OPTIMIZING THE ESTABLISHMENT OF ASSEMBLY PLANTS WITH GENETIC ALGORITHM

Miklós Gubán – Ákos Gubán

1. SUMMARY

In the last years we researched the problem of establishing postponed assembling plants. The assembling plants are established close to the users in the postponed assembling, so that their special requirements regarding to the end products can be considered. We made a new mathematical model and an effective heuristic algorithm to handle this problem.

In this paper we would like to present a new solution for the task connected to the mathematical model. This solution is based on Genetic Algorithm. At the end of this paper we also present a little sample task to present the new algorithm.

2. MATHEMATICAL MODEL OF THE ESTABLISHMENT

1) We would like to present the original MP model. With this model and the associated LP model we can prove the convergence of heuristic method [1].
2) We will give a Genetic Algorithm solution for the original problem. In favor of effectiveness, some operations were defined specially.

2.1 The model

In the following we shall give the summary of the mathematical model. In details see in paper [1,2].

$$x_{ki}^{\nu} \geq 0;\ x_{ki}^{\nu} = \mathrm{int}\left(k = 1,\ldots n; i = 1,\ldots, p_0; \nu = 1,\ldots, m\right) \tag{1}$$

$$y_{kj}^{\mu} \geq 0;\ y_{kj}^{\mu} = \mathrm{int}\left(k = 1,\ldots n;\ j = 1,\ldots, r_0; \mu = 1,\ldots, w\right) \tag{2}$$

$$\sum_{k=1}^{n} x_{ki}^{\nu} = 1; \left(i = 1,\ldots, p_0;\ \nu = 1,\ldots, m\right) \tag{3}$$

$$\sum_{j=1}^{r_0} y_{kj}^{\mu} = 1; \left(k = 1,\ldots, n; \mu = 1,\ldots, w\right) \tag{4}$$

$$\sum_{i=1}^{p_0} x_{ki}^{\nu} q_{i\nu} \leq c_k^{\nu};\ \left(k = 1,\ldots n; \nu = 1,\ldots, m\right) \tag{5}$$

$$\sum_{i=1}^{p_0} x_{ki}^{\nu} q_{i\nu} \geq c^{\nu}; \left(k = 1,\ldots n; \nu = 1,\ldots, m\right) \tag{6}$$

$$\sum_{k=1}^{n} y_{kj}^{\mu} \sum_{\nu=1}^{m} \sum_{i=1}^{p_0} x_{ki}^{\nu} d_{i\mu} \leq b_{j\mu}; \left(j = 1,\ldots, r_0; \mu = 1,\ldots, w\right) \tag{7}$$

$$K_{red}\left(\mathbf{x},\mathbf{y}\right) = \sum_{i=1}^{p_0}\sum_{k=1}^{n}\sum_{\nu=1}^{m} x_{ki}^{\nu} \sum_{j=1}^{r_0}\sum_{\mu=1}^{w} k_{kj\mu\varepsilon}^{AS}\left(\sum_{t=1}^{m} q_{it} a_{t\mu}\right) l'_{kj} y_{kj}^{\mu} + \sum_{i=1}^{p_0}\sum_{\nu=1}^{m}\sum_{k=1}^{n}\left(k_{k\nu i}^{BS} c_{ki}^{\nu} l'_{ki} + k_{k\nu}^{M} c_{ki}^{\nu}\right) x_{ki}^{\nu} \rightarrow \min \tag{8}$$

$\mathbf{Q} = \left[q_{i\nu}\right]_{p_0 \times m}$ where $q_{i\nu}$ denotes the expected demand of user *i* for product ν.

$\mathbf{D} = \left\lfloor d_{i\mu} \right\rfloor_{m \times w}$, where $d_{i\mu}$ denotes the quantity of component μ to be built in products required by user *i*.

$\mathbf{B} = \left\lfloor b_{j\mu} \right\rfloor_{r_0 \times w}$, where $b_{j\mu}$ is the available annual production capacity of supplier *j* from component μ.

$\mathbf{C}_{H}^{\nu} = \left[c_{\chi}^{\nu}\right]_{n\times m}$, where c_{χ}^{ν} denotes the maximal capacity of product ν in the possible place χ.

$\mathbf{C}_{A}^{\nu} = \left[c^{\nu}\right]_{m}$, where c^{ν} denotes the minimal production capacity of product ν.

$\mathbf{Y}^{\mu} = \left[y_{kj}^{\mu}\right]_{n\times r_0}$. $y_{kj}^{\mu} = 1$ if we deliver component μ from supplier j to assembling plant k, otherwise it is zero.

$\mathbf{X}^{\nu} = \left[x_{ki}^{\nu}\right]_{n\times p_0}$. $x_{ki}^{\nu} = 1$ if we deliver product ν to user i from assembling plant k, otherwise it is zero.

3. GENETIC ALGORITHM

Description of the Genetic Algorithm (GA) can be found in the literature for example [8]. We will give the most important operations of GA below. These operations are Creating Random Initial Population, Crossover and Mutation. The fitness function is the original objective function (reduced cost function).

3.1 The structure of chromosome

The chromosome has got two branches **X** and **Y**. Each **X** has got *w* pieces **Y**-s.

Part **X** of chromosome contains the associating of users to possible places (digit 0, 1), part **Y** of chromosome the associating of suppliers to possible places for every part with the associated volumes.

3.2 Create initial population for run

3.2.1 Create one chromosome

Denote p_o the number of users. Denote z the number of possible places.

3.2.1.1 Create part **X** of chromosome

1. Do the follow statements p_o-times: Generate a random number between 1 and z in the step i. k denotes it. Take 1 in the place k of group i.

3.2.1.2 Create part **Y** of chromosome

2. After then calculate vector **r** using matrices $\mathbf{X}$, $\mathbf{Q}$.
3. Run throw components of vector **r** with index j. If $r_j = 0$, then ignore the component. Let $\mathbf{M} = \mathbf{B}$ and $V = t_k$.
   a. Now choose a supplier randomly with condition $m_l > 0$. Let it be l. If

   $m_l \geq V$, then $y_{jl} := V$ and $m_l := m_l - V$ and stop.

   $m_l < V$, then $y_{jl} := y_{jl} + m_l$, $m_l := 0$ and $V := V - y_{jl}$. Go to statement a.

3.3 The Element mutation

3.3.1 Crossover of part X of chromosome

1. Choose a group i of user randomly from the part **X** of chromosome.
2. Choose one element randomly from this group (generate a number j between 1 and z). Do this step until the value of element will be 0 ($i \neq j$).
3. Change the value of earlier element with value 1 to 0.

3.3.2 Mutation of part Y of chromosome

We will use the matrix **Y** for the mutation (establishment of suppliers). In the previous change supplier with random selection is the new assignment to possible place. Old user was i, the new is j for the user k.

4. Subtract t_k from component i of vector **r**, and add t_k to value m_j of element j.
5. Let $V = t_k$
   a. Choose a non null element l of row i of matrix **Y**. It can be two cases:

   $y_{il} \geq V$, then $y_{il} := y_{il} - V$ and $m_l := m_l + V$

   $y_{il} < V$, then $m_l := m_l + y_{il}$, $V := V - y_{il}$, $y_{il} := 0$ and go to step *a*. This algorithm will be finished surely.
6. Let $V = t_k$
   b. Choose a supplier l randomly, according to vector **M** (where $m_l > 0$). It can be two cases:

   $m_l \geq V$, then $y_{jl} := y_{jl} + V$ and $m_l := m_l - V$

   $m_l < V$, then $y_{jl} := y_{jl} + m_l$, $m_l := 0$ and $V := V - y_{jl}$. Go to step *b*.

3.4 The Element crossover

Two new chromosomes will be created from two old chromosomes.

3.4.1 Crossover of part **X** of chromosome

1. Given two chromosomes (a,b). Choose a group *k* randomly from part **X**.
2. Choose group *k* from chromosomes *a* and *b*. Change two groups each other.

Do the next steps with two chromosomes.

3.4.2 Crossover of part **Y** of chromosome

3. Regard the matrix **Y**. Subtract t_k from component *i* of vector **r**, and add t_k to value m_j of element *j*. Let $V = t_k$

   a. Choose a non null element *l* of row *i* of matrix **Y**. It can be two cases:

   $y_{il} \geq V$, then $y_{il} \coloneqq y_{il} - V$ and $y_{jl} \coloneqq y_{jl} + V$ and stop.

   If $y_{il} < V$, then $y_{jl} \coloneqq y_{jl} + y_{il}$, $V \coloneqq V - y_{il}$, $y_{il} \coloneqq 0$ and go to step *a*. This algorithm will be finished surely.

Note: In the step *a* we can choose the next way:

Order the possible suppliers in the row *l* with the distance:

$$l_{il} - l_{jl}$$

After then choose maximum value. In this case the assignment will be oriented. This method gives better convergence.

4. SAMPLE TASK

In this paragraph we present the result of our GA algorithm. This sample contains only one end product for the transparency. Conditions of the establishment are

- The product contains 2 parts.
- The product contains 10 part 1 and 6 part 2.
- where 4 end user with the following requirements: 5, 7, 8, 10 / product
- we have 5 suppliers
- Production capacity in the suppliers part 1:100, 20, 200, 50, 300; part 2:300, 200, 70, 100, 20
- maximum number of assembly plants: 10
- Production capacity of assembly plants: minimum: 100, maximum: 120
- cost of establishment of assembly plant: 1000

4.1 The result is with the one-off cost of establishment

The optimum result is the follow:

We have to establish possible places 2 and 5. The assignment is the follow (above Figure 1.).

The optimal cost is 8260

		part 1		**part 2**	
centrum	**users**	**centrum**	**suppliers**	**centrum**	**suppliers**
2	**1 →4**	**2**	**3 →150**	**2**	**3 →70 5 →20**
5	**2 →3**	**5**	**5 →150**	**5**	**5 →90**

	F1	F2	F3	F4	B1	B2	B3	B4	B5
L1	100	30	20	200	10	300	40	10	50
L2	30	200	100	10	30	400	10	30	20
L3	400	10	30	20	100	50	100	40	200
L4	100	20	300	30	50	60	100	300	10
L5	200	30	100	200	10	30	50	40	10
L6	10	30	60	20	10	60	90	100	200
L7	100	20	300	400	50	100	400	10	30
L8	100	30	20	400	10	20	300	10	200
L9	100	300	10	30	60	100	20	300	10
L10	20	30	100	400	20	30	80	100	20

Figure 1.
The Distance matrix

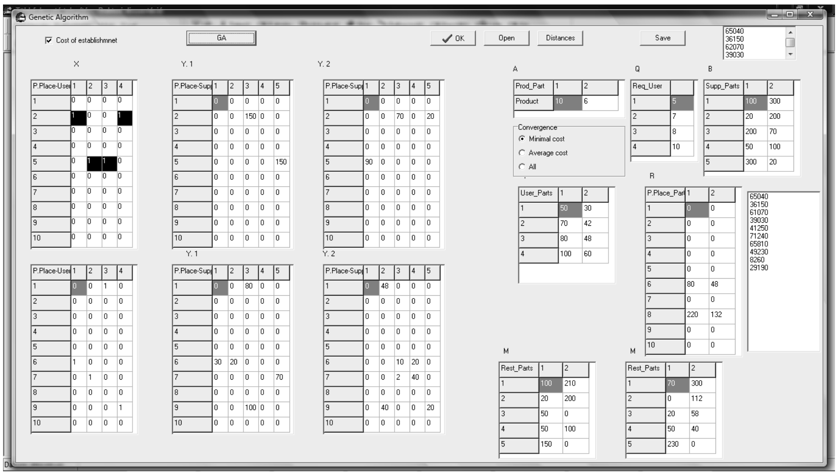

Figure 2.
Result of the demonstration program

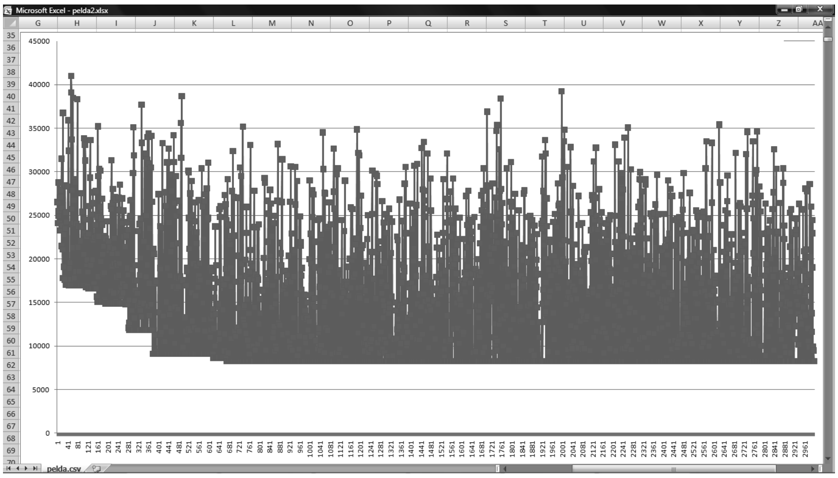

Figure 3.
Convergence of the goal function.

4.2 Convergence of the Cost Function

We want to examine the convergence of algorithm later. The next figure shows the convergence of the cost function. The goal is to find the minimum. The figure shows good the convergence.

5. CONCLUSION

In this paper we have given a mathematical model for the optimal establishment of postponed assembly plants and we have given a new solving method using GA.

By solving the problem with modified GA we have got a more effective and simplified solving method. This method gives the same results than heuristic method an LP method.

Problems: we cannot proof the convergence of the method, and stop condition. We will solve this problem later time.

The summarize we can tell the article, that the problem can be modeled well, and the GA is an effectiveness method for this problem.

6. LIST OF REFERENCES

[1] Dr. Gubán Miklós: (2006) **Késleltetett összeszerelő üzemek logisztika orientált telepítésére szolgáló matematikai modellek és módszerek fejlesztése globalizált termelés esetén,** Kiadja: GubánBros Bt, ISBN 963 06 0079 X, Salgótarján (monográfia)

[2] Gubán, M. 2001. **Heuristic algorithm to establish delayed assembling plants oriented by logistics.** 71-76 3rd International Conference of PhD Students. University of Miskolc. 13-19 August ISBN 963 661 480 6

[3] Gubán, M. & Cselényi, J. 2001. **Mathematical model and heuristic algorithm to establish delayed assembling plants oriented by logistics.** Miskolcer Gesprache

[4] Gubán, M. & Cselényi, J. 2004. **Quadratic linear-programming model to establish delayed assembling plants oriented by logistics:** 279-288 Logistics Networks. Models, Methods and Applications (Ed. T. Bányai, J. Cselényi) University of Miskolc, ISBN 963 661 641 8

[5] Gubán, M. & Cselényi, J. 2004. **The method and analysis of establishment of logistic-oriented postponement assembly plants,** Chapter 25, 255-264 DAAAM International Scientific Book, 2004, Wien, B. Katalinic (Ed.), Published by DAAAM International, ISBN 3 901509 38 0, ISSN 1726 9687, Vienna, Austria

[6] Á.Gubán, I. Kerepeszki: **Objective function analysis of biomass processing plants location,** 3rd Central European Conference in Regional Science – CERS, 2009. Košice, Slovak Republic

[7] Dr. Gubán Ákos: **Biomasszát feldolgozó üzemek telepítésének logisztikai célfüggvényei,** Válság és Megújulás, BGF, 2009.

[8] Guban, A. (1999) **The Genetic Algorithm and Application of GAs,** Analele Universităţii Din Oradea. Fascicola ELECTROTECHNICĂ Secţiunea: F. 54-59 MAI 1999. University of Oradea (Ed.), pp. 60-65, ISSN-1223-2106 Băile Felix, România