Instituto Nacional de Investigación y Tecnología Agraria y Alimentaria (INIA)
Available online at www.inia.es/sjar

Spanish Journal of Agricultural Research 2010 8(3), 672-678
ISSN: 1695-971-X
eISSN: 2171-9292

Computer aided selection in breeding programs using genetic algorithm in MATLAB program

M. Azimzadeh[1]*, R. Amiri[1], E. Davoodi-Bojd[2], H. Soltanian-Zadeh[2],
S. Vahedi[3] and M. Hoori[1]

[1] *Department of Agronomy and Plant Breeding Sciences. College of Abouraihan.*
University of Tehran. 33916-53775 Tehran. Iran
[2] *Control and Intelligent Processing Center of Excellence. University of Tehran.*
P.O. Box: 14395-515. Tehran. Iran
[3] *Plant Breeding Research Department. Sugar Beet Seed Institute.*
P.O. Box: 31585-4114. Karaj. Iran

Abstract

In plant and animal breeding, the best individuals are selected for the next breeding cycle based on the selection index computed from observed phenotypic values of several traits. However, in calculating the selection index, large amounts of data must be analyzed which is still performed by a calculator. This can cause imperfections in the breeding procedures. In this paper an automatic method for simulating a population under natural selection is proposed based on the selection operator of the genetic algorithms. The fitness function of the algorithm is a linear combination of the individual traits imported by the user. The algorithm generates both general and detailed scores of each trait for each labeled individual. The individuals are sorted with respect to their general scores and it is possible to extract individuals whose general scores are greater than a threshold defined by the user. The outlier individuals can also be eliminated. Moreover, for improved illustration and comparison, the individuals are displayed in a graph based on their index values. The proposed algorithm was applied to two distinct dataset and shown that results of the two methods coincide. The proposed method is automatic, fast, and free of human mistakes. Therefore, it is expected to improve the breeding procedures, especially when the numbers of individuals and traits are huge.

Additional key words: artificial selection, cucumber, selection index, simulation, sugarbeet.

Resumen

Selección con ayuda de ordenador en programas de mejora genética utilizando el algoritmo del programa MATLAB

Tanto en mejora vegetal como animal, se seleccionan los mejores individuos para el próximo ciclo de reproducción basándose en el índice de selección de varios caracteres calculado a partir de valores fenotípicos observados. Sin embargo, al calcular el índice de selección, se deben analizar gran cantidad de datos, lo que aún se realiza con una calculadora. Esto puede causar imperfecciones en los procedimientos de mejora. En este trabajo se propone un método automático para la simulación de una población sometida a selección natural basado en el operador de selección de los algoritmos genéticos. La función de aptitud del algoritmo es una combinación lineal de los caracteres individuales importados por el usuario. El algoritmo genera tanto puntuaciones generales como detalladas de cada carácter para cada individuo etiquetado. Los individuos son ordenados de acuerdo a su puntuación general y es posible extraer aquellos cuyos resultados generales son mayores que un umbral definido por el usuario. Los individuos anómalos también pueden ser eliminados. Para una mayor ilustración y comparación, se muestran los individuos en un gráfico según sus valores. Se aplicó el algoritmo propuesto a dos conjuntos de datos distintos y los resultados de los dos métodos coincidieron. El método propuesto es automático, rápido y libre de errores humanos. Por lo tanto, se espera que mejore los procedimientos de cultivo, sobre todo cuando el número de individuos es grande y los caracteres numerosos.

Palabras clave adicionales: índice de selección, pepino, remolacha, selección artificial, simulación.

* Corresponding author: mostafa_azimzadeh@yahoo.com

Received: 25-04-09; Accepted: 17-06-10.

Introduction

In plant and animal breeding, the best individuals are selected for the next breeding cycle on the basis of observed phenotypic values for several traits in each candidate individual (Cerón-Rojas *et al*., 2006). A selection index (Hazel, 1943) is a recommended method for selecting plants and animals when more than one trait is involved (Falconer and Mackay, 1996; Sivanadian and Smith, 1997) which usually can estimate an individual value in genotype collection (Yamada, 1989). In other words, the synchronized selection for all the important traits is the best selection method, by considering their heritability, economic value and also phenotypic and genotyping correlations between distinct traits. In this method, an index is defined as a single-trait based on which population's individuals are selected (Borojevic, 1990). The best selection method is based on the entire available information about each individual's breeding value (Hallauer and Miranda, 1981).

The first paper on selection index was written by Smith (1936) who applied directly the discriminant function of Fisher (1936) to multi traits selection in plant populations (Yamada, 1989). Breeders attained additional success using the selection index in comparison with direct selecting of the traits (Gravois and McNew, 1993; Jannink *et al.,* 2000). In other words, when improvement is desired for several traits that may differ in variability, heritability, economic importance, and in the correlation among their phenotypes and genotypes, simultaneous multiple-trait index selection was more effective than independent culling levels or sequential selection (Hazel *et al*., 1994).

Considering the importance and usefulness of the selection index and the selection processes in some breeding programs, it would be important to use the great capabilities of genetic algorithms in this field. Selection is one of the most important operators of a genetic algorithm and nowadays genetic algorithms are one of the best optimization and evolutionary computation methods (Winter *et al*., 1995; Mitchell, 1999). Evolutionary algorithms are searching and optimizing methods derived from Darwinian theory of evolution («natural selection» defined as: the process in nature by which only the organisms best adapted to their environment are selected). These algorithms try to simulate and imitate some evolutionary characteristics of survival in natural environments (Koza, 1992; Fogel, 2006).

A genetic algorithm is a search technique used in computing to find exact or approximate solutions to optimization and search problems. Genetic algorithms are a particular class of evolutionary algorithms (also known as evolutionary computation) that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (Mitchell, 1999; Reeves and Rowe, 2002). Genetic algorithms have three important operations selection, mutation and crossing-over that simulate conditions of individuals in natural environments (Mitchell, 1999; Wang *et al*., 2007).

Considering the importance of the selection indices in most breeding programs, and also the large amounts of data to be analyzed for the calculation of selection indices, a population was simulated under natural selection by using selection operator of the genetic algorithms in the MATLAB program (The MathWorks, Inc.). In this way, the best candidate individuals can be selected from the population accurately, easily, and quickly. Up to now, this field has not been extensively studied and genetic algorithms have not been used for the selection process in breeding programs. Our research is the first work in this field and thus we do not have a literature review about it.

Material and methods

Specifications of the algorithm

The proposed algorithm was developed in MATLAB environment. This software has powerful calculation facilities and is able to import and export data through spreadsheet files.

The fitness function of the algorithm (which corresponds to the selection index) is a linear combination of the individual features (traits) imported by the user. General scores, detailed scores of each trait, and also the label of each individual are displayed in the output. Individuals are sorted by their general scores and it is possible to extract individuals whose general scores are greater than a threshold defined by the user. This threshold can also be used for opposite selection in which a threshold is defined for eliminating outlier individuals. Individuals elected in that way form the next generation.

Structure of the algorithm

The proposed algorithm can be expressed in detail as follows:

```
0- begin
1- [input initial values]
    1-0   $th \leftarrow$ threshold
    1-1   $Nt \leftarrow$ Number of traits
    1-2   $Np \leftarrow$ Number of individuals (population
          size)
    1-3   $j \leftarrow 1$
    1-4   $a(j) \leftarrow \mathbf{in}$
    1-5   $j \leftarrow j+1$
    1-6   $if\ j <= Nt,$ go to 1-4
2- [input trait measures]
    2-0   $k \leftarrow 1$
    2-1   $j \leftarrow 1;$
    2-2   $T(k,j) \leftarrow \mathbf{in};$
    2-3   $j \leftarrow j+1$
    2-4   $if\ j <= Nt,$ go to 2-2
    2-5   $k \leftarrow k+1$
    2-6  $if\ k <= Np,$ go to 2-1
3- [input labels]
    3-0   $k \leftarrow 1$
    3-1   $L(k) \leftarrow \mathbf{in};$
    3-2   $k \leftarrow k+1$
    3-3   $if\ k <= N,$ go to 3-1
4- [compute the fitness function]
    4-0   $k \leftarrow 1$
    4-1   $I(k) \leftarrow \Sigma_{j}\, a_{j}(k).T_1(k,j)$
    4-2   $k \leftarrow k+1$
    4-3   $if\ k <= N,$ go to 4-1
5- [sort vector I decreasingly]
    5-0   flag$\leftarrow 1$
    5-1   $k \leftarrow 1$
    5-2   $if\ I(k) < I(k+1)$
                swap $(I(k)\ ,\ I(k+1))$
                swap $(L(k)\ ,\ L(k+1))$
                flag$\leftarrow 1$
          $else$   flag$\leftarrow 0$
    5-3   $k \leftarrow k+1$
    5-4   $if\ k < N,$ go to 5-2
    5-5   $if$ flag == 1, go to 5-1
6- [select the best individuals]
    6-0   $k \leftarrow 1$
    6-1   if $I(k) >$ threshold
                $sI(k) \leftarrow I(k)$
                $sL(k) \leftarrow L(k)$
          else  go to 7
    6-2   $k \leftarrow k+1$
    6-3  $if\ k <= N,$ go to 6-1
7- [display outputs ($sI$ and $sL$)]
8- end.
```

Step 1 of this algorithm gets the initial values of the threshold (*th*), number of traits (*Nt*), population size (*Np*), and the breeding values (a_j) from a spreadsheet file. In the step 2, the trait measures of each individual are read from the input file respectively and put in an *Np* by *Nt* matrix *T*. Similarly, in step 3 the corresponding labels of the individuals are put in to a vector *L*. So, in summary, these three steps are related to reading data from a spreadsheet file. Then, in step 4, the fitness function of each individual is computed using its traits measures. After that, in step 5, these fitness measures are sorted decreasingly and the individual's label rearranged. Finally, the best individuals are selected from the sorted fitness vector as those individuals whose fitness values are greater than the defined threshold (*th*). These selected individuals can be displayed or written to an output file. The corresponding block diagram of this algorithm is shown in Figure 1 which can facilitate its understanding.

Developing the proposed algorithm using MATLAB we enhance its performance by incorporating the powerful functionality of this software in vector and matrix calculations. Besides, the threshold and breeding values are imported straightforwardly by user. The remaining tasks such as reading the trait measures and the individual labels from a spreadsheet file, ranking, selecting, and displaying the best individuals are done by MATLAB automatically.

In fact, this algorithm simulates an evolving population by selecting the superior individuals, which can be very helpful and useful in breeding programs. The flow chart of the method is shown in Figure 1. In this figure, the release phase is the time in which the selected individuals are released as new cultivars or varieties and introduced to the farmers. The dashed part in the figure shows the phases of the breeding programs themselves and recombination of the next generation (offspring from the selected

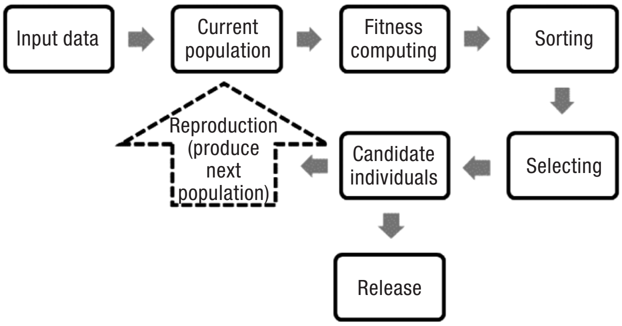

Figure 1. Schematic diagram of the proposed selection algorithm.

parents) which in turn can be used for the next selection cycle.

Efficiency test of the algorithm

To evaluate the validity and efficiency of the algorithm, results of Vahedi *et al.* (2005) and Aliabadi (2009) researches were used.

Vahedi *et al.* (2006) crossed five monogerm O-type lines of sugar beet (*Beta vulgaris* L.) with 15 monogerm CMS lines and resulted 75 monogerm F1 hybrids were divided to three sets containing 25 entries and each set compared by using a factorial North Carolina Design II (NCD II) of Comstock and Robinson (1952) in a randomized complete blocks designs with 25 entries and four replications. Combined analyses of the sets were performed and additive and dominance variances and covariances of important traits including root yield (RY), sugar content (SC), white sugar content (WSC), sugar yield (SY), nitrogen (N), sodium (NA) and potassium (K) content were estimated by expected values of mean squares (Hallauer and Miranda, 1981). The genetic variances and covariances of the important traits were considered to estimate heritability and correlation between the traits and to develop a selection index for monogerm germplasm of sugar beet. Four selection indices (Hazel, 1943) were developed by considering heritabilities, economic values and also phenotypic and genotyping correlations between the traits. The selection indices were tested on 300 individuals for choosing the most advantageous one (see Function [1]). This index includes root yield (RY), nitrogen (N), sodium (NA) and potassium (K) content which are important traits and have eligibility to be attended in indices for screening sugar beet genotypes.

$$I = -0.0183\ \mathrm{RY} - 7.957\ \mathrm{N} - 1.115\mathrm{Na} - 0.893\ \mathrm{K} \quad [1]$$

Aliabadi (2009) crossed seven female with eight male parents of cucumber (*Cucumis sativus* L.) based on a 7 × 8 factorial design (NCD II). Nine quantitative and qualitative components including length (L) and width (W), flesh fruit size and placental diameter/fruit diameter ratio (PD/FD) and skin texture firmness, flesh texture firmness, aroma, taste, crunchy and fruit dry matter content (DM) were evaluated in F_1 results based on a completely random design with three replications in greenhouse condition. Similar to Vahedi *et al.* (2006), this study was conducted to estimate genetic variance and covariances of important traits effective on flavour and to introduce a selection index for improving consumer savor of cucumber. In this study, three selection indices were computed and the best selection index was as Function [2]. This index includes taste (TST), aroma (ARM), crunchy (CRN), flesh texture firmness (FTF) and placental diameter/fruit diameter ratio (PD/FD):

$$I = 1\ \mathrm{TST} + 0.9\ \mathrm{ARM} + 0.8\ \mathrm{CRN} + 0.6\ \mathrm{FTF} + + 0.6\ \mathrm{PD/FD} \quad [2]$$

Results and discussion

The proposed method was evaluated through its performance calculating the selection index and selecting best individuals for the next generation and then comparing with the results of the method proposed by Vahedi *et al*. (2005) and Aliabadi (2009). In both cases, the dataset generated by these two researchers were used separately.

For better illustration of the outputs, the results of the algorithm are shown in Table 1 and Figure 2 for Vahedi's research and Table 1 and Figure 3 for Aliabadi's research. Columns in Table 1 represent: individual's number or label, total score of each individual and each trait's partial score (the number of these columns are equal to the number of trait) while rows represent individuals. Based on that information, individuals can be selected for achieving the breeding goals. It can be seen in Table 1 that 11 candidate individuals are selected by the algorithm based on their highest fitness value (higher than the defined threshold) out of 300 individuals of the entire population. Also for Aliabadi's research, it can be seen in Table 1 that five candidate individuals are selected by the proposed algorithm based their highest fitness function (higher than the defined threshold) out of 44 initial population in her research.

The output diagram of the algorithm schematically represents the fitness of each individual (Fig. 2 and Fig. 3 for both researches). In these figures, each point represents the fitness value of an individual, and the dashed line represents the user defined threshold which separates superior individuals. As seen in Figure 2a, 11 individuals have fitness values greater than the threshold, *th = –12,* and therefore they are selected. Also, in Figure 3a, five individuals have fitness values greater that the threshold, *th* = 16.5, and therefore they are selected. Note that by changing the threshold, the final number of selected individuals can be adjusted. In addition, by using this method, the outliers can be

Table 1. The result of selecting best individuals from the initial population in Vahedi *et al.* (2005) (threshold = –12) and Aliabadi (2009) (threshold = 16.5) researches using the proposed algorithm

Individual label	Variation 1	Variation 2	Variation 3	Variation 4	Variation 5	Final value
Vahedi *et al.* (2005)						
65	37.17	0.34	3.26	3.98		–10.57
18	48.06	0.41	2.9	3.96		–10.91
104	47.93	0.34	3.3	4.35		–11.15
55	36.98	0.33	3.95	3.9		–11.19
105	45.18	0.33	3.33	4.53		–11.21
31	45.77	0.37	3.99	3.39		–11.26
22	35.83	0.37	3.34	4.5		–11.34
91	43.81	0.35	3.44	4.91		–11.81
90	42.36	0.37	3.75	4.44		–11.87
64	36.77	0.37	4.25	3.96		–11.89
98	48.72	0.39	3.56	4.49		–11.97
Aliabadi (2009)						
33	1.775	2.75	3.025	17.175	1.915	18.124
44	2.650	2.60	2.484	15.360	1.640	17.178
25	2.308	2.70	3.167	14.280	2.064	17.078
23	2.250	1.517	4.500	14.405	1.635	16.830
6	2.458	1.984	2.925	15.053	1.712	16.643

eliminated. This can be done by adjusting the threshold in order to eliminate individuals very low fitness values. For example in Vahedi's research, by adjusting the threshold to –28, two individuals with very low fitness value are eliminated (Fig. 2b) and in Aliabadi's research by adjusting the threshold to 13.5, four individuals with very low fitness value are eliminated (Fig. 3b).

The results illustrate that the proposed method generates the same results as compared to Vahedi *et al.* (2005) and Aliabadi (2009) methods which were separately tested. However, the proposed method has some advantages. First, the whole procedure is done automatically which prevents from human inaccuracy and reduces the computation time to less than 0.1 seconds. Secondly, the proposed method has the flexibility for selecting the new individual by considering a threshold for the fitness value or by considering a fixed number of selected individuals. Therefore, the candidate individuals are easily selected based on the priorities of the user. Finally, the method has the capability to eliminate the outlier individuals.

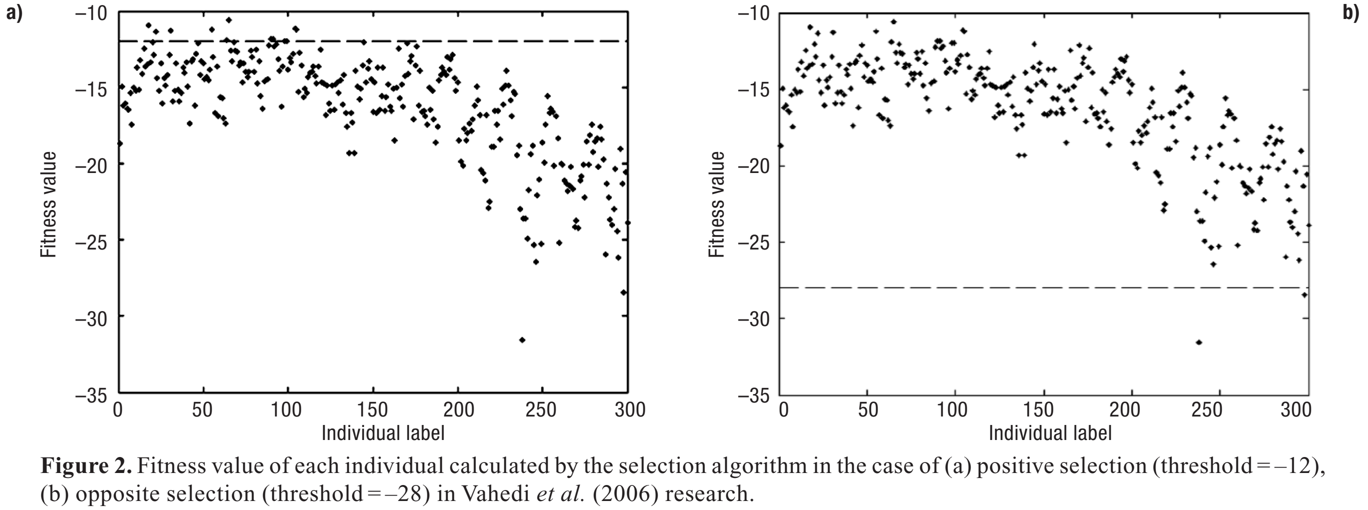

Figure 2. Fitness value of each individual calculated by the selection algorithm in the case of (a) positive selection (threshold = –12), (b) opposite selection (threshold = –28) in Vahedi *et al.* (2006) research.

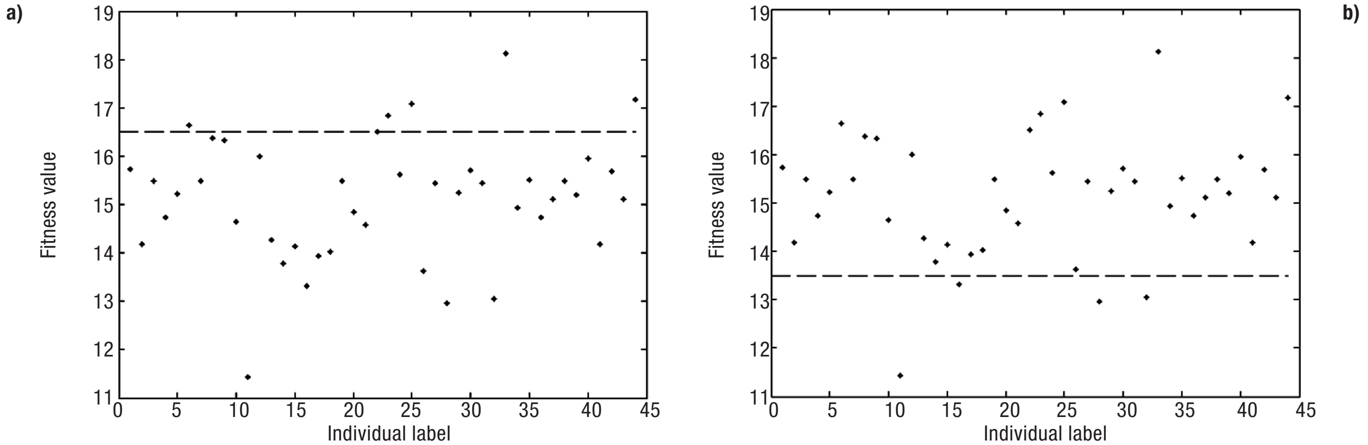

Figure 3. Fitness value of each individual calculated by the selection algorithm in the case of (a) positive selection (threshol = 16.5), (b) opposite selection (threshold = 13.5) in Aliabadi (2009) research.

Conclusions

In this paper an automatic method for selecting best individuals of a population is proposed which can be useful in plant and animal breeding tasks. The proposed algorithm is inspired from the evolutionary process of the genetic algorithms. The whole procedure is done automatically which prevents from human inaccuracy and reduces the computation time. On the other hand, this method has the flexibility for selecting new individuals by considering a fitness value threshold or a selected individual number threshold. This method has the capability for eliminating outlier individuals.

In future works, this algorithm can be used for simulating a plant or animal breeder task. This means that, it is possible to extend the algorithm and design fully automatic software for simulating these tasks. For example, in Figure 1, by designing a system for simulating the reproduction step, this goal can be achieved. It should be noticed that for designing this block some operators for combining the population individuals must be defined which should be consistent with the real reproduction actions.

References

ALIABADI E., 2009. Evaluating the traits related with fruit flavor and their heritability in cucumber. Master's thesis. University of Tehran, Iran. 110 pp. [In Persian].

BOROJEVIC S., 1990. Principles and methods of plant breeding. Elsevier, NY, USA. 396 pp.

CERÓN-ROJAS J.J., CROSSA J., SAHAGUN-CASTELLANOS J., CASTILLO-GONZÁLEZ F., SANTACRUZ-VARELA A., 2006. A selection index method based on Eigenanalysis. Crop Sci 46, 1711-1721.

COMSTOCK R.E., ROBINSON H.F., 1952. Estimation of the average dominance of genes. In: Heterosis (Gowen J.W., ed). Iowa State College Press, Ames, Iowa, USA. pp. 494-516.

FALCONER D.S., MACKAY T.F.C., 1996. Introduction to quantitative genetics. Longman, London, UK. 480 pp.

FISHER R.A., 1936. The use of multiple measurements in taxonomic problems. Annals of Eugenics 7, 179-178.

FOGEL D.B., 2006. Evolutionary computation: toward a new philosophy of machine intelligence. IEEE Press, Piscataway, USA. 274 pp.

GRAVOIS K.A., McNEW R.W., 1993. Genetic relationships among and selection for rice yield and yield components. Crop Sci 33, 249-252.

HALLAUER A.R., MIRANDA J.B., 1981. Quantitative genetics in maize breeding. Iowa State University Press, Ames, Iowa, USA. 468 pp.

HAZEL L.N., 1943. The genetic basis for constructing selection indexes. Genetics 28, 476-490.

HAZEL L.N., DICKERSON G.E., FREEMAN A.E., 1994. The selection index, then, now, and for the future. J Dairy Sci 77, 3236-3251.

JANNINK J.L., ORF J.H., JORDAN N.R., SHAW R.G., 2000. Index selection for weed suppressive ability in soybean. Crop Sci 40, 1087-1094.

KOZA J., 1992. Genetic programming: on the programming of computers by means of natural selection. MIT Press, Cambridge, MA, USA. 819 pp.

MITCHELL M., 1999. An introduction to genetic algorithms. MIT Press, Cambridge, MA, USA. 209 pp.

REEVES C., ROWE J.E., 2002. Genetic algorithms, principles and perspectives: a guide to GA theory. Cambridge Univ Press, NY, USA. 332 pp.

SIVANADIAN B., SMITH C., 1997. The effect of adding further traits in index selection. J Anim Sci 75, 2016-2023.

SMITH H.F., 1936. A selection index for optimum genotype. Biometrics 18, 120-122.

VAHEDI S., AMIRI R., MESBAH M., BIHAMTA M.R., RANJI Z., SADEGHZADEH S., 2006. Introducing selection index for monogerm germplasm of sugar beet. Proc 9th Intnl Cong Plant Breeding and Agronomy, Tehran, Iran, August 13-15. pp. 168-172.

WANG S., WANG Y., DU W., SUN F., WANG X., ZHOU C., LIANG Y., 2007. A multi-approaches-guided genetic algorithm with application to operon prediction. Artif Intell Med 41, 151-159.

WINTER G., PERIAUX J., GALAN M., CUESTA P., 1995. Genetic algorithms in engineering and computer science. John Wiley & Sons, NY, USA. 464 pp.

YAMADA Y., 1989. Selection index for attaining breeding goals. Proc 38th annual National Poultry Breeders Roundtable. St Louis, MO, USA, May 4-5, 1989. pp. 127-148.