Genetics/ Original Article

Stopping criteria for genetic improvement software for beef-cattle mating selection

Abstract – The objective of this work was to propose a new stopping criterion to shorten the computing time of the PampaPlus genetic improvement software, while maximizing the genetic qualification index (GQI) of the progeny, controlling inbreeding, and avoiding unintended culling. Data from two beef-cattle herds integrating PampaPlus were used. Five mating scenarios were built using different numbers of sires (9 to 37) and dams (142 to 568). The analyzed algorithm inputs were: expected progeny differences, pedigree information, maximum inbreeding, maximum and minimum number of matches for each sire, and penalty weights for poor performance. The analyzed response variables were computing time and the GQI of the progenies. Three stopping criteria were used: original stopping criterion fixed at 1,000 iterations; saturation stopping criterion (SSC), based on GQI variance; and Bhandari's stopping criterion (BSC), which includes the generation interval parameter. SSC and BSC reduced processing time in 24.43–53.64% and in 14.32–50.87%, respectively. BSC reaches solution in less time, without losses in GQI quality. BSC is generalizable and effective to reduce the processing time of mating recommendations.

Index terms: algorithm, animal breeding, decision support, mating systems.

Critérios de parada para programa de melhoramento genético para seleção de acasalamentos em gado de corte

Resumo – O objetivo deste trabalho foi propor um novo critério de parada para diminuir o tempo de processamento do programa de melhoramento genético PampaPlus, além de maximizar o índice de qualificação genética (GQI) da progênie, controlar a endogamia e evitar o descarte não intencional. Foram utilizados dados de dois rebanhos integrantes do PampaPlus. Cinco cenários de acasalamento foram elaborados com diferentes números de touros (9 a 37) e vacas (142 a 568). Os dados analisados foram: diferenças esperadas na progênie, informações de pedigree, máxima endogamia, número máximo e mínimo de acasalamentos por touro, e penalidades para desempenho inferior. As variáveis analisadas foram tempo de processamento e o GQI das progênies. Foram utilizados três critérios de parada: critério de parada original, fixado em 1.000 iterações; critério de parada por saturação (SSC), baseado na variância do GQI; e critério de parada de Bhandari (BSC), que inclui o parâmetro de intervalo de gerações. O SSC e o BSC reduziram o tempo de processamento em 24,43–53,64% e em 14,32–50,87%, respectivamente. O BSC atinge solução em menos tempo, sem perda da qualidade do GQI. O BSC é generalizável e efetivo em reduzir o tempo de processamento das recomendações de acasalamento.

Termos para indexação: algoritmo, melhoramento animal, suporte à decisão, sistemas de acasalamento.

Ezequiel Luís Vidal[(1 ✉)] (iD),
Sandro da Silva Camargo[(1)] (iD) and
Fernando Flores Cardoso[(2)] (iD)

[(1)] Universidade Federal do Pampa, Programa de Pós-Graduação em Computação Aplicada, Avenida Maria Anunciação Gomes de Godoy, nº 1.650, Malafaia, CEP 96413-172 Bagé, RS, Brazil.
E-mail: ezequiel_vidal@outlook.com, sandrocamargo@unipampa.edu.br

[(2)] Embrapa Pecuária Sul, Rodovia BR-153, Km 632,9, Vila Industrial, Zona Rural, Caixa Postal 242, CEP 96401-970 Bagé, RS, Brazil.
E-mail: fernando.cardoso@embrapa.br

✉ Corresponding author

Introduction

Population growth worldwide has been increasing the demand for livestock products (Fukase & Martin, 2020), which requires the development of new strategies to increase food production, while adding quality value and promoting sustainability (Herrero & Thornton, 2013). A strategy that stands out are genetic improvement software used for advances in livestock genetic traits (Malhado et al., 2010), which allows of increasing animal productivity and quality through mating selection (Miglior et al., 2017).

In Brazil, PampaPlus is the genetic improvement software used in the beef cattle breeding program of the country's association of Hereford and Braford breeds, Associação Brasileira de Hereford e Braford. In a partnership with Embrapa Pecuária Sul, the software collects and analyzes the genetic performance of herds, with animal genetic traits expressed as expected progeny difference (EPD) and weighed through the genetic qualification index (GQI) used to guide the selection of semen available in the program or of embryo by allocating dams to specific sires (Costa et al., 2017; Fontoura et al., 2019). A single GQI value is assigned to each animal.

As other techniques that adopt genetic algorithms, PampaPlus has been using evolutionary computing for mating selection (Storn & Price, 1997; Carvalheiro et al., 2010; Kinghorn, 2011; Barreto Neto, 2014; Henryon et al., 2019). According to Fontoura et al. (2019), the software maximizes the offspring's GQI value and minimizes inbreeding rates, with each iteration of the genetic algorithm presenting different mating combinations. The same authors highlighted that the inputs of the used algorithm are the EPDs of selected animals, the maximum desired inbreeding rate, and the number of mates per sire, whereas the output, when optimization is feasible, is a set of mating pairs that meets the given restrictions and presents the maximum mean GQI possible. The breeder can customize restrictions, but cannot change genetic algorithm internal parameters such as number of chromosomes, mutation rate, selection methods for genes, or penalty weights. The main limitation of the PampaPlus software is the convergence time of genetic algorithms, which is why it is important to use the stopping criterion parameter that directly affects the amount of time to compute a solution (Fontoura et al., 2019).

The objective of this work was to propose a new stopping criterion to shorten the computing time of the PampaPlus genetic improvement software, while maximizing the GQI of the progeny, controlling inbreeding, and avoiding unintended culling.

Materials and Methods

The PampaPlus database described in Fontoura et al. (2019) was used in the present study. The data collected on the animals available for mating were: breeder identification number, animal identification number, and EPDs. The experiments were carried out in the five following scenarios (farms), using data from three different herds: farm 1, with 37 sires and 568 dams; farm 2, with 17 sires and 148 dams; farm 3, with 48 sires and 258 dams; farm 4, with a random selection of 25% of the animals from farm 1; and farm 5, with a random selection of 50% of the animals from farm 1. As suggested by Bouthillier et al. (2021), these different scenarios allowed of verifying the robustness of the proposed approaches.

The following EPDs were used: total maternal gain (TM), post-weaning gain (PWG), yearling weight (YW), muscling score (MSC), height score (HSC), and scrotal circumference (SC). The respective weights of these EPDs in the GQI of the PampaPlus software were:

GQI = 30% TM + 15% PWG + 15% YW + 12.5% MSC + 12.5% HSC + 15% SC

Some breeder-defined restrictions were adopted: sires breeding up to 30 dams, minimum amount of matings per sire set to 0, and maximum inbreeding set to a default value of 3.0%. The penalties were calculated based on the standard deviations from the herd average EPD towards the unfavorable direction, whereas the mating GQI was penalized proportionally to the deviations, which were set at 20%. The penalties were applied to a single trait or a set of critical traits, which may or may not be in the GQI. In addition, mates that exceeded the maximum allowed inbreeding were considered invalid and their GQI was not computed.

Two stopping criteria were investigated and compared with the original one of 1,000 iterations used in the PampaPlus software: the saturation stopping criterion (SSC) and Bhandari's stopping criterion (BSC) proposed by Yeng et al. (2019) and Bhandari et al. (2012), respectively.

According to Yeng et al. (2019), the SSC is based on the assumption that population fitness variance decreases as the genetic algorithm converges. The authors pointed out that this stopping criterion compares the fitness variance value of each generation, computed as a function of population size (PS), fitness of each chromosome (F_i), and average fitness of the population (F), with a predefined threshold in order to stop the execution of the genetic algorithm. The threshold is usually close to zero, but should be tested in order to define the best one depending on the specific problem to be solved and the used data. The SSC is determined through the following inequality equation:

$$SSC = \left((1/PS) \times \sum_{i=1}^{PS} (F_i - F)^2 \right) < threshold$$

SSC was used in farms 1 and 2, each subjected to seven tests (ST1 to ST7): ST1, a control test using the original stopping criterion of 1,000 iterations; and ST2 to ST7, six tests, each with a different threshold. The upper bound used for defining threshold values was 0.03, obtained through the equation proposed by Yeng et al. (2019) for this purpose.

BSC is based on the variation of the best fitness values obtained over generations, interrupting the execution of the genetic algorithm if the increase in the fitness of the best chromosome after n generations is below the determined threshold (Bhandari et al., 2012). This stopping criterion was calculated using the generation interval (n), the best fitness of each generation (BF_i), and the best fitness average for the last n generations (BF), as follows:

$$BSC = \left((1/n) \times \sum_{i=1}^{n} (BF_i - BF)^2 \right) < threshold$$

BSC was used in farms 1, 2, 3, 4, and 5, each subjected to ten tests (BT1 to BT10): BT1, a control test using the original stopping criterion of 1,000 iterations; and BT2 to BT10, nine tests combining three different thresholds and three different generation intervals. The setup described by Bhandari et al. (2012) of 10^{-5} and 10^{-4} as thresholds and n = 200 as the generation interval was used as the starting point in the present study. The thresholds of 10^{-3}, 10^{-2}, and 10^{-1} were tested here, and the values of 300, 200, and 100 were evaluated for the generation interval parameter.

The tests for each scenario were performed in a computer with a 2.5GHz Intel Core i5-7200U processor, 8GB RAM, and 1TB hard disk, with a Linux-based operating system, with an Ubuntu distribution, version 16.04, with 64 bits.

The two following metrics were collected from all tests: elapsed processing time of the genetic algorithm and best chromosome fitness. Since the genetic algorithm has a stochastic component, i.e., the sire is randomly selected for each dam, each test was run ten times. The one-way analysis of variance (ANOVA) was performed to verify differences in processing time and chromosome fitness between each stopping criterion. ANOVA assumptions were checked through Shapiro's test for normality, Levene's test for homogeneity of variances, and Durbin-Watson's test for independence of errors. Mean differences were evaluated by Tukey's test, which was carried out using the Agricolae package, version 1.3-3 (De Mendiburu, 2020).

PampaPlus mating algorithm was implemented using the R software, version 3.6.2 (R Core Team, 2021). Its database was accessed through the RMySQL package, version 0.10.18 (Ooms et al., 2019), and managed with MySQL, version 5.7.28 (MySQL, 2019). The RStudio Integrated Development Environment, version 1.2.1335, was used as the user interface (RStudio Team, 2020). The RCPP package, version 1.0.7, was used to boost the performance of processor-intensive functions (Eddelbuettel & Balamuta, 2018), whereas the Plotly package, version 4.9.1, was used for data visualization (Sievert, 2020).

Results and Discussion

Regarding SCC, in tests ST2 to ST7 in farms 1 and 2, the processing time decreased as expected when the threshold value was increased from 0.01 to 0.03 (Table 1). The algorithm became less restrictive as the threshold was increased, with a noticeable exponential growth in processing time due to dataset size. The elapsed processing time was a few minutes for farm 1 and less than 1 min for farm 2.

In farm 1, the fitness averages of ST2 to ST5 did not differ significantly from that of ST1, the control. Under a threshold of 0.025, ST5 was the only one that reduced processing time in 11.5% compared with ST1 (Table 1). Since the genetic algorithm usually converges before 1,000 iterations (Fontoura et al., 2019), this limit was overestimated to avoid a premature stopping. After ten runs, ST1 always stopped at 1,000 iterations,

whether the algorithm had converged or not, whereas ST5 stopped at 756 iterations, on average, that is, took less iterations than STI to reach the same fitness value (Figure 1).

In farm 2, ST2 and ST3 presented the same fitness averages as ST1. However, ST3, under a threshold of 0.02, was the only one that decreased processing time in 35.13% compared with ST1 (Table 1). In this scenario, the genetic algorithm converged faster than in farm 1. ST3 performed 429 generations, on average, while ST4 to ST7 showed an even lower average numbers of generations, but with losses in fitness value.

Yeng et al. (2019) used SSC to solve a specific problem in a single-test scenario. However, in the present study, it was not possible to define a unique threshold value for the different evaluation scenarios (Table 1). Therefore, SSC cannot be used as a usual stopping criterion in different datasets. Since SSC is scale dependent, it works differently depending on the signal and magnitude of the GQI. If the average performance of a farmer's herd is worse than that of the population, the GQI value will be negative; alternatively, if the herd's performance is better than the average, GQI will be positive and high.
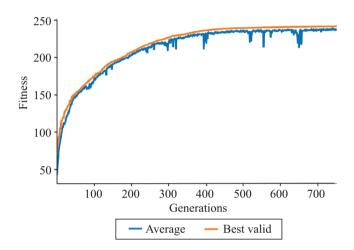
Table 1. Averages of processing time and fitness for each of the seven tests (ST1 to ST7) performed in two scenarios (farms) using the saturation stopping criterion for genetic algorithms of the PampaPlus software[1].

Test	Threshold	Farm 1[2]		Farm 2[2]	
		Fitness	Time (min)	Fitness	Time (min)
ST1	-	241.58a	18.6a	57.27a	1.0a
ST2	0.0100	241.59a	18.7a	57.18a	0.9a
ST3	0.0200	241.55a	18.4a	56.83ab	0.6b
ST4	0.0225	241.40a	18.0a	55.96bc	0.5c
ST5	0.0250	241.20a	16.4b	55.88bc	0.5c
ST6	0.0275	239.94b	14.5c	55.61c	0.5c
ST7	0.0300	239.34b	14.0c	55.34c	0.4c

[1]Means followed by equal letters do not differ by Tukey's test, at 5% probability. [2]Datasets: 37 sires and 568 dams in farm 1, and 17 sires and 148 dams in farm 2. Initial best fitness value: 75.54 in farm 1 and 9.86 in farm 2.

Figure 1. Evolution of the genetic algorithm in test ST5 in farm 1 using the saturation stopping criterion for genetic algorithms of the PampaPlus software.

Table 2. Averages of runtime and fitness test performed in five scenarios (farms) using Bhandari's stopping criterion for genetic algorithms of the PampaPlus software[1].

Test	Threshold	GI	Farm 1[2]		Farm 2[2]		Farm 3[2]		Farm 4[2]		Farm 5[2]	
			Fitness	Time (min)	Fitness	Time (min)	Fitness	Time (min)	Fitness	Time (min)	Fitness	Time (min)
BT1	-	-	241.58a	18.6a	57.27a	0.9a	-18.86ab	3.2bcd	-34.82a	0.8a	48.54a	3.3a
BT2	0.001	300	241.66a	18.3a	57.21a	0.8b	-18.74a	3.6a	-34.86a	0.6b	48.44a	2.9b
BT3	0.010	300	241.57a	18.1a	57.16ab	0.7c	-18.78a	3.4ab	-34.85a	0.5c	48.52a	2.7c
BT4	0.100	300	241.55a	16.9b	57.23a	0.7d	-18.76a	3.1cde	-34.89a	0.5cd	48.48a	2.5cde
BT5	0.001	200	241.53a	17.9a	57.18a	0.7c	-18.72a	3.6a	-34.87a	0.5c	48.54a	2.6cd
BT6	0.010	200	241.52a	16.9b	57.16ab	0.7d	-18.78a	3.3bc	-34.87a	0.5de	48.40a	2.4def
BT7	0.100	200	241.53a	16.1cd	57.19a	0.6e	-19.04ab	2.8ef	-34.86a	0.5ef	48.51a	2.3fg
BT8	0.001	100	241.54a	16.7bc	57.12ab	0.6d	-19.05ab	2.9def	-34.95a	0.5de	48.55a	2.3efg
BT9	0.010	100	241.49a	15.9d	57.10ab	0.6e	-19.44b	2.7fg	-34.91a	0.4fg	48.40a	2.2g
BT10	0.100	100	241.00b	14.7e	56.89b	0.5f	-20.40c	2.4g	-34.96a	0.4g	47.85b	1.9h

[1]Means followed by equal letters do not differ by Tukey's test, at 5% probability. [2]Datasets: 37 sires and 568 dams in farm 1; 17 sires and 148 dams in farm 2; 48 sires and 258 dams in farm 3; and random selection of 25 and 50% of the animals from farm 1 in farms 4 and 5, respectively. Initial best fitness: 75.54 in farm 1, 9.86 in farm 2, -132.47 in farm 3, -63.40 in farm 4, and -31.83 in farm 5. GI, generation interval.

Regarding BSC, BT2 to BT10 showed the same pattern in every scenario (Table 2). Furthermore, only the threshold of $\geq 10^{-3}$ stopped the algorithm before 1,000 generations. When generation intervals were decreased from 300 to 100, a decrease in elapsed processing time also occurred. For different generation intervals, processing time decreased as the threshold values increased in the range of 10^{-3} to 10^{-1}.

BT10 showed an average fitness value lower than those of ST1 in farms 1, 2, 3, and 5 (Table 2). In farms 3 and 4, negative fitness values were observed because the selected animals had a GQI lower than the average of the PampaPlus software.

In farm 1, BT4, BT6, BT7, BT8, and BT9 presented similar average fitness values and a shorter processing time than BT1, especially BT7 and BT9. In farm 2, BT2 to BT9 had similar fitness values, whereas BT7 and BT9 showed the shortest processing time. In farm 3, BT1 to BT8 did not differ for fitness, BT9 presented a fitness value similar to that of BT1, and BT7 and BT8 showed the shortest processing time. In farm 4, all tests presented similar fitness values and a reduced processing time, which was shorter for BT9 and BT10. In farm 5, BT2 to BT9 presented similar fitness values, whereas BT7, BT8, and BT9 showed the shortest processing time.

As an overall result for BSC, only BT7 and BT9 presented similar fitness values and a shorter processing time in all scenarios when compared with
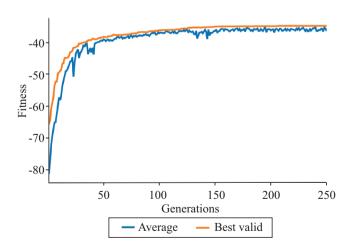
BT1. Moreover, under a 10^{-2} threshold value and a 100-generation span, BT9 showed the best processing time. In farms 1, 2, 3, 4, and 5, respectively, processing time showed reductions of 14.32, 36.50, 15.87, 50.87, and 32.72% in comparison with the control.

Comparing the SSC and BSC stopping criteria, the number of iterations in farms 1 and 2 was very similar. Considering only BT9, the test with the best results, the average number of iterations was 713 for farm 1, 429 for farm 2, 767 for farm 3, 256 for farm 4 (Figure 2), and 487 for farm 5.

Although SSC and BSC showed a similar number of iterations, the performance of SSC was affected since it was not possible to define a set of parameters for this stopping criterion that could be successfully used in different datasets. Therefore, BSC is more advantageous than SSC because it can be used to reduce processing time without losses in fitness values and the breeder will not need to adjust stopping criterion parameters to obtain mating recommendations for each evaluation scenario. In summary, depending on the amount of sires and dams, a suitable adaptive stopping criterion allows of a significant gain in performance when compared with the criterion of a fixed number of iterations.

Conclusions

1. Bhandari's stopping criterion presents the best processing time, which is 14.32 to 50.87% shorter than that of the original criterion of 1,000 iterations, without losses in the genetic qualification index of the PampaPlus software.

2. Although the saturation stopping criterion shows a processing time 24.43 to 53.64% shorter than that of the original criterion, a different threshold must be determined for each dataset.

Acknowledgments

Figure 2. Evolution of the genetic algorithm of the BT9 test in farm 4 using Bhandari's stopping criterion for genetic algorithms of the PampaPlus software.

References

BARRETO NETO, A.D. **Estrutura populacional e otimização de esquemas de acasalamento em ovinos com uso de algoritmos evolucionários**. 2014. 62p. Dissertação (Mestrado) – Universidade Federal de Sergipe, São Cristóvão.

BHANDARI, D.; MURTHY, C.A.; PAL, S.K. Variance as a stopping criterion for genetic algorithms with elitist model. **Fundamenta Informaticae**, v.120, p.145-164, 2012.

BOUTHILLIER, X.; DELAUNAY, P.; BRONZI, M.; TROFIMOV, A.; NICHYPORUK, B.; SZETO, J.; SEPAH, N.; RAFF, E.; MADAN, K.; VOLETI, V.; KAHOU, S.E.; MICHALSKI, V.; SERDYUK, D.; ARBEL, T.; PAL, C.; VAROQUAUX, G.; VINCENT, P. Accounting for variance in machine learning benchmarks. In: MACHINE LEARNING AND SYSTEMS CONFERENCE, 3., 2021, Austin. **Proceedings**. [S.l.: s.n.], 2021. p.747-769. DOI: https://doi.org/10.48550/arXiv.2103.03098.

CARVALHEIRO, R.; QUEIROZ, S.A. de.; KINGHORN, B. Optimum contribution selection using differential evolution. **Revista Brasileira de Zootecnia**, v.39, p.1429-1436, 2010. DOI: https://doi.org/10.1590/S1516-35982010000700005.

COSTA, R.F.; TEIXEIRA, B.B.M.; YOKOO, M.J.; CARDOSO, F.F. Economic selection indexes for Hereford and Braford cattle raised in southern Brazil. **Journal of Animal Science**, v.95, p.2825-2837, 2017. DOI: https://doi.org/10.2527/jas.2016.1314.

DE MENDIBURU, F. **Agricolae**: Statistical Procedures for Agricultural Research. R package version 1.3-3. 2020. Available at: <https://cran.r-project.org/src/contrib/Archive/agricolae/>. Acessed on: June 12 2021.

EDDELBUETTEL, D.; BALAMUTA, J.J. Extending R with C++: A Brief Introduction to Rcpp. **The American Statistician**, v.72, p.28-36, 2018. DOI: https://doi.org/10.1080/00031305.2017.1375990.

FONTOURA, D. de C.N. da; CAMARGO, S. da S.; TORRES JUNIOR, R.A. de A.; CARVALHO, H.G. de; CARDOSO, F.F. Optimizing mate selection: a genetic algorithms approach. In: ICAR CONFERENCE, 43., 2019, Prague. **Proceedings**. Rome: ICAR, 2019. p.55-62.

FUKASE, E.; MARTIN, W. Economic growth, convergence, and world food demand and supply. **World Development**, v.132, art.104954, 2020. DOI: https://doi.org/10.1016/j.worlddev.2020.104954.

HENRYON, M.; LIU, H.; BERG, P.; SU, G.; NIELSEN, H.M.; GEBREGIWERGIS, G.T.; SØRENSEN, A.C. Pedigree relationships to control inbreeding in optimum-contribution selection realise more genetic gain than genomic relationships. **Genetics Selection Evolution**, v.51, art.39, 2019. DOI: https://doi.org/10.1186/s12711-019-0475-5.

HERRERO, M.; THORNTON, P.K. Livestock and global change: emerging issues for sustainable food systems. **PNAS**, v.110, p.20878-20881, 2013. DOI: https://doi.org/10.1073/pnas.1321844111.

KINGHORN, B.P. An algorithm for efficient constrained mate selection. **Genetics Selection Evolution**, v.43, art.4, 2011. DOI: https://doi.org/10.1186/1297-9686-43-4.

MALHADO, C.H.M.; CARNEIRO, P.L.S.; MALHADO, A.C.M.; MARTINS FILHO, R.; BOZZI, R.; LADLE, R.J. Genetic improvement and population structure of the Nelore breed in Northern Brazil. **Pesquisa Agropecuária Brasileira**, v.45, p.1109-1116, 2010. DOI: https://doi.org/10.1590/S0100-204X2010001000009.

MIGLIOR, F.; FLEMING, A.; MALCHIODI, F.; BRITO, L.F.; MARTIN, P.; BAES, C.F. A 100-year review: identification and genetic selection of economically important traits in dairy cattle. **Journal of Dairy Science**, v.100, p.10251-10271, 2017. DOI: https://doi.org/10.3168/jds.2017-12968.

MYSQL. **MySQL Release Notes**: Changes in MySQL 5.7.28 (2019-10-14, General Availability). 2019. Available at: <https://dev.mysql.com/doc/relnotes/mysql/5.7/en/news-5-7-28.html>. Acessed on: June 23 2021.

OOMS, J.; JAMES, D.; DEBROY, S.; WICKHAM, H.; HORNER, J. **RMySQL**: Database Interface and 'MySQL' Driver for R. R package version 0.10.18. 2019. Available at: <https://cran.r-project.org/src/contrib/Archive/RMySQL/>. Acessed on: June 14 2021.

R CORE TEAM. **R**: a language and environment for statistical computing. Vienna: R Foundation for Statistical Computing, 2021.

RSTUDIO TEAM. **RStudio**: Integrated Development for R. Boston, 2020. Available at: <http://www.rstudio.com/>. Acessed on: May 14 2021.

SIEVERT, C. **Interactive web-based data visualization with R, plotly, and shiny**. Florida: CRC Press, 2020. Available at: <https://plotly-r.com>. Acessed on: May 6 2021.

STORN, R.; PRICE, K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. **Journal of Global Optimization**, v.11, p.341-359, 1997. DOI: https://doi.org/10.1023/A:1008202821328.

YENG, F.F.; YOKE, S.K.; SUHAIMI, A. The saturation of population fitness as a stopping criterion in genetic algorithm. **International Journal of Electrical and Computer Engineering**, v.9, p.4130-4137, 2019. DOI: http://doi.org/10.11591/ijece.v9i5.pp4130-4137.