



## **BrangusSelection: um algoritmo para seleção ótima de acasalamentos com índice de seleção customizável**

**Bruno Ernesto Techera da Motta<sup>1</sup>, Ana Paula Lüdtke Ferreira<sup>1</sup>,  
Marcos Jun-Iti Yokoo<sup>2</sup>**

<sup>1</sup>Universidade Federal do Pampa - UNIPAMPA

<sup>2</sup>EMBRAPA Pecuária Sul

{brunomotta.aluno, anaferreira}@unipampa.edu.br  
marcos.yokoo@embrapa.br

**Abstract.** *This work presents the BrangusSelection algorithm, which produces an optimal mating scheme, maximizing the expected value of the herd's next generation. Tests with the algorithm show a polynomial runtime in the number of animals received as input, which indicates that the problem is not computationally intractable. The algorithm depends on a selection index built from the animals' selection objectives/criteria and can operate on different animal species and selection indexes.*

**Resumo.** *Este trabalho apresenta o algoritmo BrangusSelection, que produz um esquema ótimo de acasalamentos, maximizando o valor esperado da próxima geração do rebanho. Os testes com o algoritmo mostram um tempo de execução polinomial na quantidade de animais recebida como entrada, o que indica que o problema não é intratável computacionalmente. O algoritmo depende da existência de um índice de seleção, construído a partir dos objetivos e critérios de seleção dos animais e pode operar sobre qualquer espécie de animal e índice de seleção.*

### **1. Introdução**

O crescimento da população demanda aumento da produção alimentos sem, contudo, o aumento do uso de recursos naturais, que já estão no seu limite [FAO 2009]. Uma das estratégias para o aumento da produção é melhorar o potencial genético dos rebanhos. O melhoramento genético animal consiste no direcionamento da genética das novas gerações para o aumento da saúde e das características dos animais que se traduzam em ganhos para o sistema de produção. As principais estratégias são os sistemas de seleção e de acasalamentos [Eler 2017a, Cardoso 2009]. A seleção escolhe os animais que se tornarão

pais da próxima geração e o sistema de acasalamento determina quem são os pares de animais que devem reproduzir. Os pais passarão sua superioridade genética para a prole, minimizando características indesejáveis e perpetuando as vantajosas [Nieto et al. 2013].

Um *objetivo de seleção* é uma característica desejável que se quer obter no futuro rebanho, que pode compreender características genéticas ou econômicas [Simões et al. 2020]. Os *critérios de seleção* são as características pelas quais os animais são avaliados para que os objetivos de seleção sejam atingidos [Abreu et al. 2013]. Critérios de seleção podem ser combinados, gerando um *índice de seleção* que expressa, em um único valor, o mérito de cada animal [Eler 2017b]. Índices de seleção são utilizados pelos diversos programas de melhoramento genético, visto que facilitam a decisão de seleção dos reprodutores [Cardoso 2009]. Dados os animais selecionados para reprodução, o *problema de maximização de escolha de acasalamentos* visa encontrar o melhor esquema de pareamento de machos e matrizes, objetivando o maior índice possível da prole.

Os trabalhos relacionados encontrados na literatura fazem uso majoritário de algoritmos genéticos, meta-heurística apropriada para problemas para os quais não se conhece algoritmo eficiente. Cita-se Fontoura et al. (2020), que busca a otimização de acasalamentos com base nas *diferenças esperadas de progênie* (DEP); Carvalho et al. (2016) apresenta um sistema com o objetivo de maximizar o ganho de peso e minimizar a endogamia do rebanho caprino em estudo e, para isso, faz uso de algoritmos genéticos também com uso de DEP; Barreto et al. (2014) otimizou esquemas de acasalamento em ovinos, também com abordagem genética com valores estimados por DEP-BLUP e também com o objetivo de minimizar a consanguinidade; Carvalheiro et al. (2010) usa algoritmos genéticos para seleção de animais para reprodução também visando a diminuição da coancestralidade da nova geração do rebanho. Nenhum dos trabalhos citados investiga a complexidade do problema de encontrar o melhor esquema possível de acasalamentos, ainda que alguns afirmem que o problema é computacionalmente intratável, sem referências ou prova para essa afirmação. O argumento de que o espaço de estados tem um tamanho exponencial no número de animais envolvidos e que considerá-lo por inteiro é inviável é tautológica, mas não informa sobre a real complexidade do problema. Um dos objetivos deste trabalho foi fazer essa investigação, na busca de um algoritmo ótimo com tempo de execução polinomial no número de animais envolvidos.

Neste texto, é apresentado o algoritmo BrangusSelection, que usa uma estratégia que combina programação linear e o método *branch and bound* para encontrar a solução ótima do problema e estabelecer diferentes cenários para investigar a eficiência com a qual o problema é resolvido. O índice bioeconômico desenvolvido em [Simões et al. 2020], com adição de variáveis correspondentes ao temperamento, será usado para apresentação dos resultados. No entanto, o algoritmo proposto é genérico e pode ser executado com qualquer índice.

O restante do texto está assim organizado: a Seção 2 apresenta a caracterização do problema, o modelo do algoritmo e o ferramental tecnológico usado para a construção da solução; a Seção 3 discute os resultados obtidos; a Seção 4 apresenta as considerações finais e uma discussão sobre a complexidade do problema de otimização dos acasalamentos.

## 2. Caracterização e modelo do problema

O problema de maximização de escolha de acasalamentos procura otimizar as opções de acasalamento de acordo com o índice de seleção escolhido, objetivando o maior índice possível da prole, calculado como a média dos índices dos progenitores. O processo de seleção dos animais que vão acasalar é feito previamente e está fora do escopo deste trabalho. O índice de seleção consiste em um conjunto de características (desejadas ou não) que podem ocorrer nos animais, com um valor de ponderação sobre essas características. Formalmente, um *índice de seleção* é um par  $\mathcal{I} = (T, w)$  em que  $T$  é o conjunto de características consideradas no índice e  $w : T \rightarrow \mathbb{R}$  é a função de ponderação do índice que designa o peso relativo de cada característica. Não há restrições quanto aos valores que a ponderação das características e nem os valores precisam estar normalizados. Valores negativos são aceitáveis, visto que eles representam penalidades, no sentido de que um valor alto para a característica correspondente torna o valor do índice do animal menor.

O objetivo do índice de seleção é estimar a contribuição individual de cada animal para a próxima geração do rebanho, que é a soma do valor de suas características ponderadas pela importância relativa de cada uma delas. Se  $A$  é um conjunto de animais,  $\mathcal{I} = (T, w)$  um índice de seleção e  $\nu : T \times A \rightarrow \mathbb{R}$  a função que retorna o valor de uma característica para cada animal, a *contribuição individual*  $\iota^{\mathcal{I}}$  de um animal  $a \in A$  é

$$\iota^{\mathcal{I}}(a) = \sum_{k \in T} \nu(k, a) \cdot w(k) \quad (1)$$

quando o índice de seleção é entendido a partir do contexto, escreve-se  $\iota(a)$  em vez de  $\iota^{\mathcal{I}}(a)$ .

O valor de um acasalamento é a média da contribuição individual dos progenitores, a menos que o grau de endogamia  $r : A \times A \rightarrow \mathbb{R}^+$  supere o limite máximo  $l$  definido pelo usuário; nesse caso, é atribuído à contribuição o valor  $-\infty$  (menos infinito) para excluir a possibilidade dessa opção de acasalamento compor a solução final. O algoritmo pode ser transformado para adicionar punições menos drásticas; aqui, assume-se que se trata de uma restrição. Logo, pares com nível de endogamia acima do aceitável não podem mesmo fazer parte da solução sem que a restrição seja violada. Formalmente, a contribuição do acasalamento do par  $(m \in M, f \in F)$  é dado por

$$\pi(m, f) = \begin{cases} (\iota^{\mathcal{I}}(m) + \iota^{\mathcal{I}}(f))/2 & r(m, f) \leq l \\ -\infty & r(m, f) > l \end{cases} \quad (2)$$

onde  $\iota^{\mathcal{I}}$  é a contribuição individual de cada animal, dado um índice de seleção  $\mathcal{I}$ . O problema de maximização de seleção de acasalamento é apresentado a seguir.

### Problema 1 (Problema de maximização de seleção de acasalamentos)

**Entrada:** uma matriz de contribuição  $\Pi_{|M| \times |F|}$  em que cada  $\pi_{ij} = \pi(i, j)$  (equação 2) e uma função  $t : M \rightarrow \mathbb{N}$  que indica o limite máximo de utilização de cada macho no processo de cruzamento.

**Saída:** uma matriz binária  $\mathcal{B}^*$ , com dimensão  $|M| \times |F|$  onde cada elemento  $b_{ij}^*$  de  $\mathcal{B}^*$  é

$$b_{ij}^* = \begin{cases} 1 & \text{se o macho } i \text{ e a fêmea } j \text{ devem acasalar} \\ 0 & \text{caso contrário} \end{cases}$$

respeitando as seguintes restrições:

$$\begin{aligned} \sum_{i=1}^{|M|} b_{ij}^* &= 1, \text{ para cada fêmea } j \\ \sum_{j=1}^{|F|} b_{ij}^* &\leq t_i, \text{ para cada macho } i \text{ e limite de reprodução individual } t_i \end{aligned}$$

e de modo que, para qualquer outra matriz binária  $B$  com as mesmas características, temos que

$$\sum_{i=1}^{|M|} \sum_{j=1}^{|F|} \pi_{ij} b_{ij} \leq \sum_{i=1}^{|M|} \sum_{j=1}^{|F|} \pi_{ij} b_{ij}^*$$

Resumidamente, a entrada do Problema 1 é uma matriz  $\Pi$  com dimensão  $|M| \times |F|$  e a saída é uma matriz binária  $\mathcal{B}^*$  com dimensão  $|M| \times |F|$ . Cada elemento da matriz binária  $\mathcal{B}^*$  deverá ser 1, indicando que o par deve acasalar, ou 0, indicando que o par em questão não faz parte da solução do problema. Em relação às restrições, cada fêmea  $j$  deve ser coberta por um único touro ( $\sum_{i \in M} b_{ij}^* = 1$ ) e o número de vezes que um macho  $i$  é usado não excede sua utilização máxima ( $\sum_{j \in F} b_{ij}^* \leq t(i)$ ).

O problema da maximização pode ser caracterizado como um problema de otimização em virtude de sua saída ser a melhor solução dentre um conjunto de soluções possíveis, no que é denominado espaço de estados das soluções [Cormen et al. 2002]. Como a saída do problema é uma função inteira com domínio em  $F$  e contradomínio em  $M$ , pode-se modelar o problema como uma instância de programação linear inteira [Hillier e Lieberman 2013], cuja função objetivo pode ser expressa como

$$\text{MAX} \sum_{i=1}^{|M|} \sum_{j=1}^{|F|} \pi_{ij} c_{ij} \quad (3)$$

estando seu valor sujeito às seguintes restrições:

$$\begin{aligned} \forall j \in F \quad \sum_{i=1}^{|M|} c_{ij} &= 1 \\ \forall i \in M \quad \sum_{j=1}^{|F|} c_{ij} &\leq t(i) \end{aligned} \quad (4)$$

Note-se que a função objetivo é uma função linear, visto que o valor da contribuição de cada par  $(i, j) \in M \times F$  é um valor constante. As variáveis do problema são os valores  $c_{ij}$ , que podem tomar valores no conjunto  $\{0, 1\}$ . O modelo construído, dados os conjuntos de machos  $M$  e de fêmeas  $F$ , gera uma função objetivo com  $|F| \cdot |M|$  termos e um conjunto de  $|F| + |M|$  restrições.

A estratégia de resolução do problema de programação inteira por meio da combinação das técnicas de programação linear e *branch and bound* consiste no relaxamento do problema de programação inteira em um problema de programação linear em que as variáveis podem ter valores reais. O método (polinomial) simplex [Hillier e Lieberman 2013] pode ser usado para resolver o problema linear, que contém todas as soluções inteiras do problema. O algoritmo *branch and bound* investiga as soluções inteiras dentro do espaço de soluções, eliminando aquelas que não podem retornar um valor maior do que a melhor solução encontrada até o momento. Essa estratégia permite que o espaço de soluções não precise ser todo investigado. Problemas NP-completos

ainda geram uma árvore de tamanho proporcional a uma função exponencial no número de variáveis, mesmo que nem todo o espaço seja investigado. A análise da solução construída e dos resultados da técnica apontaram para a existência de um algoritmo de complexidade polinomial, como será discutido ao final do trabalho.

### 3. Resultados e discussão

O algoritmo foi desenvolvido e testado em um computador com processador Intel Core i7-6500U de 2.50GHz; 12GB de memória RAM; Armazenamento SSD de 500GB. O sistema operacional usado foi a distribuição Linux Ubuntu 20.04 LTS 64-bit. Para a implementação do algoritmo foi utilizada a linguagem de programação R [Ihaka e Gentleman 1996], versão 4.0.5. A IDE usada foi o RStudio versão 1.4. Os pacotes R lpSolve [Berkelaar et al. 2020], lpsymphony [Kim 2020] e Rcpp [Eddelbuettel e Balamuta 2018] foram usados solução do modelo de programação linear, *branch and bound* e integração com C++.

A base de dados dos animais utilizada neste trabalho é oriunda da EMBRAPA, em formato de planilha eletrônica, transformada em arquivo *.csv* e importada para o ambiente RStudio. A base de dados contém dados referentes a 1366 bovinos da raça *Brangus*, sendo 511 machos e 855 fêmeas. Os dados correspondem aos valores de DEP aprimorada pela Genômica (DEPG) das características dos animais que compõem o índice de seleção bioeconômico construído em [Simões et al. 2020], acrescido da variável de temperamento. O algoritmo pode, contudo, ser executado para qualquer índice, visto que trabalha somente sobre a matriz de contribuição. A complexidade do cálculo dos valores do índice para cada animal é constante, dado um índice fixo, visto que o valor de uma característica de um animal pode ser buscada em tempo constante em uma tabela indexada. Ambas as tabelas de DEPG e consanguinidade são indexadas. Após esse cálculo, que leva à construção da matriz de contribuição, o algoritmo só executa sobre a matriz de contribuições.

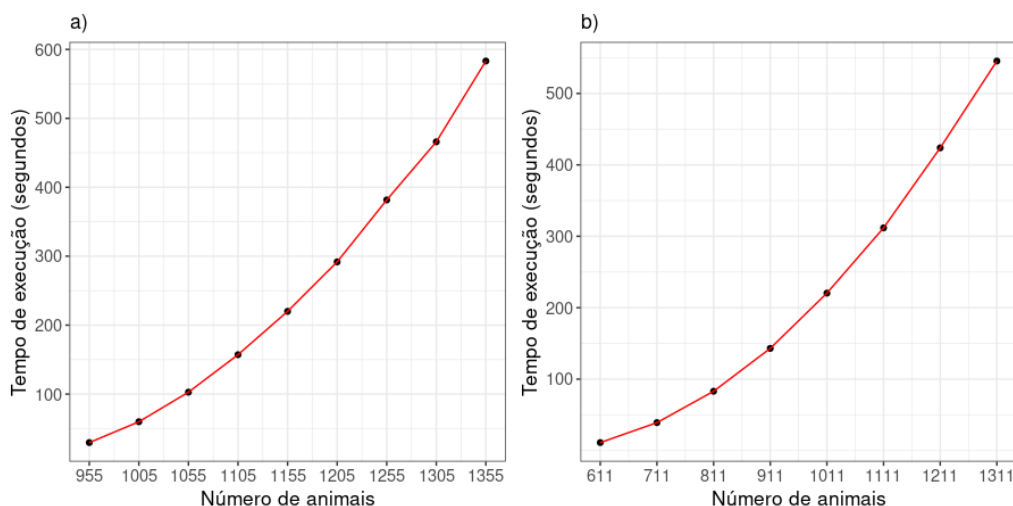
Os testes objetivaram verificar como a função do tempo de execução cresce na medida que o tamanho da entrada (número de machos e fêmeas) aumenta. Para a execução dos testes, aumentou-se gradualmente o número de animais medindo-se o tempo de execução do algoritmo. A matriz de contribuição foi construída de forma que os machos e fêmeas estejam ordenados por índice de contribuição. Essa ordenação fez com que o algoritmo executasse mais rapidamente (da ordem de minutos na configuração dada acima). As consequências desse fato serão discutidas na conclusão deste trabalho.

Para o teste, foram configuradas 9 instâncias, que são determinadas pelo número de machos, número de fêmeas e uso máximo dos touros. Cada instância foi executada 10 vezes, utilizando em cada execução uma amostra diferente da base de dados. O tempo médio, variância e desvio padrão de cada instância executada foram calculados. A Tabela 1 mostra os resultados do teste. As colunas indicam o número da instância, número de machos, número de fêmeas, uso máximo de cada touro, tempo médio de execução e suas variância e desvio padrão, respectivamente. Nesse experimento, fixou-se o número de fêmeas (total das fêmeas da base de dados) e aumentou-se o número de machos progressivamente. O valor do uso máximo de cada touro foi definido como o menor número possível para cobrir o total de vacas. Na instância 9, foram utilizados praticamente todos os animais da base de dados, na qual o BrangusSelection utilizou em média 583,11 segundos (aproximadamente 10 minutos) para encontrar a solução ótima.

**Tabela 1. Teste do BrangusSelection com incremento dos machos**

$n$	$ M $	$ F $	max	tempo médio	$\sigma^2$	$\sigma$
1	100	855	9	29,6 s	0,08	0,29
2	150	855	6	59,9 s	0,19	0,43
3	200	855	5	102,7 s	0,44	0,67
4	250	855	4	157,1 s	1,43	1,19
5	300	855	3	220,0 s	5,13	2,26
6	350	855	3	291,7 s	7,19	2,68
7	400	855	3	381,6 s	4,90	2,21
8	450	855	3	465,9 s	19,49	4,41
9	500	855	3	583,11 s	3,19	1,78

Um teste similar, para observar como o tempo de execução varia ao aumentar o número de fêmeas com um número fixo de machos, foi realizado. O comportamento foi equivalente aos dados apresentados na Tabela 1. A Figura 1a) apresenta a relação do tempo de execução e número de animais da Tabela 1 e a Figura 1b) mostra a relação do tempo de execução e número de animais correspondendo ao aumento do número de fêmeas. Em ambos os gráficos é possível observar a similaridade na curvatura dos tempos de execução em relação ao tamanho das entradas em ambas variações do teste. Da mesma forma, pode-se verificar que tanto os dados da tabela quanto a forma do gráfico apontam para uma função polinomial.



**Figura 1. Gráficos dos resultados dos testes realizados com o BrangusSelection.**

O formato da função do tempo de execução indicar uma complexidade polinomial tem implicações importantes. A complexidade do problema da programação linear é  $O(nm)$  em que  $n$  é o número de variáveis e  $m$  é o número de restrições do problema [Daguspta et al. 2006]. No caso do BrangusSelection, visto que o número de variáveis do problema é  $|M| \cdot |F|$  e o número de restrições é  $|M| + |F|$  (cf. Seção 2), a complexidade é da ordem de  $O(|M| \cdot |F| \cdot (|M| + |F|))$ , ou seja, quadrática no número de animais. Se o algoritmo *branch and bound* não está influenciando no aumento da complexidade, é uma pista importante de que o problema pode ter algoritmo ótimo mais eficiente que o resolva,

dispensando abordagens com meta-heurísticas que não garantem uma solução ótima ao final de sua execução.

A saída do algoritmo é uma tabela com a recomendação dos acasalamentos, já sumarizada a partir da matriz binária de saída do algoritmo (cf. definição do Problema 1). No arquivo de entrada, cada animal é identificado com um código, que é usado para apresentação do resultado, também em arquivo com formato csv. A Tabela 2 exibe um exemplo de recomendação de acasalamentos, obtida a partir da saída do programa. O valor da função objetivo, que determina o valor médio esperado da próxima geração, também é apresentado.

**Tabela 2. Exemplo de recomendação de acasalamentos**

<b>Touro</b>	<b>Vacas</b>			
L102	K482	K477	K500	I252
L840	L343	M76	L879	L363
M122	L372	M313	M212	M29
M175	M134	K407	K369	K331
M243	K554	M53	K418	M121
Valor função objetivo: 278,2408				

#### 4. Conclusão

Este trabalho apresentou o algoritmo BrangusSelection, com o objetivo de produzir uma estratégia ótima de acasalamentos entre animais selecionados para reprodução. O algoritmo depende da existência de um índice de seleção, mas funciona independente de que índice é esse, que pode variar em relação às espécies, raças e objetivos de seleção. O exemplo usado envolveu gado bovino da raça Brangus, mas o algoritmo é genérico e funciona para qualquer espécie de animais.

A utilização do algoritmo depende a existência de duas bases de dados: dos valores individuais dos animais para cada característica de interesse (ou os valores das contribuições individuais dos animais, já computados) e da matriz de coancestralidade (ou de consanguinidade) existente entre os animais selecionados para acasalamento. Programas de melhoramento animal mantém esses dados, frequentemente no formato de planilhas eletrônicas e, com menos frequência, em bancos de dados. De toda forma, a informação está disponível e pode ser transformada em arquivos csv, para entrada no algoritmo. Os demais parâmetros – número máximo de vezes que cada macho pode ser utilizado no processo de acasalamento à frente e índice máximo de consanguinidade aceitável, que pode ser zero – são passados diretamente ao algoritmo.

O BrangusSelection está correntemente codificado na linguagem de programação R, com pacotes para otimizar os laços iterativos por meio da inserção de código em C++, usando pacotes específicos para o algoritmo simplex e o método *branch and bound*. Todo o ferramental tecnológico usado é livre e pode ser utilizado imediatamente. Estamos em processo de disponibilização de um pacote R para esse fim específico e, no futuro, vislumbra-se a incorporação do algoritmo a uma API (*application programming interface*) para disponibilização do código como serviço.

Os resultados referentes à ordem de complexidade do problema são outra contribuição importante deste trabalho. Algoritmos genéticos e outras meta-heurísticas são valiosas quando o problema a ser resolvido é computacionalmente intratável. O indicativo da execução deste trabalho, com resultado ótimo produzido em um tempo da ordem de minutos para uma base com cerca de 1500 animais é compatível com uma função polinomial quadrática. Isso quer dizer que o problema da maximização de seleção de acasalamentos é computacionalmente tratável. A prova de que o problema pertence à classe de complexidade P já foi construída [Ferreira et al. 2021]. De fato, o problema pode ser resolvido em tempo  $O(|F| \cdot |M|)$ .

Como trabalhos futuros, prevê-se a análise da complexidade de outros sistemas de acasalamentos. Particularmente, de acasalamentos compensatórios, em que o objetivo, juntamente com a solução máxima da contribuição, é produzir um rebanho mais homogêneo, com menos animais excepcionais mas também com menos animais com deficiências em relação ao objetivo de seleção. Este trabalho também já está em andamento. Estratégias para diminuir a consanguinidade do rebanho a longo prazo começam a ser estudadas também.

## Referências

- Abreu, U. G. P. d., Sonohata, M. M., e Lopes, P. S. (2013). Definição de pesos econômicos e de índices de seleção para sistemas de produção. In Rosa, A. d. N., Martins, E. N., Menezes, G. R. d. O., e Silva, L. O. C. d., editors, *Melhoramento Genético Aplicado em Gado de Corte. Programa Geneplus-Embrapa*, volume 1, chapter 11. Embrapa Gado de Corte.
- Barreto Neto, A. D. et al. (2014). Estrutura populacional e otimização de esquemas de acasalamento em ovinos com uso de algoritmos evolucionários.
- Berkelaar, M. et al. (2020). lpsolve: Interface to 'lp\_solve' v. 5.5 to solve linear/integer programs. R package version 5.6.15.
- Cardoso, F. (2009). Ferramentas e estratégias para o melhoramento genético de bovinos de corte. *Embrapa Pecuária Sul-Documentos (INFOTECA-E)*.
- Carvalho, R., Queiroz, S. A. d., e Kinghorn, B. (2010). Optimum contribution selection using differential evolution. *Revista Brasileira de Zootecnia*, 39(7):1429–1436.
- Carvalho, T., Santos, N., Lira, W., Oliveira, P. A., Neto, P. S., Lindenberg, J., e Rabêlo, R. (2016). Um sistema de informação para melhoramento genético de caprinos e ovinos. In *Anais Principais do XII Simpósio Brasileiro de Sistemas de Informação*, pages 100–107. SBC.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., e Stein, C. (2002). Algoritmos: teoria e prática. *Editora Campus*, 2:296.
- Daguspta, S., Papadimitriou, C. H., e Vazirani, U. V. (2006). *Algorithms*. McGraw-Hill Higher Education, 1 edition.
- Edelbuettel, D. e Balamuta, J. J. (2018). Extending extitR with extitC++: A Brief Introduction to extitRcpp. *The American Statistician*, 72(1):28–36.



- Eler, J. P. (2017a). *Teorias e métodos em melhoramento genético animal: bases do melhoramento genético animal*. Faculdade de Zootecnia e Engenharia de Alimentos da USP.
- Eler, J. P. (2017b). *Teorias e métodos em melhoramento genético animal: seleção*. Faculdade de Zootecnia e Engenharia de Alimentos da USP.
- FAO (2009). Food and agriculture organization of the united nations (2009) how to feed the world 2050. Discussion paper prepared for Expert Forum: 12–13 October.
- Ferreira, A. P. L., Yokoo, M. J.-I., e Motta, B. E. T. (2021). On the problem of optimal mating in animal breeding. In *Anales de CLEI 2021*.
- Fontoura, D. C. N., Camargo, S. S., Jr, R. A. A. T., Carvalho, H. G., e Cardoso, F. F. (2020). Optimizing mate selection: a genetic algorithm approach. In *Proceedings of the ICAR Conference*, number 24 in ICAR Technical Series, Prague, CZ.
- Hillier, F. S. e Lieberman, G. J. (2013). *Introdução à pesquisa operacional*. McGraw Hill Brasil.
- Ihaka, R. e Gentleman, R. (1996). R: a language for data analysis and graphics. *Journal of computational and graphical statistics*, 5(3):299–314.
- Kim, V. (2020). lpsymphony: Symphony integer linear programming solver in r. <http://R-Forge.R-project.org/projects/rsymphony>, <https://projects.coin-or.org/SYMPHONY>, <http://www.coin-or.org/download/source/SYMPHONY/>.
- Nieto, L. M., Alencar, M. M. d., e Rosa, A. d. N. (2013). Critérios de seleção. In Rosa, A. d. N., Martins, E. N., Menezes, G. R. d. O., e Silva, L. O. C. d., editors, *Melhoramento Genético Aplicado em Gado de Corte. Programa Geneplus-Embrapa*, volume 1, chapter 10. Embrapa Gado de Corte.
- Simões, M. R. S., Leal, J. J. B., Minho, A. P., Gomes, C. C., MacNeil, M. D., Costa, R. F., Junqueira, V. S., Schmidt, P. I., Cardoso, F. F., Boligon, A. A., e Yokoo, M. J. (2020). Breeding objectives of brangus cattle in brazil. *Journal of Animal Breeding and Genetics*, 137(2):177–188.