computer simulations for the said problem. Conclusions are listed in section 5.

2. Problem formulation

Let two robots R_1 and R_2 apply forces at point E (x_e, y_e) and F (x_f, y_f) respectively on a rectangular box ABCD, whose front wall is AD, current centre of gravity (x_c, y_c) and the final goal position of the centre of gravity is (x_{cg}, y_{cg}), as shown in Figure 1. If box turns around a point on (and within) the edge EF in clockwise direction, then R_1 pushes and R_2 pulls the box. For counter-clockwise turning, the robots change their role of push and pull operations.

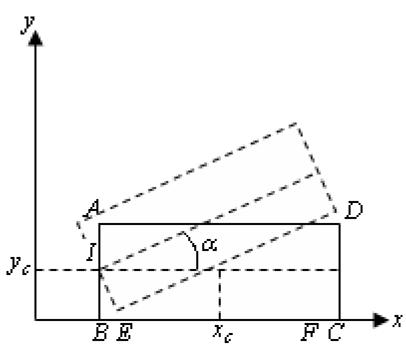

Figure 1. Position of the box before (solid line) and after rotation (dashed line)

Let α be the angle of rotation and $I(x_I, y_I)$ be the point around which the box is to be rotated. Suppose, after rotation the centre of gravity becomes (x_{cnew}, y_{cnew}), new positions of robots are (x_{enew}, y_{cnew}) and (x_{fnew}, y_{fnew}).

Using principles of kinematics, we can easily derive the following expressions,

$$\left.\begin{aligned}
x_{cnew} &= x_I(1-\cos\alpha) + x_c\cos\alpha - \sin\alpha(y_c - y_I)\\
y_{cnew} &= y_I(1-\cos\alpha) + y_c\cos\alpha - \sin\alpha(x_c - x_I)\\
x_{enew} &= x_I(1-\cos\alpha) + x_e\cos\alpha - \sin\alpha(y_e - y_I)\\
y_{enew} &= y_I(1-\cos\alpha) + y_e\cos\alpha - \sin\alpha(x_e - x_I)\\
x_{fnew} &= x_I(1-\cos\alpha) + x_f\cos\alpha - \sin\alpha(y_f - y_I)\\
y_{fnew} &= y_I(1-\cos\alpha) + y_f\cos\alpha - \sin\alpha(x_f - x_I)
\end{aligned}\right\} \quad (1)$$

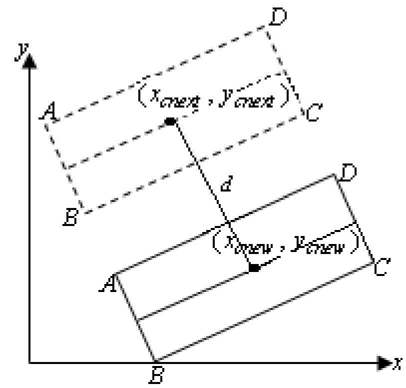

Figure 2. Current (solid line) and next position (dashed line) of the box after rotation

Now, after turning by an angle α (positive counterclockwise), the box moves with an alignment angle θ with the x-axis.

Let, the box moves d distance with the angle θ and its next centre of gravity is (x'_c, y'_c), where

$$\left.\begin{aligned}
x'_c &= x_{cnew} + d\cos\theta\\
y'_c &= y_{cnew} + d\sin\theta
\end{aligned}\right\} \quad (2)$$

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

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

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).

$$t_2 = \sqrt{\frac{2md}{F_{1t} + F_{2t}}} \tag{4}$$

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

where, m=mass of the box and k 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,

$$S = \sqrt{(x'_c - x_{cg})^2 + (y'_c - y_{cg})^2}$$
$$= \sqrt{\{x_{cnew} + d\cos\theta - x_{cg}\}^2 + \{y_{cnew} + d\sin\theta - y_{cg}\}^2}$$

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

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

$\theta = \frac{1}{2}\omega t^2$, $S = \frac{1}{2}at^2$ and $F = ma$, where a denotes acceleration of the mass m.

So, the first objective function is,

$$f_1 = t_1 + t_2 + t_3. \tag{6}$$

Our 2nd objective function concerning minimization of energy consumption has also three components, energy consumption for rotation, 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

$$f_2 = E_1 + E_2 + E_3 \tag{7}$$

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

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

$E_3 = k_1 S$ where, k_1 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,

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

Here, the objectives are the functions of $(x_I, y_I), F_{1r}\ F_{1t},$, and α, which we have to determine to optimize the objective functions.

3. Approach

As we have mentioned earlier that we use NSGA-II to solve the problem. For convenience of the readers, we now briefly outline NSGA-II before to state the pseudo code for the entire scheme.

3.1. Non-dominated Sorting GA-II (NSGA-II)

The presence of multiple objectives in a problem, gives rise to a set of optimal solutions known as Pareto-optimal solutions. Several formulations for determining Pareto-optimal solutions to multi-objective optimization problem, employing evolutionary algorithm (EA) have been reported in the literature [1], [2], [3], [4], [6]. 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 better spread of solutions and better 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, here also at first, 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 a primary criteria and crowding distance as secondary. The ranking is done based on the non-domination [1]. All the non-dominated solutions (front set F_1) in the merged population are assigned rank 1. Removing these solutions from the merged population if we find non-dominated solutions (front set F_2) from the reduced merged population then they will be assigned rank 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 is 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 from F_l, the set will be sorted using the concept of crowding distance [2], which is the sum of the difference between the function values of two adjacent solutions for all objectives. Adjacent solutions are identified with respect to each objective function value. In this way next population P_{t+1} is generated. This process is iterated 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 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:

3.2. Pseudo code

Input: Initial CG of the box (x_c, y_c), final CG of the box (x_{cg}, y_{cg}), called goal position, Euclidean distance d between rotated CG and the next CG, 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.

Output: forces applied by the two robots to move the box from (x_c, y_c) to (x_{cg}, y_{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 ($x_{curr}, y_{curr}, x_e, y_e, x_f, y_f; \alpha, x_I, y_I, F_{lr}, F_{lt}$);

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($x_{curr}, y_{curr}, x_e, y_e, x_f, y_f; \alpha, x_I, y_I, F_{lr}, F_{lt}$)

Begin

Initialize a random parent population of size N=100;

Sort initial population based on the non-domination.

For

K< Maxiter 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.

Update

1. (x_{cnew}, y_{cnew}), (x_{enew}, y_{enew}), (x_{fnew}, y_{fnew}) using expression 1.
2. $x_{curr} \to x_{curr} + d \cos\theta$; $y_{curr} \to y_{curr} + d \sin\theta$;
3. $x_e \to x_e + d \cos\theta$; $y_e \to y_e + d \sin\theta$;
4. $x_f \to x_f + d \cos\theta$; $y_f \to y_f + d \sin\theta$;

Return.

End.

4. Computer simulations and experimental results

In this section, we provide the results of computer simulations of the proposed scheme for the box shifting problem. Figure 3a demonstrates an initial configuration of the world map whose boundary region is shown by the dark color with the initial and final position of a rectangular box. The robots R_1 and R_2 (not shown in the Figure) apply forces at points E and F respectively on the front wall to move the box towards the given fixed goal position. The steps of movement of the box for the given world map is shown in Figure 3b. It is apparent that the box will first turn and then move by the forces applied by the two robots.

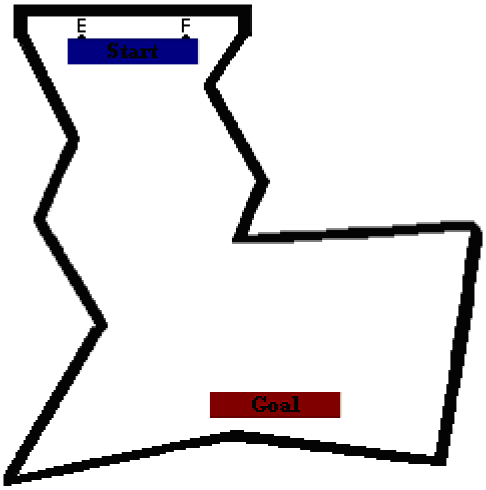

Figure (3a). Initial configuration

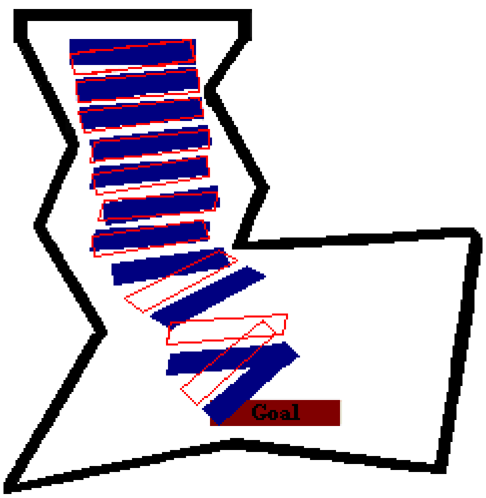

Figure (3b). Final configuration

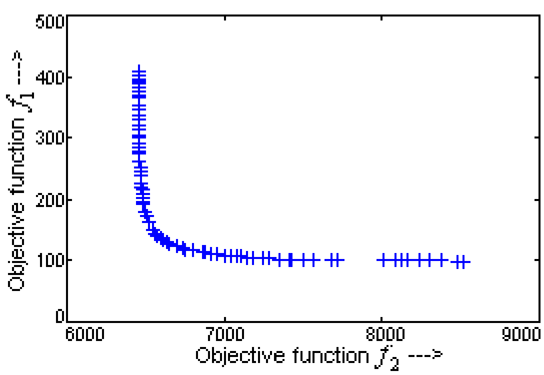

Figure (4a). Non-dominated solutions with NSGA-II for step 1

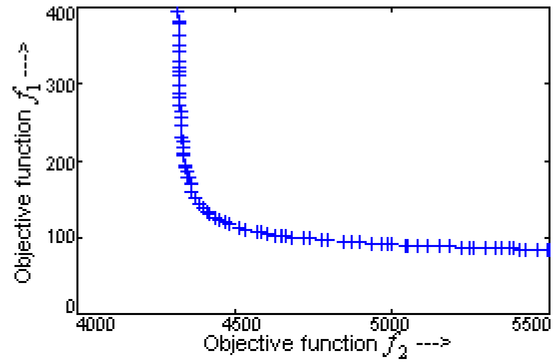

Figure (4b). Non-dominated solutions with NSGA-II for step 5

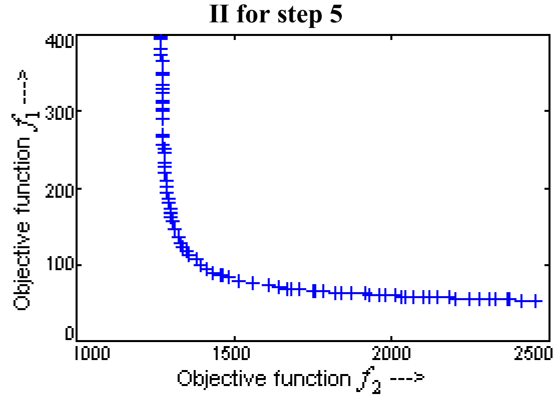

Figure (4c). Non-dominated solutions with NSGA-II for step 9

The Pareto-optimal fronts obtained in few successive steps of movements of the box are indicated in Figures 4a to 4c. Since Pareto-front represents contour of optimal solutions of the objective function we cannot use all of them in practice for application in the box-shifting problem. Consequently, we need to select a single solution. For the selection we normalize the fitness function of the solutions and select the one for which sum of the normalized fitness value is minimized.

The experimental simulations for the environment require 10 steps to complete the motion of the box. Summery of the results for these 10 steps are presented in Table 1 and Table 2. In Table 1, 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 2. These are selected from the Pareto-front obtained by NSGA-II program.

Table 1

Step	F_{1r}	F_{2r}	α	x_i	y_i
1	0.2100	4.26616	-0.1326	119.90	30.0000
2	3.5401	0.10369	0.0334	58.161	58.4174
3	0.2475	8.46313	-0.0417	129.71	71.3925
4	30.868	0.90781	0.0493	63.170	101.038
5	0.6440	21.8965	-0.0892	134.64	114.552
6	28.460	0.82977	0.1091	69.339	146.826
7	0.3290	1.01138	-0.1445	122.35	163.067
8	0.1451	4.93599	-0.6535	144.36	178.708
9	3.4499	0.10145	1.0340	114.46	245.151
10	0.3118	0.59344	-1.4677	155.83	272.278

Table 2

Step	F_{1t}	x_c	y_c	Time	Energy
1	3.8094	92.845	53.788	53.476	155.88
2	2.5531	94.899	74.858	64.124	102.59
3	3.2273	97.822	96.119	56.778	130.49
4	2.9603	99.791	117.75	58.752	124.50
5	2.8692	103.75	140.53	60.045	122.58
6	3.9077	105.60	164.27	51.570	168.62
7	2.9633	110.18	186.22	60.190	123.27
8	2.4665	137.03	218.40	69.426	111.56
9	2.9056	145.71	270.60	67.321	130.49
10	4.4863	172.40	287.52	54.587	219.06

5. Conclusions

The 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 NSGA-II programs. 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-shifting 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.

6. 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] R. Storn, and K. Price, "Differential evolution – A Simple and Efficient Heuristic for Global optimization over continuous spaces", *Journal of Global Optimization*, 1997, vol. 11, no. 4, pp. 341–359.