COMPUTAÇÃO PARALELA APLICADA À SELEÇÃO GENÔMICA VIA INFERÊNCIA BAYESIANA

Marcos Rodrigues LAGROTTA[1,2]
Fabyano Fonseca e SILVA[2]
Marcos Deon Vilela de RESENDE[3]
Moysés NASCIMENTO[4]
Darlene Ana Souza DUARTE[2]
Camila Ferreira AZEVEDO[4]
Rodrigo Reis MOTA[5]

- RESUMO: Em seleção genômica (SG), o grande número de marcadores moleculares utilizados, bem como a demanda computacional dos modelos bayesianos, fundamentados nos algoritmos Monte Carlo Via Cadeias de Markov, faz com que as análises exijam semanas ou até meses de processamento. A computação paralela representa uma solução natural para este problema, visto que esta subdividi um algoritmo em várias tarefas independentes, as quais podem ser processadas em paralelo, reduzindo o tempo de processamento. Objetivou-se comparar a eficiência de processamento do método BayesCπ programado em paralelo com o seu algoritmo sequencial padrão. Duas estratégias de paralelização foram estudadas. A primeira envolveu a análise de múltiplas cadeias MCMC em paralelo, e a segunda referiu-se à paralelização de uma única cadeia MCMC. Utilizou-se a biblioteca MPI e o pacote OpenMPI associado ao compilador gfortran para execução em paralelo desses algoritmos. Foram utilizados dados simulados considerando 10.000 marcadores SNPs e 4.100 indivíduos. O algoritmo sequencial padrão foi processado em 77,29 horas. Ao usar múltiplas cadeias em paralelo o processamento foi 77% mais rápido (17,75hs), enquanto que a estratégia de paralelização de uma única cadeia apresentou um ganho de desempenho de 15% (65,37hs). Conclui-se que a computação paralela é eficiente e pode ser aplicada à SG.

- PALAVRAS-CHAVE: Melhoramento genético; marcadores SNP; regressão Bayesiana.

[1] Universidade Federal dos Vales do Jequitinhonha e Mucuri – UVJM, Departamento de Zootecnia, CEP 39100-000, Diamantina, MG, Brasil (*in memorian*)
[2] Universidade Federal de Viçosa - UFV, Departamento de Zootecnia, CEP 36570-000, Viçosa, MG, Brasil, E-mail: *fabyanofonseca@ufv.br; darlene.duarte@ufv.br*
[3] Embrapa Florestas, Caixa Postal 319, CEP 83411-000, Colombo, PR, Brasil, E-mail: marcos.deon@gmail.com
[4] Universidade Federal de Viçosa, Departamento de Estatística, CEP 36570-000, Viçosa, MG, Brasil, E-mail: *moysesnascim@ufv.br; camila.azevedo@ufv.br*
[5] University of Liège, Gembloux Agro-Bio Tech, TERRA unit, B-5030 Gembloux, Belgium, E-mail: *rodrigo.mota@ufv.br*

1 Introdução

A seleção genômica tem proporcionado expressivos ganhos em programas de melhoramento animal e vegetal. O emprego de ferramentas moleculares avançadas possibilita a varredura uniforme do genoma para milhares de marcadores da classe SNP (*single nucleotide polymorphisms*). Assim, é possível obter mapas genômicos densos e extremamente informativos e, consequentemente, predições de valores genéticos com alta acurácia para características de interesse econômico (MEUWISSEN *et al*., 2001; HABIER *et al*., 2011). Contudo, existem muitos desafios para aplicação da seleção genômica, tanto metodologicamente quanto computacionalmente (WU *et al*., 2011).

O desafio estatístico está em predizer o valor genético genômico (VGG) em situações nas quais o número de indivíduos (genótipos, famílias e clones de várias espécies vegetais) é muito menor do que o número de marcadores. Para resolver este problema, métodos estatísticos sofisticados têm sido propostos (MEUWISSEN *et al*., 2001; de LOS CAMPOS, 2009). Os modelos mais utilizados são baseados na Inferência Bayesiana por meio dos métodos de Monte Carlo Via Cadeias de Markov (MCMC). Tais métodos são iterativos e computacionalmente intensivos. A complexidade desses métodos, associado às informações de milhares de marcadores e indivíduos genotipados, tem resultado em alta demanda computacional, de forma que algumas análises na área de melhoramento animal e vegetal chegam a perdurar por semanas ou meses em computadores de alto desempenho.

A solução para esse problema encontra-se na computação paralela. Para tanto, o algoritmo deve ser dividido em n tarefas independentes que podem ser processadas simultaneamente (em paralelo), contribuindo para redução do tempo de processamento (WILKINSON e ALLEN, 1999; WILKINSON, 2005). A melhora no desempenho computacional obtido com a paralelização do código pode significar mais lucratividade e/ou economia para empresas e instituições de pesquisa, uma vez que fornecem rapidamente os resultados (efeitos de marcadores SNPs) necessários para a utilização da seleção genômica em programas de melhoramento de plantas e animais. Ressalta-se também que, diferente dos supercomputadores, os computadores em paralelo são construídos a partir de componentes baratos (commodities). Assim, os usuários da computação paralela vêm, nos últimos anos, migrando para o uso de um grande número de computadores organizados em estruturas denominadas clusters. Estas estruturas se tornaram muito populares e já estão disponíveis em diversas unidades de pesquisa.

Embora o processamento paralelo possa ser aplicado no contexto dos algoritmos MCMC, as técnicas para paralelizá-los não são triviais por serem processos iterativos, em que a simulação do próximo valor da cadeia não pode começar até que o valor atual tenha sido gerado. Em outras palavras, o estado atual dos parâmetros da cadeia de Markov depende do estado imediatamente anterior (WILKINSON, 2005; REN e ORKOULAS, 2007), o que resulta em um importante problema a ser tratado pela abordagem em paralelo.

Diante do exposto, objetivou-se principalmente demonstrar o quão importante a computação paralela é para a seleção genômica aplicada ao melhoramento. Para isso, comparou-se a eficiência de processamento da metodologia BayesCπ programada em paralelo com a forma sequencial padrão. Duas estratégias de paralelismo foram estudadas

com o intuito de demonstrar qual é a mais recomenda em termos de eficiência computacional.

2 Material e métodos

Foi utilizado um banco de dados simulado disponibilizado no XVI QTLMAS (qtl-mas-2012.kassiopeagroup.com/en/program.php). Uma população base (G0) de 1.020 indivíduos não aparentados (20 parentais 1 e 1000 paramentais 2) foi gerada com um genoma de tamanho 499,75 Mb consistido de 5 cromossomos. Cada cromossomo tinha um tamanho de 99,95 Mb e 2.000 SNPs igualmente distribuídos (1 SNP a cada 0,05 Mb ou cM). Cada uma das quatro gerações seguintes (G1-G4) consistiu de 20 parentais 1 e 1.000 paramentais 2, sendo geradas a partir do acasalamento aleatório de cada parental 1 com 51 parentais 2. O conjunto de dados continha o pedigree de 4.100 indivíduos (apenas os 20 parentais 1 da G0 e demais parentais nas gerações de G1 a G4). Os indivíduos das três primeiras gerações (G1-G3) tinha informações de pedigree e fenótipo, e os 1.020 indivíduos da G4 não tinham informações de fenótipo, mas possuiam informações completas de genótipos para todos os marcadores.

Para análise dos dados utilizou-se o método BayesCπ, proposto por Habier et al. (2011). Esse método tem o objetivo de solucionar alguns problemas dos métodos BayesA e BayesB (MEUWISSEN *et al.*, 2001) no que diz respeito ao impacto dos hiperparâmetros na redução dos efeitos dos SNPs, além de tratar o parâmetro π (proporção de SNPs com efeito nulo) como desconhecido. Este modelo é descrito por:

$$\boldsymbol{y} = \boldsymbol{X}\beta + \sum_{k=1}^{K} \boldsymbol{z_k}\, a_k + \boldsymbol{e},$$

em que: $\boldsymbol{y}$ é um vetor N × 1 de fenótipos da característica; $\boldsymbol{X}$ é uma matriz de incidência dos efeitos fixos em β (média geral); K é o número de SNPs; $\boldsymbol{z_k}$ é um vetor N × 1 de genótipos, 0 (aa), 1(aA) ou 2 (AA) do SNP k; a_k é o efeito genético aditivo do SNP; $\boldsymbol{e}$ é um vetor de efeitos residuais.

Sob a abordagem Bayesiana, a distribuição a *priori* para μ é uma constante, e para a_k depende da variância σ_a^2 e da probabilidade *priori* π de que o SNP *k* tenha efeito zero, isto é:

$$a_k|\pi, \sigma_a^2 = \begin{cases} 0 & \text{com probabilidade } \pi, \\ \sim N(0, \sigma_a^2) & \text{com probabilidade } (1-\pi). \end{cases} \quad (1)$$

O parâmetro π é tratado como desconhecido com distribuição a *priori* uniforme [0,1]. Os efeitos de SNP têm variância comum, σ_a^2. A distribuição a *priori* desta variância é uma qui-quadrada invertida escalada com graus de liberdade $\nu_a < 4$ (para ser pouco informativo) e parâmetro escala S_a^2. O parâmetro S_a^2 foi derivado por Habier et al. (2011) a partir do valor esperado da variável aleatória distribuída de uma qui-quadrada invertida escalada, $E(\sigma_a^2) = \frac{\nu_a S_a^2}{\nu_a - 2} = \tilde{\sigma}_a^2$; portanto:

$$S_a^2 = \frac{\tilde{\sigma}_a^2(\nu_a - 2)}{\nu_a}, \quad (2)$$

em que $\tilde{\sigma}_a^2$ é a variância do efeito aditivo para um locus amostrado aleatoriamente, o qual pode estar relacionado com a variância genética aditiva explicada pelos SNPs, $\tilde{\sigma}_s^2$, como

$$\tilde{\sigma}_a^2 = \frac{\tilde{\sigma}_s^2}{(1-\pi)\sum_{k=1}^{K} 2p_k(1-p_k)}, \quad (3)$$

em que p_k é a frequência alélica do SNP *k* (FERNANDO *et al*., 2008; VAN RADEN, 2009; GIANOLA *et al*., 2009). O componente de variância $\tilde{\sigma}_s^2$ pode ser obtido diretamente das análises tradicionais via matriz de parentesco, uma vez que representa a variância genética aditiva da característica.

A distribuição a *priori* para σ_e^2 segue uma distribuição qui-quadrada invertida escalada com valor arbitrariamente menor que 4 para os graus de liberdade, e parâmetro escala S_e^2. Este parâmetro escalar é produzido pela fórmula $\frac{\tilde{\sigma}^2(4,2-2)}{4,2}$, onde $\tilde{\sigma}^2$ pode ser a variância residual obtida da análise tradicional.

O método BayesCπ necessita apenas do algoritmo Gibbs Sampler, uma vez que as distribuições condicionais completas a posteriori para todos os parâmetros (μ, a_k, σ_e^2, σ_a^2, S_a^2 e π) são conhecidas. A decisão de incluir o SNP *k* no modelo depende da condicional completa a *posteriori* para a variável indicadora δ_k, a qual é introduzida para este propósito. Esta variável indicadora é igual a 1 se SNP *k* é ajustado no modelo e é zero caso contrário. Seguindo o teorema de Bayes, a condicional completa a *posteriori* para $\delta_k = 1$ é dada por:

$$p(\delta_k|y,\theta) = \frac{p(y|\delta_k = 1, \sigma_a^2, \theta)p(\delta_k = 1|\pi)}{p(y|\theta)}.$$

Em todas estas expressões o vetor θ representa todos os outros parâmetros do modelo. As condicionais completas a posteriori para os demais parâmetros do modelo BayesCπ podem ser obtidas por meio das condicionais completas do modelo BayesB detalhadas no trabalho de Meuwissen et al. (2001).

O algoritmo BayesCπ foi escrito em linguagem FORTRAN, sendo a análise da convergência das cadeias MCMC realizadas por meio do pacote BOA (*Bayesian Output Analysis*) (SMITH, 2007) do software R (R CORE TEAM, 2016). Estas análises foram fundamentadas nos diagnósticos de Raftery e Lewis (1992) e Heidelberger e Welch (1983) usando três cadeias MCMC com diferentes valores iniciais e 100.000 iterações, com um *burn-in* de 10.000 iterações (iterações descartadas).

Para paralelização do algoritmo foi utilizada a biblioteca MPI, sendo os códigos compilados pelo gfortran (gcc.gnu.org/wiki/GFortran) associado ao pacote OpenMPI (www.open-mpi.org). Duas estratégias de paralelismo do algoritmo BayesCπ foram estudadas: múltiplas cadeias MCMC em paralelo e paralelização de uma única cadeia MCMC. Todos os códigos, e respectivos bancos de dados fenotípicos e genotípicos encontram-se no seguinte endereço eletrônico: http://www.det.ufv.br/?page_id=1410.

A primeira estratégia é a mais simples, sendo recomendada nas situações em que o *burn-in* é caracterizado por poucas iterações. Nesta estratégia, cada cadeia em cada processador fornece uma diferente sequência de amostras aleatórias, visto que cada cadeia inicia com um valor inicial diferente (WILKINSON, 2005). Todas as cadeias executadas em paralelo possuem o mesmo número de amostras descartadas (*burn-in)* e as iterações restantes de cada cadeia são divididas para os *N* processadores em uso. Para determinar o

quanto o algoritmo com múltiplas cadeias em paralelo foi mais rápido que o algoritmo sequencial, foi utilizado uma medida de aceleração denominada *speedup* (WILKINSON, 2005). A segunda estratégia de paralelização da própria cadeia ocorreu de forma que, em cada iteração, algumas etapas (tarefas) do algoritmo fossem realizadas concomitantemente. Os resultados gerados pelos diferentes processadores foram utilizados pelo processador mestre. Assim, a cada iteração da cadeia a troca de informações entre processadores foi muito intensa.

As análises foram realizadas em um computador constituído por um processador AMD com seis núcleos de processamento de 3,3 GHz e 16 GB de memória RAM (*Random Access Memory*). O tempo médio de processamento para cada estratégia de programação foi calculado e verificou-se se os algoritmos produziram os mesmos resultados de acurácias de predição dos valores genéticos, a qual foi obtida correlacionando o valor genético verdadeiro proveniente da simulação com o valor genético genômico predito.

3 Resultados e discussões

O número de iterações utilizado foi suficiente para assegurar a convergência das cadeias MCMC via critérios de Raftery e Lewis (1992) e Heidelberger e Welch (1983). Na Tabela 1 estão apresentadas as médias e desvios padrão a posteriori para a média geral, π (proporção de SNPs com efeito nulo) e componentes de variância genética e residual obtidos via diferentes formas de paralelização da cadeia MCMC inerente ao método BayesCπ. Observa-se que independentemente do método de paralelização, as estimativas foram muito próximas, indicando que em termos de qualidade de ajuste, a paralelização não influenciou nos resultados da seleção genômica. Nota-se ainda que as medidas de autocorrelação e os erros-padrão de Monte Carlo foram de baixa magnitude, indicando que a convergência foi alcançada pelas cadeias de Gibbs. O fator de dependência foi menor que cinco para todos os parâmetros, o que confirma também que as cadeias convergiram (RAFTERY e LEWIS, 1992).

De acordo com as estimativas dos componentes de variância na Tabela 1, ressalta-se que as variâncias estimadas nos três algoritmos foram similares e próximas do que se esperava com os dados simulados, pois se estimou a herdabilidade em 0.30 para todos os métodos de paralelização, de forma que a verdadeira h^2 fixada na simulação foi igual a 0.28. Utilizando os mesmos 3.000 indivíduos genotipados e fenotipados, referentes às três primeiras gerações (G1-G3), nas populações de treinamento e validação, a acurácia foi de 0,84 para todos os métodos, o que também indica alto poder preditivo do método BayesCπ independentemente do algoritmo de paralelização utilizado.

Em resumo, os resultados da Tabela 1 mostraram que as diferentes formas de paralelização mostraram a mesma eficiência preditiva e precisão na estimação de parâmetros genéticos (componentes de variância e herdabilidades). Porém, ainda é necessário compará-los quanto a eficiência computacional em termos de unidade de tempo demandada para o ajuste do modelo BayesCπ considerando o número de indivíduos genotipados e de marcadores SNPs no presente trabalho.

Tabela 1 - Médias e desvios padrão (DP) *a posteriori* para a média geral (μ), proporção de SNPs com efeitos nulos (π), variância genética aditiva (σ_a^2) e residual (σ_e^2) com respectivas correlações entre amostras sucessivas (r), erro-padrão de Monte Carlo (EpMC) e fator de dependência (FD) calculados a partir das cadeias MCMC geradas por diferentes métodos de paralelização do método de seleção genômica BayesCπ

Método de paralelização	Parâmetros	Média	DP	r	EpMC	FD
Tradicional (sem paralelizar)	μ	19,1147	171,977	0,09	11,618	2,7471
	π	0,9818	0,0045	0,02	0,0001	1,0405
	σ_e^2	21.670,61	680,461	0,03	12,597	1,1046
	σ_a^2	9.392,95	39,939	0,02	0,9515	1,0107
Única cadeia MCMC em paralelo	μ	19,2122	171,663	0,09	11,512	2,3568
	π	0,96985	0,0051	0,03	0,0001	1,4495
	σ_e^2	21.630,55	680,450	0,03	12,120	1,1186
	σ_a^2	9.388,96	39,928	0,03	0,9456	1,0102
Múltiplas cadeias MCMC em paralelo	μ	19,1345	171,902	0,08	11,235	2,7352
	π	0,9769	0,0052	0,01	0,0001	1,0408
	σ_e^2	21.654,71	680,455	0,02	12,253	1,1021
	σ_a^2	9.382,99	39,591	0,01	0,9499	1,0101

Na Figura 1 observa-se a redução do tempo de computação dos algoritmos estudados usando diferentes estratégias de paralelização. No presente trabalho, com 10.000 marcadores e 4.100 indivíduos genotipados, o algoritmo MCMC implementado na forma sequencial demorou 77,29 horas para finalizar. Usando múltiplas cadeias paralelas, observa-se que à medida que o número de processadores aumentava, o tempo de execução diminui consideravelmente até atingir um limite mínimo de 65,37 e 17,75 hs, respectivamente para as estratégias de cadeia única e múltipla.

Em relação ao desempenho referente à paralelização da própria cadeia, usando dois processadores simultaneamente, o processamento foi 19% mais rápido que o algoritmo sequencial. Todavia, a partir de três processadores o desempenho computacional foi comprometido devido a problemas de sincronização das tarefas e da intensa troca de informações entre os processadores a cada iteração do algoritmo. Assim, embora esta estratégia de paralelização seja simples de programar, o desempenho computacional foi relativamente baixo ao comparar com as múltiplas cadeias em paralelo.

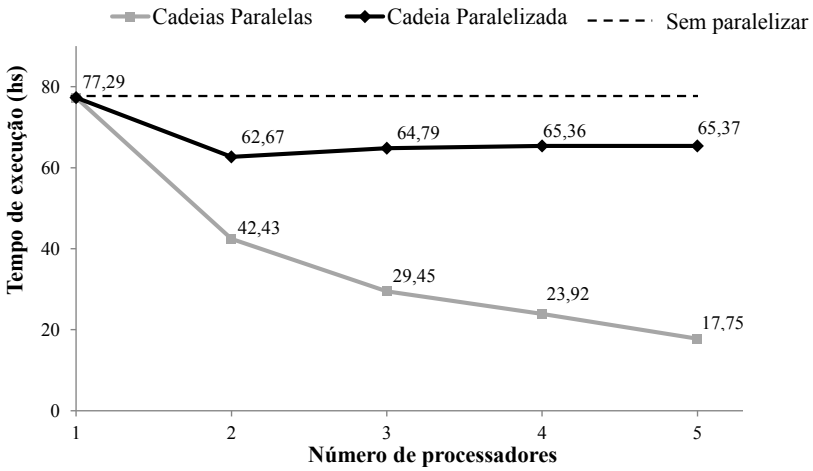

Figura 1 - Gráfico da curva de redução do tempo de computação para as duas estratégias de paralelização (múltiplas cadeias em paralelo e única cadeia paralelizada).

Neste contexto, Wu *et al*. (2011) demonstraram a importância da computação com múltiplas cadeias paralelas em seleção genômica ajustando o modelo Lasso Bayesiano. Na análise, a cadeia principal, com cem mil iterações, foi dividida em 10 menores com o mesmo número de interações. O *burn-in* de cada cadeia foi de mil iterações. Os autores utilizaram 147 indivíduos genotipados para 50.000 SNPs, e concluíram que o processamento via cadeias múltiplas foi 7,7 vezes mais rápido do que o algoritmo baseado em paralelização de cadeia única.

Segundo Kuss e Rasmussen (2005) e Barbu e Zhu (2005) o algoritmo Gibbs sampler geralmente é lento para convergir, principalmente quando se trata de modelos com grande número de parâmetros altamente dependentes, tal como o BayesCπ utilizado no presente trabalho. Tal modelo contempla grande número de parâmetros (efeitos de 10.000 marcadores), e os mesmos são correlacionados devido ao desequilíbrio de ligação que causa naturalmente esta dependência paramétrica. Neste cenário, um método comum para acelerar o Gibbs sampler é introduzir atualizações em cadeias múltiplas (BARBU e ZHU, 2005). Enquanto as atualizações usando cadeias individuais faz com que a cadeia convirja mais lentamente, o Gibbs sampler em cadeias múltiplas acelera a convergência por permitir que parâmetros fortemente acoplados (dependentes), tais como efeitos de marcadores SNP no persente trabalho, atualizem conjuntamente, reduzindo assim o tempo de computação requerido para a convergência.

Conclusões

Diante do objetivo de se comparar a eficiência de processamento em paralelo do método de seleção genômica BayesCπ, conclui-se que as diferentes formas de

paralelização mostraram a mesma eficiência preditiva e precisão na estimação de parâmetros genéticos (componentes de variância e herdabilidades). Porém, em termos de eficiência computacional, o algoritmo sequencial padrão foi processado em 77,29 horas e ao usar múltiplas cadeias em paralelo o processamento foi 77% mais rápido (17,75hs), enquanto que a estratégia de paralelização de uma única cadeia apresentou um ganho de desempenho de 15% (65,37 hs).

Agradecimentos

Aos revisores e editor pelas sugestões.

LAGROTTA, M. L.; SILVA, F. F.; RESENDE, M. D. V.; NASCIMENTO, M.; DUARTE, D. A. S.; AZEVEDO, C. F.; MOTA, R. R. Parallel computation applied to genome selection via Bayesian inference. *Rev. Bras. Biom.,* Lavras, v.35, n.3, p.440-448, 2017.

- *ABSTRACT: In genomic selection (GS), the data analysis using large number of genetic markers based on high computational demand from Bayesian models via Markov Chain Monte Carlo algorithms requires weeks or months to be finished. The parallel computing is a natural solution to this problem, since it splits an algorithm in several independent tasks that are simultaneously (in parallel) processed. It reduces the required computational time when compared with the traditional data processing approach. To demonstrate the importance of parallel computing in GS, its efficiency was compared with the standard sequential algorithm (traditional) by using the BayesCπ method. Two parallelization strategies were studied. The first one involved the analysis of multiple parallel MCMC chains, and the second one referred to the parallelization of the chain itself. The MPI library and OpenMPI package from the gfortran compiler were used for the parallel execution of these algorithms. Simulated data considering 10,000 markers and 4,100 individuals were used. The sequential algorithm was processed at 77.29 hours. The parallel multiple chains were 80% more efficient, while the second parallelization strategy presented an efficiency of 19%. In summary, the parallel computing was efficient and can be applied to GS.*

- *KEYWORDS: Genetic improvement; SNP markers; Bayesian regression.*

Referências

BARBU, A.; ZHU, S. Generalizing swendsen-wang to sampling arbitrary posterior probabilities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v.27, n.8, p.1-35, 2005.

DE LOS CAMPOS G., NAYA, H., GIANOLA, D., CROSSA, J., LEGARRA, A., MANFREDI, E., WEIGEL, K.; COTES, J. M. Predicting quantitative traits with regression models for dense molecular markers and pedigree. *Genetics*, v.182, n.1, p.375-385, 2009.

FERNANDO, R. L., HABIER, D., STRICKER, C., DEKKERS, J. C. M.; TOTIR, L. R. Genomic selection. *Acta Agriculturae Scandinavica, Section A - Animal Science*, v.57, n.4, p.192-195, 2008.

GIANOLA, D., DE LOS CAMPOS, G., HILL, W. G., MANFREDI, E.; FERNANDO, R. Additive Genetic Variability and the Bayesian Alphabet. *Genetics*, v.183, n.1, p.347-363, 2009.

HABIER, D., FERNANDO, R. L., KIZILKAYA, K.; GARRICK, D. J. Extension of the bayesian alphabet for genomic selection. *BMC Bioinformatics*, v.12, n.186, 2011.

HEIDELBERGER, P.; WELCH, P. Simulation run length control in the presence of an initial transient. *Operations Research*, v.31, p.1109-1144, 1983.

KUSS, M.; RASMUSSEN, C. E. Assessing approximate inference for binary gaussian process classification. *Journal of Machine Learning Research*, v.6, p. 1679-1704, 2005.

MEUWISSEN, T. H. E., GODDARD, M. E.; HAYES, B. J. Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, v.157, p.1819-1829, 2001.

RAFTERY, A. E.; LEWIS, S. How many iterations in the Gibbs sampler? In: *Bayesian statistics 4*. Oxford: Oxford University Press, 1992. p.763-773.

REN, R.; ORKOULAS, G. Parallel Markov chain Monte Carlo simulations. *Journal of Chemical Physics*, v.126, p.211102, 2007.

VANRADEN P .M., VAN TASSELL, C. P, WIGGANS, G. R., SONSTEGARD, T. S., SCHNABEL, R. D., TAYLOR, J. F.; SCHENKEL, F. S. Invited review: Reliability of genomic predictions for North American Holstein bulls. *Journal of Dairy Science*, v.92, p.16-24, 2009.

WILKINSON, B.; ALLEN, C. M. *Parallel Programming: Techniques and Applications Using Workstation and Parallel Computers*. Upper Saddle River: Prentice Hall, 1999. 496p.

WILKINSON, D. J. Parallel Bayesian Computation. In Kontoghiorghes, E. J. (Ed.). *Handbook of Parallel Computing and Statistics*. London: Marcel Dekker/CRC Press, p.481-512, 2005.

WU, X. L., BEISSINGER, T. M., BAUCK, S., WOODWARD, B., ROSA, G. J. M., WEIGEL, K. A., GATTI, N. L., GIANOLA, D. A primer on high-throughput computing for genomic selection. *Frontiers in Genetics*, v.2, p.1-10, 2011.

Recebido em 01.02.2016

Aprovado após revisão em 26.04.2017