Songklanakarin J. Sci. Technol.
41 (5), 1116-1123, Sep. – Oct. 2019

http://www.sjst.psu.ac.th

Original Article

Improved differential evolution algorithms for solving multi-stage crop planning model in southern region of Thailand

Phajongjit Pijitbanjong[1], Raknoi Akararungruangkul[2], Rapeepan Pitakaso[3*], and Kanchana Sethanan[4]

[1, 3] *Department of Industrial Engineering, Faculty of Engineering, Ubon Ratchathani University, Warin Chamrap, Ubon Ratchathani, 34190 Thailand*

[2, 4] *Department of Industrial Engineering, Faculty of Engineering, Khon Kaen University, Mueang, Khon Kaen, 40002 Thailand*

Received: 23 March 2018; Revised: 18 May 2018; Accepted: 12 June 2018

Abstract

This paper presents algorithms based on Differential Evolution and Improved Differential Evolution for solving a multi-stage crop planning problem in southern region of Thailand to maximize the profit. Four types of algorithms were tested: 1) Differential Evolution (DE), 2) Differential Evolution with local search by adding the step of local search after the selection process, which used insert algorithm (DE-I), 3) Random best of Differential Evolution improved by mutations (DE-R), 4) Random best of Differential Evolution with local search as a mixture of types 2 and 3 (DE-IR). The results show that with small problem instances all the algorithms found a 100% optimal solution. In medium and large problem instances DE-IR shown the best solutions among the proposed algorithms.

Keywords: differential evolution, improved differential evolution, crop planning model

1. Introduction

Agriculture is important for the survival of modern dense populations. Agriculture requires good planning to produce sufficient quality and quantity to meet the necessities of consumers with reasonable costs. Previously, Thailand's agriculture was mostly monoculture with only the crop that had the greatest market value by locality. Most of the agriculture still is lacking in conventional systems and cost estimation, and the broad market price mechanisms that are crucial in determining cost. Moreover, market data also helps predict investment trends, including incomes to be derived. Rubber and Oil Palm are the major economic crops in the southern region of Thailand. They occupy more than 80% of the agricultural area. The volume of agricultural production increases every year. Farmers in the southern Thailand have focused on Rubber and Oil Palm because these are the economic crops that are essential for both domestic and international markets. Farmers have earned a lot of money during high market prices, but over the years, the global economy and trade have slowed down. Demand for rubber decreased causing continuous decline of the prices in the world market. Furthermore, the price of palm oil tends to fluctuate in accordance with economic and oil prices in the world market. Regarding Rubber and Oil Palm in the southern region of Thailand, both quality and price depend on the environment of the planting area and the volume of output. In this research, Rubber and Oil Palm plantations have been planned in areas that can provide good quality and high prices. It is necessary to consider the location of the purchasing place and the factory that affect transportation costs, in order to earn the maximum profit in the system.

This is an NP-hard problem, which is difficult to solve exactly as that would require a long computation time and is quite complicated. Researchers prefer to use meta-

*Corresponding author
Email address: enrapepi@ubu.ac.th

heuristic methods to solve such problems, with reduced time to find some near optimal solution. The Differential Evolution (DE) algorithm is an efficient way to finding such approximate solution and consume less time. *Adeyemo* and Otieno *(2010)* employed DE to solve multi-objective crop problems for water use planning, to maximize the profit. Similarly, *Zou, Liu, Gao, and Li (2011)* have adopted the DE algorithm and improved two parameters in it, namely factor size and crossover rate (CR). Scale factor can be adjusted and the CR was changed in steps. The answers were compared between two DE methods: Opposition-based Differential Evolution (ODE) and Adaptive Differential Evolution with Optional External Archive (JADE) in terms of cost and efficiency in the system. For this problem, there are many variables to be solved. It may take a long time to find the optimal solution. For example, Thongdee and Pitakaso (2012) developed a program to solve comparatively large problems. The computation time showed the status of the possible solution. And under all conditions in the design of mathematical models. {lacks rationality, makes no sense} Pitakaso and Thongdee (2014) used the DE algorithm to explore solutions of a Multi-Objective, Source and Stage Location-Allocation Problem. The results showed that DE-PSO gave better solution and took less time in solving problems than the standard DE and the modified DE for either small, medium or large problems. DE-PSO was 6.5% better than the standard DE and 2.8% better than the modified DE. In 2016, Sethanan and Pitakaso (2016) have proposed a method for improving the DE algorithm for solving the General Assignment Problem. Three methods of localization are used to improve the solution compared to the BEE algorithm and the Tabu algorithm in the Gapa-Gape trial suite, and the DE-SK out performs all other proposed algorithms.

This research applied DE and IDE in multi-stage crop planning. The first step is to plan the cultivation at a sub-district level, and the next is to find the location of the purchasing place and the factory for Rubber and Oil Palm, with regard to economic value to maximize profit for farmers, including operators, purchasing places, and factories. The next section presents a description of the problem. The methodology employed to solve the problem is presented in Section 2. Section 3 presents comprehensive details of our solution procedures and an outline of the experimental results is presented in Section 4. Finally, a summary of the main findings is given in Section 5.

2. Mathematical Model of the Crop Planning Problem

Mathematical model was developed created for solving crop planning problems in order to find suitable locations for Rubber and Oil Palm purchasing places and factories in order to gain the maximum profit. Parameters and decision variables used in formulating the model were defined. The 0-1 mixed integer programming formulation is presented below with a brief explanation of each constraint.

Indices

- i — farmer at sub-district; i in set of N = {1, 2, …, n}
- j — purchasing place; j in set of M = {1, 2, …, m}
- k — factory; k in set of O = {1, 2, …, o}
- l — type of crop; l in set of P = {1, 2} (1 = Rubber, 2 = Oil Palm)

Parameters

- N — number of farmers at sub-districts
- M — number of available purchasing places
- O — number of available factories
- P — number of types of crop
- a_i — planting area at sub-district i (rai)
- r_{il} — production rate of crop l at sub-district i (kg/rai)
- C^1_l — production cost of crop l (baht/kg)
- C^2_{jkl} — purchasing cost of crop l of purchasing place j to sell at factory k (baht/kg)
- d_{ij} — distance from sub-district i to purchasing place j (km)
- e_{jk} — distance from purchasing place j to factory k (km)
- P^1_{il} — selling price of crop l at sub-district i (baht/kg)
- P^2_{kl} — selling price of purchasing place l at factory k (baht/kg)
- H_l — maximum number of purchasing place of crop l
- Q_l — maximum number of factory of crop l
- F^1_{jl} — fixed cost to open the purchasing place j for crop l (baht)
- F^2_{kl} — fixed cost to open the factory k for crop l (baht)
- G^1_{ijl} — transportation cost of crop l from sub-district i to purchasing place j (baht/km)
- G^2_{jkl} — transportation cost of crop l from purchasing place j to factory k (baht/km)
- Z^1_{ijl} — vehicle loading capacity of crop l from sub-district i to purchasing place j (ton)
- Z^2_{jkl} — vehicle loading capacity of crop l from purchasing place j to factory k (ton)
- $S_{\min l}$ — minimum purchasing capacity of crop l for purchasing places (ton)
- $S_{\max l}$ — maximum purchasing capacity of crop l for purchasing places (ton)
- $T_{\min l}$ — minimum purchasing capacity of crop l for factories (ton)
- $T_{\max l}$ — maximum purchasing capacity of crop l for factories (ton)
- B_{kl} — profit per unit of crop l for factory k (baht/unit)
- T^1_l — number of round to transport crop l from sub-district to purchasing places (round)
- T^2_l — number of round to transport crop l from purchasing places to factories (round)

Decision Variables

- U_{il} — = 1 if farmer i to plant crop l, Otherwise 0
- X_{ijl} — = 1 if farmer at sub-district i to plant crop l and deliver to purchasing place j, Otherwise 0
- Y_{jkl} — = 1 if purchasing place j to deliver crop l to factory k, Otherwise 0
- V_{jl} — = 1 if purchasing place j to purchase crop l, Otherwise 0

W_{kl} = 1 if factory k to purchase crop l, Otherwise 0

Objective Function

$$\text{Maximize} \sum_{i\in N}\sum_{l\in P} a_i r_{il}\,(P^1_{il} - C^1_{l})U_{il} - \sum_{i\in N}\sum_{j\in M}\sum_{l\in P} d_{ij}\; G^1_{ijl} X_{ijl} T^1_l$$

$$+\sum_{j\in M}\sum_{k\in O}\sum_{l\in P}((P^2_{kl} - C^2_{jkl})\sum_{i\in N} a_i r_{il} \mathrm{V}_{jl}\; X_{ijl}) - \sum_{j\in M}\sum_{k\in O}\sum_{l\in P} e_{jk} G^2_{jkl} Y_{jkl} T^2_l$$

$$-\sum_{j\in M}\sum_{l\in P} \mathrm{F}^1_{jl}\mathrm{V}_{jl} + \sum_{i\in N}\sum_{j\in M}\sum_{k\in O}\sum_{l\in P} a_i r_{il} B_{kl} X_{ijl} Y_{jkl} - \sum_{k\in O}\sum_{l\in P} F^2_{kl} W_{kl} \quad (1)$$

Constrains

$$\sum_{l\in P} U_{il} = 1 \qquad \forall_{i\in N} \quad (2)$$

$$\sum_{j\in M}\sum_{l\in P} X_{ijl} = 1 \qquad \forall_{i\in N} \quad (3)$$

$$\sum_{j\in M} V_{jl} \le H_l \qquad \forall_{l\in P} \quad (4)$$

$$\sum_{i\in N} a_i\, r_{il} X_{ijl} \ge S_{\min\, l} \qquad \forall_{j\in M}\ \forall_{l\in P} \quad (5)$$

$$\sum_{i\in N} a_i\, r_{il} X_{ijl} \le S_{\max\, l} \qquad \forall_{j\in M}\ \forall_{l\in P} \quad (6)$$

$$X_{ijl} \le U_{il} \qquad \forall_{i\in N}\,\forall_{j\in M}\ \forall_{l\in P} \quad (7)$$

$$X_{ijl} \le V_{jl} \qquad \forall_{i\in N}\,\forall_{j\in M}\ \forall_{l\in P} \quad (8)$$

$$V_{jl} \le U_{jl} \qquad \forall_{j\in M}\,\forall_{l\in P} \quad (9)$$

$$\sum_{k\in O} Y_{jkl} = V_{jl} \qquad \forall_{j\in M}\,\forall_{l\in P} \quad (10)$$

$$\sum_{k\in O} W_{kl} \le Q_l \qquad \forall_{l\in P} \quad (11)$$

$$\sum_{j\in M}(\sum_{i\in N} a_i\, r_{il} X_{ijl})\, Y_{jkl} \ge W_{kl} T_{min\, l} \qquad \forall_{k\in O}\,\forall_{l\in P} \quad (12)$$

$$\sum_{j\in M}(\sum_{i\in N} a_i\, r_{il} X_{ijl})\, Y_{jkl} \le W_{kl} T_{max\, l} \qquad \forall_{k\in O}\,\forall_{l\in P} \quad (13)$$

$$Y_{jkl} \le W_{kl} \qquad \forall_{j\in M}\,\forall_{k\in O}\forall_{l\in P} \quad (14)$$

$$W_{kl} \le V_{kl} \qquad \forall_{k\in O}\,\forall_{l\in P} \quad (15)$$

$$C^2_{jkl} = P^1_{il} \qquad \forall_{i\in N}\,\forall_{j\in M}\,\forall_{k\in O}\forall_{l\in P} \quad (16)$$

$$T^1_l = \left\lceil \frac{\sum_{i\in N} a_i\, r_{il} X_{ijl}}{Z^1_{ijl}} \right\rceil \ge 0 \;\; and\ Integer \qquad \forall_{l\in P} \quad (17)$$

$$T^2_l = \left\lceil \frac{\sum_{i\in N} a_i\, r_{il} X_{ijl}}{Z^2_{jkl}} \right\rceil \ge 0 \;\; and\ Integer \qquad \forall_{l\in P} \quad (18)$$

$$U_{il}, X_{ijl}, Y_{jkl}, V_{jl}, W_{kl} \;\in \{0, 1\} \quad (19)$$

Equation 1; Objective function is to gain the maximum benefit, Equation 2; The farmer can plant only one type of crop in each sub-district, Equation 3; Each sub-district can deliver the crop to only one purchasing place, Equation 4, Numbers of purchasing places must not exceed the maximum numbers being allowed to open, Equation 5-6; The total volume of crop delivered to purchasing places must be in the range of minimum and maximum capacities of purchasing places. Equation 7; the number of sub-districts that deliver crop to purchasing places must not exceed the number of sub-districts that plant the crop. Equation 8; the number of purchasing places that purchase the crop must not exceed the number of available purchasing places. Equation 9; the number of purchasing places must not exceed the number of famers at sub-district. Equation 10; the number of purchasing places that deliver crop to factories must not exceed the number of available purchasing places. Equation 11; the number of factories must not exceed the number of the maximum number of available factories. Equation 12-13; the total volume of the crop delivered to factories must be between minimum and maximum capacities of the factory. Equation 14; the number of factories must not exceed the number of available factories. Equation 15; the number of factories must not exceed the number of available purchasing places. Equation 16; the selling price in each sub-district is equal to purchasing price of each purchasing places. Equation 17–18; the determination of the number of rounds to transport from any sub-district to purchasing places and from purchasing places to factories, and Equation 19; the determination of decision variable to become only 0 or 1.

3. Proposed Heuristics

3.1 Differential evolution algorithm

Differential Evolution algorithm can in a reasonable time find an approximately optimal solution. The process of DE has 4 main stages: 1) generate initial solution, 2) mutation, 3) recombination, and 4) selection. It was used to solve the multi-stage crop planning model as shown below.

3.1.1 Generate initial solution

Creating the initial response of the crop planning of farmers at a sub-district, locating the purchasing places as well as Rubber and Oil Palm factory by creating a target vector. The vector encoding has binary variables, and decoding these vectors gives an initial fitness value.

1) Vector encoding

Determine the target vector with size equal to the number of farmers. Each vector had dimension D. However, in this study, there was a division of crops for farmers by sub-district, so the number of vector coordinates for the answer is D+1. Vector 0 which is a division of the crop will be random (0 or 1) in each coordinate, for example, dividing the cropping groups into 13 farmers in a sub-district, the vector has 13+1 = 14 elements, and the number of population for the target vector = NP, for example as shown in Figure 1.

2) Vector decoding

The decoding is used to transform a vector into the problem's solution. For this problem the proposed decoding was as follows.

Step 1: Starting from sorting the random numbers in each vector respectively from the least to the largest. In Figure 2 is the sorting example of farmers at a sub-district and crop planning of the target vector.

Step 2: Grouping of farmers in each sub-district into 2 groups: in Figure 3 (A) is group 1, Rubber plantation; and in Figure 3 (B) is group 2, Oil Palm plantation. The coordinate 0 is the crop's division.

Step 3: Assign the farmer to deliver crop to the purchasing places. In order to assign a group of farmers, the total quantity of the delivered crop must be in the range of minimum and maximum capacities that can be purchased at each purchasing place. The first farmer at sub-district to be grouped is the location of each purchasing place.

Step 4: Assign the purchasing places to deliver crop to the factory similarly as grouping farmers at sub-district to each purchasing place. The first purchasing place to be grouped is the location of each factory.

Step 5: Fitness value from the objective function as in Equation 1.

Repeat steps 1-5 until all NPs are complete.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0.50	0.54	0.22	0.33	0.70	0.09	0.54	0.40	0.30	0.94	0.06	0.84	0.21	0.26

Figure 1. Example of the vector encoding

10	5	12	2	13	8	3	7	0	1	6	4	11	9
0.06	0.09	0.21	0.22	0.26	0.3	0.33	0.4	0.5	0.54	0.54	0.7	0.84	0.94

Figure 2. Sequence of the vector after sorting

10	5	12	2	13	8	3	7

(A) Farmers Group 1 (Rubber)

1	6	4	11	9

(B) Farmers Group 2 (Oil Palm)

Figure 3. The group of farmers

3.1.2 Mutation process

Modify the coordinates of the target vectors ($X_{i,j,G}$) by randomly assigning the values of same coordinates from 3 random vectors to the mutation process with Equation 20 in order to create a new vector, then the value of this new vector is called mutant vector.

$$V_{i,j,G} = X_{r1,j,G} + F(X_{r2,j,G} - X_{r3,j,G}) \quad (20)$$

as

$V_{i,j,G}$	mutant vector
X_{r1}, X_{r2} and X_{r3}	random vectors
F	scaling factor which is set to be 0.8 in the proposed heuristics (Qin, Huang, & Suganthan, 2009)
i	vector number; i = 1, 2, …., NP
j	position within vector; j = 0, 1, 2, …., D

3.1.3 Recombination process

When adjusting the value of the target vector until all NPs steps are taken, the next step is to find the trial vector ($U_{i,j,G}$) in the exchange process which is a mix of species to get new varieties of better and worse answers, random number (0,1) for all coordinates of the vector. Then, compare to the Crossover Rate (CR) = 0.8 (Qin et al., 2009). If the comparison shows that the value of the random number is less than or equal to the value of CR, select that position as the value of the mutant vector. If the value of the random number is more than the value of CR, choose the value of the target vector as shown in equation 21.

$$U_{i,j,G} = \begin{cases} V_{i,j,G} \text{ if } rand_{ij} \leq CR \text{ or } j = Irand \\ X_{i,j,G} \text{ if } rand_{ij} > CR \text{ or } j \neq Irand \end{cases} \quad (21)$$

3.1.4 Selection process

The selection is based on fitness by the objective function from the trial vector compared with the target vector, by choosing more profitable answers and appointing the new vector from this selection as the next target vector (G + 1), as in Equation 22.

$$X_{i,j,G+1} = \begin{cases} U_{i,j,G} \text{ if } (U_{i,j,G}) \geq f(X_{i,j,G}) \\ X_{i,j,G} \text{ if otherwise} \end{cases} \quad (22)$$

3.2 Improved differential evolution

The improved DE methods are of three types. 1) DE with local search, by adding the step of local search after the selection process, using insert algorithm. This step is adapted from method of Diaz and Fernandez (2001). 2) Random best of DE improves the process during the mutation. 3) Random best of DE with local search is a mixture of types 1 and 2, as follows.

3.2.1 DE with local search

This is the additional local search in the DE after the selection to improve the answer with an insert algorithm. as follows.

Insert algorithm is the way to change the position of a farmer who delivers the crop to the purchasing place, in order to receive a better solution. When the farmer at a sub-district i is assigned to deliver crop to the purchasing place j, the famer's position will be changed to deliver to the new purchasing place to increase the profit. For example, the farmer at sub-district 2 is appointed to deliver crop to the purchasing place 10 (Total profit 24,000 baht), and the algorithm changes the position of the farmer at sub-district 2 to deliver crop to the purchasing place 13 where the farmer can gain higher profit (Total profit 27,000 baht). However, the purchasing capacity of the purchasing place 13 must be sufficient for purchasing crop from a farmer at sub-district 2. Figure 4 (A) is division of 8 farmers at each sub-district who will deliver crop to the purchasing place into 2 groups. Group 1 consists of the farmers at sub-districts 10, 5, 12 and 2, appointed to deliver crop to the purchasing place 10. Group 2 consists of farmers at sub-districts 13, 8, 3 and 7, appointed to deliver crop to the purchasing place 13. After applying Insert algorithm, the farmer at sub-district 2's position will be changed to deliver the crop to the purchasing place 13 instead of purchasing place 10 as shown in Figure 4 (B).

The DE improves by local search can be implemented as the following steps.

Step 1 Generate initial solution
Step 2 Mutation process by equation (20)
Step 3 Recombination process by equation (21)
Step 4 Fitness value and Selection process by equation (22)
Step 5 Apply local search algorithm
Step 6 Repeat steps 2-5 until the loop is complete

3.2.2 Random best of DE

The approach to improve the DE method was in the mutation, using the random best algorithm after the selection. The X_{r1}, X_{r2}, and X_{r3} are random vectors mutate the target vector. Therefore, to keep the random vectors from the best new target vector after selection in each iteration was done as follows.

Step 1 Generate initial solution
Step 2 Mutation process
by equation (20) in iteration 1
by equation (23) from iteration 2

$$V_{i,j,G+1} = X_{r1best,j,G} + F(X_{r2best,j,G} - X_{r3best,j,G}) \quad (23)$$

Here

$V_{i,j,G+1}$ The next mutant vector

X_{r1best}, X_{r2best} and X_{r3best} random vectors of the best new target vector kept after the selection process, which can provide better solution.

Step 3 Recombination process by equation (21)

Step 4 Fitness value and selection process by equation (22)

Step 5 Apply random best algorithm

Step 6 Repeat steps 2 – 5 until the loop is completed

3.2.3 Random best of DE with local search. This is a mixture of 3.2.1 and 3.2.2.

4. Computational Experiment and Results

The DE implementations were of four types. 1) DE. 2) DE with local search. 3) Random best of DE. 4) Random Best of DE with local search. This C++ coded algorithms were applied to solve real problems of different sizes, namely small, medium and large problem instances. The computer had processor Intel (R) Core i3-3240 CPU 3.40 GHz and 4 GB memory. The parameters were NP = 50, G = 10,000, F = 2 and CR = 0.8. The results are presented in Table 1. A statistical comparison between DE and IDE is in Table 2 and the interval plot with 95% CI for the mean is in Figure 5. Figure 6 shows a crop planning model for southern region of Thailand.

The results showed that in small problem instances, all algorithms found a 100% optimal solution. In medium and large problem instances, DE-IR was efficient in maximizing the profit.

5. Conclusions

The developed DE algorithm for solving multi-stage crop planning model in the southern region of Thailand for maximum profit was tested in four variations: 1) Differential

Purchasing place	10				13			
Farmer at sub-district	10	5	12	2	13	8	3	7

(A) Result before applying insert algorithm

Purchasing place	10			13				
Farmer at sub-district	10	5	12	13	2	8	3	7

(B) Result after applying insert algorithm

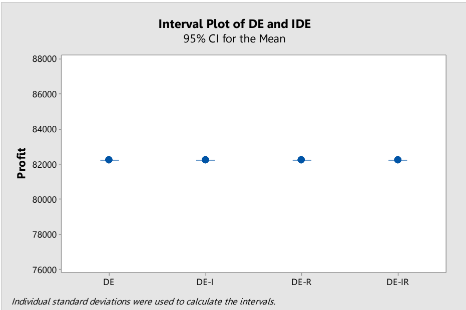
Figure 4. Example of insert algorithm

Table 1. Results of calculation with the same time interval.

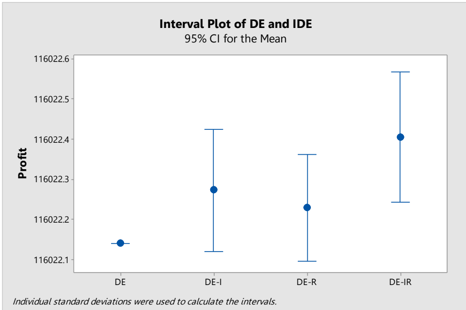
Size	Number of purchasing place		Number of factory		Maximum profit (baht)				
	Rubber	Oil Palm	Rubber	Oil Palm	Opt	DE	DE-I	DE-R	DE-IR
10	4	4	2	2	82,223.54	82,223.54	82,223.54	82,223.54	82,223.54
15	5	5	3	3	116,022.58	116,022.14	116,022.58	116,022.58	116,022.58
20	6	6	4	4	169,412.83	169,272.56	169,280.92	169,079.96	169,412.83
40	10	10	5	5	n/a	276,897.35	276,822.57	276,583.77	281,670.59
60	14	14	6	6	n/a	427,592.64	438,429.10	443,450.64	466,119.49
80	17	17	7	7	n/a	579,858.81	586,857.34	576,634.69	630,156.93
100	20	20	8	8	n/a	645,495.77	644,172.69	670,104.98	648,045.01
500	60	60	17	17	n/a	8,807,774.77	8,965,574.92	8,903,164.31	10,996,310.72
1083	100	100	30	30	n/a	10,752,348.02	10,831,669.78	10,918,446.59	12,366,624.02

Table 2. Statistical comparison between DE and IDE

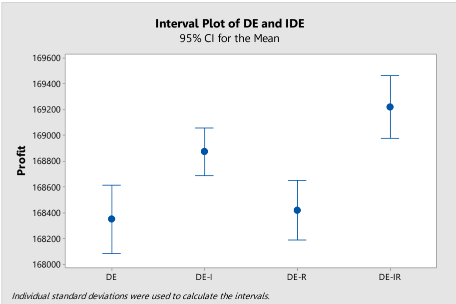
N	Difference of Levels	Difference of Means	SE of Difference	95% CI	Adjusted P-Value
15	DE-IR - DE	0.264	0.081	(0.0458, 0.4822)	0.012
20	DE-I - DE	522	145	(131, 914)	0.005
	DE-IR - DE	870	145	(478, 1261)	0.000
40	DE-IR - DE	1607	355	(652, 2562)	0.000
60	DE-I - DE	10497	2928	(2607, 18386)	0.005
	DE-R - DE	10278	2928	(2388, 18167)	0.006
	DE-IR - DE	37508	2928	(29618, 45397)	0.000
80	DE-I - DE	16219	5820	(541, 31898)	0.040
	DE-IR - DE	71058	5820	(55380, 86737)	0.000
100	DE-R - DE	43736	6802	(25410, 62063)	0.000
500	DE-IR - DE	1036739	243002	(382074, 1691404)	0.001
1083	DE-IR - DE	713660	249914	(40374, 1386947)	0.034



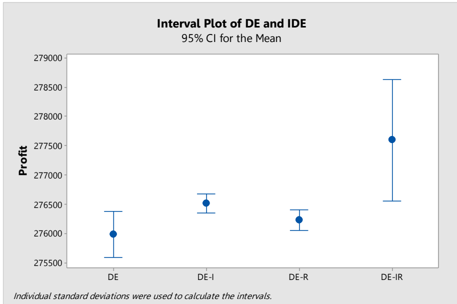
N = 10



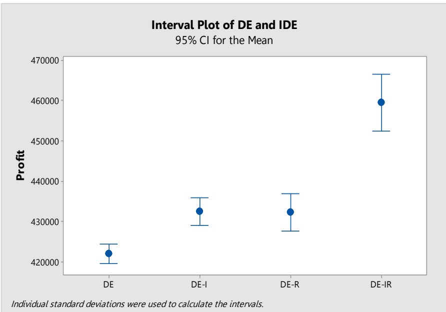
N = 15



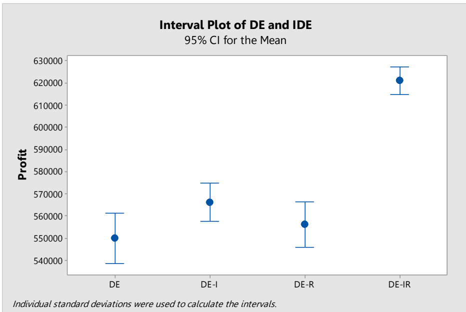
N = 20



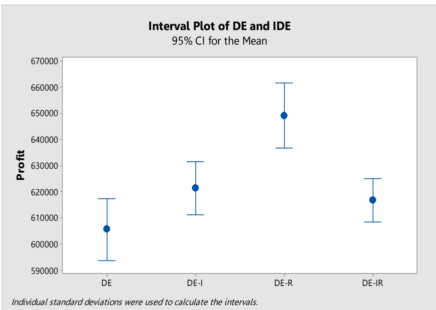
N = 40



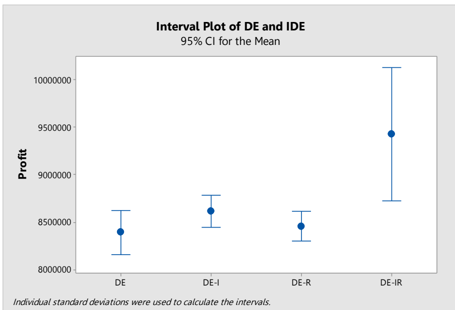
N = 60



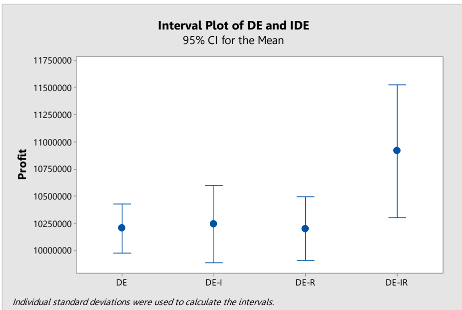
N = 80



N = 100



N = 500



N = 1,083

Figure 5. Interval plot with 95% CI for the mean

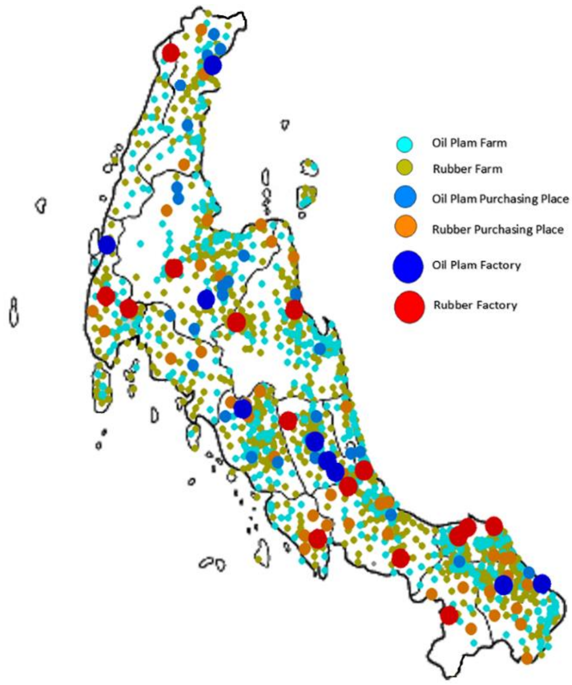


Figure 6. Crop planning model in southern region of Thailand

Evolution (DE), 2) Differential Evolution with local search (DE-I), 3) Random best of Differential Evolution (DE-R), and 4) Random best of Differential Evolution with local search (DE-IR). In small problem instances, all the algorithms found a 100% optimal solution. In medium and large problem instances, DE-IR was the best for maximizing the profit. Apparently the local search process and random best algorithm improved the method. Additionally, in small problem instances the small population makes it possible to find the optimal solution. However, with larger problem instances finding the optimal solution is not possible in a reasonable time. The DE-IR method increased the chances of finding a near optimal approximate solution, performing better than other proposed algorithms.

For this problem, the results for farmers in 14 provinces with 1,083 sub-districts suggested growing Rubber trees in 567 sub-districts and Oil Palm trees in 396 sub-districts. Additionally, 120 sub-districts had land free from growing any crops before. The findings also suggested 95 locations of purchasing places to be developed. They could be divided into 59 places for Rubber and 36 places for Oil Palm purchasing. Also 24 locations of factories (15 for Rubber and 9 for Oil Palm) were suggested as near optimal.

References

Adeyemo, J. & Otieno, F. (2010). Differential evolution algorithm for solving multi-objective crop planning model. *Agricultural Water Management, 97*(6), 848-856. Retrieved from https://www.sciencedirect.com/science/article/pii/S0378377410000314

Diaz, J. A. & Fernandez, E. (2001). A Tabu search heuristic for the generalized assignment problem. *European Journal of Operational Research, 132*(1), 22–38. Retrieved from https://www.sciencedirect.com/science/ article/abs/pii/S0377221700001089

Pitakaso, R. & Thongdee, T. (2014). Solving a multi-objective, source and stage location-allocation problem using differential evolution. In P. Golinska (Ed.), *Logistics operations, supply chain management and sustainability*. *EcoProduction environmental issues in logistics and manufacturing* (pp. 507-524). Cham, Switzerland: Springer. doi:10.1007/978-3-319-07287-6_36

Qin, A. K., Huang, V. L., & Suganthan, P. N. (2009). Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation, 13*(2), 398–417. Retrieved from http://ieeexplore.ieee.org/document/4632146/

Sethanan, K. & Pitakaso, R. (2016). Improved differential evolution algorithms for solving generalized assignment problem. *Expert Systems with Applications, 45*, 450-459. Retrieved from https://www.sciencedirect.com/science/article/pii/S095741741500696X

Thongdee, T. & Pitakaso, R. (2012). Solving a multi-objective, source & stage location-allocation problem: a case study of a bagasse and cassava pulp ethanol plant in northeastern Thailand. *Asia-Pacific Journal of Science and Technology, 17*(1), 71-87. Retrieved from https://tci-thaijo.org/index.php/APST/article/view/82826

Zou, D., Liu, H., Gao, L. & Li, S. (2011). An improved differential evolution algorithm for the task assignment problem. *Engineering Applications of Artificial Intelligence, 24*(4), 616-624. Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S0952197611000108