

UNIVERZA V LJUBLJANI
BIOTEHNIŠKA FAKULTETA

Jana Obšteter

**SELEKCIJA Z OPTIMALNIMI PRISPEVKI V SLOVENSKIH REJSKIH
PROGRAMIH ZA GOVEDO**

Seminarska naloga pri predmetu MATEMATIČNI MODELI IN OPERACIJSKE RAZISKAVE V
BIOZNANOSTIH
Doktorski študij Bioznanosti

Ljubljana, 2018

Kazalo vsebine

1 PREDSTAVITEV RAZISKOVALNEGA PROBLEMA IN CILJEV	1
1.1 SELEKCIJA PRI GOVEDU	1
1.1.1 <i>Napoved plemenskih vrednosti</i>	1
1.1.2 <i>Cilji selekcije</i>	3
1.2 RAZVOJ REŠITVE ZA SELEKCIJO Z OPTIMALNIMI PRISPEVKI.....	5
1.2.1 <i>Deterministični algoritmi</i>	6
1.2.1.1 Razvoj rešitve za OCS s semidefiniranim programiranjem.....	7
1.2.1.2 Razvoj rešitve za OCS s stožčastim programiranjem drugega reda	8
1.2.2 <i>Hevristični algoritmi</i>	9
1.2.2.1 Razvoj rešitve za OCS z evolucijskim algoritmom	10
1.4 RAZISKOVALNI CILJI	12
2 METODE IN MATERIALI DELA	13
2.1 PRIPRAVA ORODJA IN PARAMETROV ZA SIMULACIJO	13
2.2 TESTIRANI SCENARIJI	13
2.3 OPTIMIZACIJA PRISPEVKOV	14
2.3.1 DETERMINISTIČNA OPTIMIZACIJA S STOŽČASTIM PROGRAMIRANJEM DRUGEGA REDA.....	14
2.4 PRIMERJALNI PARAMETRI.....	16
3 REZULTATI IN RAZPRAVA	16
3.1 GENETSKI NAPREDEK	17
3.2 GENETSKA VARIABILNOST	18
3.3 UČINKOVITOST SELEKCIJE	19
4 ZAKLJUČKI	21
5 VIRI IN LITERATURA	23

1 PREDSTAVITEV RAZISKOVALNEGA PROBLEMA IN CILJEV

1.1 SELEKCIJA PRI GOVEDU

Selekcija stremi h genetskemu izboljšanju populacije. V živinoreji izvajamo umetno selekcijo na gospodarsko pomembne lastnosti. To so predvsem proizvodne lastnosti, npr. prireja mleka ali prirast mesa, lastnosti zunanosti živali in lastnosti zdravja. Ključen korak selekcije je odbira najboljših živali za starše naslednje generacije. Zaradi dolgih generacijskih intervalov, velikega vpliva nekaj izbranih moških živalih ter visokih stroškov testiranja in vzreje živali je odbira genetsko najboljših živali izrednega pomena. Hkrati pa je identifikacija teh živali težavna zaradi več virov variabilnosti. Med temi so genetska variabilnost za lastnost, velik okoljski vpliv, segregacija in rekombinacija pri potomcih že identificiranih elitnih živali ter vključenost različnih organizacij v selekcijsko delo. Živali so rangirane in odbrane na podlagi plemenskih vrednosti (PV), ki so dvakratno odstopanje potomcev od povprečja populacije (Falconer in MacKay, 1996).

Uspešnost selekcije merimo z doseženim genetskim napredkom (ΔG), za izračun katerega velja:

$$\Delta G = \frac{i \times r \times \sigma_A}{L} \quad (1)$$

; kjer je ΔG = letni genetski napredek, merjen v enotah standardnega odklona na leto, i = intenzivnost selekcije, r = točnost selekcije, σ = genetski standardni odklon, L = generacijski interval (Lush, 1945). Le ta je odvisen od točnosti selekcije (r), intenzivnosti odbire (i), genetske variabilnosti za lastnost (σ_A) in generacijskega intervala (L). Ti parametri se bistveno razlikujejo med moško in žensko potjo selekcije pri govedu, zato sta Rendel in Robertson, 1950, definirala štiri poti selekcije pri govedu: matere mater, matere očetov, očetje mater in očetje očetov. Najstrožja selekcija se vrši pri odbiri očetov, saj je vpliv moških živali zaradi velikega števila potomcev, ki jih dosežemo preko umetnega osemenjevanja, znatno večji od vpliva ženskih živali, ki imajo v povprečju le enega potomca letno.

1.1.1 Napoved plemenskih vrednosti

V selekciji živali lahko PV napovemo na dva načina: klasičnega ali genomskega. Oboje PV napovemo z metodo mešanega modela, pri čemer je vpliv živali naključen vpliv (enačba 2). Za napoved potrebujemo vektor fenotipskih vrednosti (rešitev), matriko sistematskih vplivov in matriko sorodstva. Model za napoved je sledeč:

$$y = Xb + Za + e \quad (2)$$

; kjer je X matrika dogodkov, b je vektor ocen za sistematske vplive, Z je matrika sorodstva, zgrajena na podlagi rodovniških podatkov, a je vektor iskanih plemenskih vrednosti, e pa je vektor ostankov.

V **klasični** napovedi je matrika sorodstva zgrajena na podlagi rodovniških podatkov, ki jo imenujemo tudi matrika sorodstva C (angl. *coancestry matrix*). V napovedi uporabimo $\frac{1}{2} C$, ki jo

imenujemo matrika **A**. Slednja odraža pričakovano sorodstvo oz. kovarianco med živalmi (slika 1). Matriki **C** in **A** sta pozitivni definitni matriki. Elementi matrike **C** so koeficienti sorodstva c (angl. *coancestry coefficient*), matrike **A** pa koeficienti a , ki so $\frac{1}{2}$ koeficientov sorodstva c . Diagonalni elementi matrike **A** izražajo sorodnost posameznika samega s sabo, in jih zapišemo kot $a_{ii} = (1 + F_i)$, kjer je F_i koeficient inbridinga za žival i . Nediagonalni elementi pa predstavljajo sorodstvo med dvema različnima posameznikoma in so oblike $a_{ij} = \frac{1}{2} (a_{ij'} + a_{ij'')}$, kjer sta j' in j'' oče in mati živali j . Pri tem je $F_i = \frac{1}{2} (a_{sd})$, kjer je s oče, d pa mati živali i .

	#1	#2	#3	#4	#5	#6	#7	#8
#1	1.00							
#2		1.00						
#3			1.00					
#4				1.00				
#5	0.50	0.50			1.00			
#6	0.50	0.50			0.50	1.00		
#7	0.50	0.50			0.50	0.50	1.00	
#8	0.50	0.50			0.50	0.50	0.50	1.00

Slika 1: Primer matrike **A**, ki je $\frac{1}{2}$ matrika sorodstva **C**, za posameznike 1 - 8.

Točnost klasičnih PV je tako močno odvisna od lastnih izmerjenih vrednosti oz. izmerjenih vrednosti potomcev in posledično zaznamovana z dolgimi generacijskimi intervali (povprečna starost starša ob rojstvu izbranih potomcev). Ker mlade živali še nimajo lastnih fenotipskih vrednosti, je njihova PV preprosto povprečje PV njihovih staršev (angl. *parent average*, PA), točnost katere je zaradi rekombinacije in segregacije majhna.

V zadnjih letih se je povečala dostopnost genomske informacije. Znižanje cene genotipizacije (postopek pridobivanja genomske informacije za posameznika) je vodilo do razvoja t.i. genomske selekcije (Meuwissen in sod., 2001). Genomska selekcija omogoča napoved PV, imenovanih genomske plemenske vrednosti (gPV), za vse genotipizirane živali, tudi mlade brez lastnih fenotipskih vrednosti. Točnost gPV je večja od točnosti PA, kar omogoča zgodnejšo odbiro staršev (krajši generacijski interval) in bolj točno odbiro mladih živali (Pryce in Daetwyler, 2012).

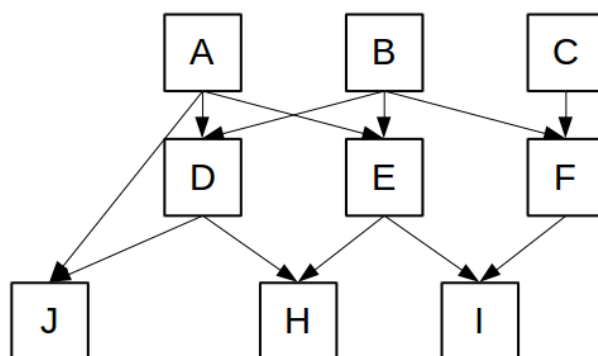
V **genomski** napovedi je matrika sorodstva zgrajena na podlagi genomske informacije. To zahteva genotipizacijo živali na izbranih številnih mestih (označevalci). Pri napovedi genomske PV matrike sorodstva namesto na podlagi rodovniških podatkov zgradimo na podlagi genotipov živali na pregledanih označevalcih. Takšno matriko imenujemo genomska matrika sorodstva ali matrika **G**, ki je pozitivna semidefinitna matrika (slika 2). Genomsko sorodstvo imenujemo tudi realizirano sorodstvo. V genomski napovedi ocenjujemo vplive posameznih označevalcev, na podlagi katerih napovedmo gPV. Za napoved gPV zaradi večjega števila spremenljivk kot podatkov uporabimo metodo L2 regularizacije (Ridge regresija) ali pa uporabimo Bayesovski pristop, kjer za vplive posameznih označevalcev uporabimo apriorij distribucijo (Meuwissen in sod., 2001).

	#1	#2	#3	#4	#5	#6	#7	#8
#1	0.31							
#2	-0.47	0.83						
#3	-0.47	0.83	0.83					
#4	0.27	-0.51	-0.51	0.74				
#5	-0.11	0.42	0.42	-0.66	1.29			
#6	-0.11	0.42	0.42	-0.66	1.29	1.29		
#7	-0.01	-0.03	-0.03	-0.06	0.34	0.34	0.43	
#8	-0.06	-0.07	-0.07	0.41	-0.98	-0.98	-0.38	0.86

Slika 2: Primer genomske matrike sorodstva za posameznike 1 - 8.

1.1.2 Cilji selekcije

Čeprav si selekcija prizadeva za genetski napredek populacije, je treba ob tem paziti tudi na ohranjanje genetske variabilnosti. Iz enačbe (1) je razvidno, da manjšanje genetske variabilnosti manjša tudi genetski napredek. Povečanje intenzivnosti selekcije oz. uporaba majhnega števila elitnih posameznikov kratkoročno vodi do večjega genetskega napredka, vendar pa ne omogoča dolgoročnega napredka zaradi zmanjšanja genetske variabilnosti v populaciji. Slednje je posledica večjega parjenja v sorodstvu – inbridinga – preko večje možnosti za odbiro sorodnikov (Woolliams in sod., 2015). Koeficient inbridinga F je lastnost diploidnih organizmov – ki imajo dva seta kromosomov – in je definiran kot verjetnost, da sta v posamezniku oba alela na nevtralnem mestu na genomu (brez pritiska selekcije) identična po izvoru (Wright, 1921). Večji inbriding poleg onemogočanja dolgoročnega genetskega napredka vodi tudi do depresije zaradi inbridinga, ki pomeni zmanjšano preživetje in plodnost potomcev sorodnih posameznikov. Čeprav je inbriding neizbežen v končno velikih populacijah, je pomembno, da uravnavamo stopnjo, s katero koeficient inbridinga narašča (ΔF). Samo spremljanje koeficienta inbridinga F v starših in njihovih potomcih ni zadosten kriterij za trajnostno upravljanje s populacijo, saj se vpliv paritvenih odločitev na koeficient inbridinga pozna z zamikom dveh generacij, ko se ustvarijo zanke v rodovniku (slika 3) (Woolliams in sod., 2015). Poleg tega je F v populaciji relativen na F v prvi generaciji rodovnika, za katero predpostavimo, da je neinbridirana in nesorodna. Posledično sama globina rodovnika vpliva na ocenjen F v populaciji (Woolliams in sod., 1998).



Slika 3: Primer rodovnika z zankami, ki ustvarjajo inbriding.

Selekcija z optimalnimi prispevki

Upravljanje ΔF je pomembno v dveh situacijah: prva je ohranjanje populacij, kjer je cilj minimiziranje ΔF , druga pa selekcija, kjer je cilj čimbolj učinkovito uporabiti razpoložljivo genetsko variabilnost za ustvarjanje genetskega napredka oz. maksimiranje genetskega napredka ob omejitvi ΔF (Wooliams in sod., 2015). Za trajnostno upravljanje s populacijo v selekciji in za dolgotrajni uspeh selekcije moramo tako stremeti k povečanju genetskega napredka in hkrati nadzirati ΔF v populaciji. James in McBride sta leta 1958 prvič postavila koncept dolgoročnih genetskih prispevkov posameznikov, r_i , ki so definirani kot natančen prispevek genov, ki jih posameznik prispeva v bodoči genetski »bazen«. Z drugimi besedami – r za posameznika i je delež genov v populaciji veliko generacij kasneje, ki izhajajo iz posameznika i . Dolgoročni genetski prispevki so preko preprostih razmerij povezani s ključnimi parametri populacijske dinamike:

- z genetskim napredkom preko

$$\Delta G = \sum r_i x a_i, \quad (3)$$

kjer je r dolgoročen prispevek posameznika in a prispevek Mendelskega vzorčenja (angl. *Mendelian sampling term*), ki je odstopanje potomca od povprečja staršev (Wooliams in sod., 1999);

- s stopnjo inbridinga, kjer je

$$\Delta F = 1/4 \sum r_i^2 \quad (4)$$

(Wray in Thompson, 1990).

Rejske organizacije imajo cilj maksimirati ΔG ob vzdrževanju predhodno določene vrednosti za stopnjo inbridinga ΔF (Gorjanc in Hickey, 2018). Ta ideja je implementirana v selekciji z optimalnimi prispevki (angl. *optimal contribution selection*, OCS) (Meuwissen 1997; Grundy et al. 1998), ki teži k uravnoteženju dveh kontradiktornih ciljev - dosegi genetskega napredka in ohranjanju genetske variabilnosti. Nasprotje temu imenujemo selekcija s selektivno odbiro (angl. *truncation selection*), pri čemer je cilj samo maksimiranje genetskega napredka brez obzira na ohranjanje genetske variabilnosti. Genetska varianca skozi generacije inbridinga pada in po t generacijah znaša

$$\sigma_{A,t}^2 = (1 - F_t) x \sigma_{A,0}^2 \quad (5)$$

; kjer je $\sigma_{A,t}^2$ genetska varianca v generaciji t in $\sigma_{A,0}^2$ genetska varianca v generaciji 0 (Falconer in Mackay, 1996). Vidimo, da je padec genetske variance po generacijah sorazmeren z naraščanjem inbridinga. Posledično moramo za upravljanje z genetsko varianco nadzorovati ΔF , saj je F posameznika enak koeficientu sorodstva njegovih staršev (Wooliams in sod., 2015). Povprečni inbriding v generaciji t je tako enak povprečnemu sorodstvu izbranih očetov in mater v generaciji $t - 1$. Sorodstvo med dvema posameznikoma i in j je enako:

$$C_{ij} = 1/2(C_{oče(i), oče(j)} + C_{oče(i), mati(j)} + C_{mati(i), oče(j)} + C_{mati(i), mati(j)}) \quad (6)$$

Ker je ΔF povezan z r^2 prednikov (enačba (4)), je minimiziranje ΔF ekvivalentno minimiziranju $\sum r_i^2$. OCS tako nadzoruje povečanje inbridinga preko nadzora povečanja **skupnega sorodstva** izbranih staršev (Wooliams in sod., 2015). Skupno sorodstvo staršev je definirano kot povprečno sorodstvo te skupine uravnoteženo glede na posameznikove bodoče prispevke k naslednji generaciji. Je tudi verjetnost, da sta dva naključno izbrana alela iz genskega »bazena« populacije

identična po izvoru (Woolliams, 2015). Stopnjo inbridinga ΔF lahko tako nadzorujemo preko skupnega sorodstva izbranih staršev. Da bi se izognili ustvarjanju ozkih grl, mora biti ΔF konstanten, F pa mora slediti:

$$F_t = 1 - (\delta f)^{t-1} \quad (7)$$

; kjer je δf tarčen ΔF . Ekvivalentno mora skupno sorodstvo isti poti slediti eno generacijo prej:

$$C_t = 1 - (\delta f)^t. \quad (8)$$

V OCS se določi omejitev za skupno sorodstvo, ki je:

$$C_t = 1 - (1 - C_o)(1 - \delta f)^t \quad (9)$$

; če se optimizacija začne v generaciji ena ali:

$$C_t = C_{t-1} + \delta f(1 - C_{t-1}) \quad (10)$$

; če je sorodstvo v prvi generaciji (C_o) povišano (Woolliams in sod., 2015). Ker nastavimo fiksno pot za omejitev C_t , bo vsako odstopanje od optimalnih prispevkov v generaciji t , ki bi povzročilo preveliko ali premajhno sorodstvo populacije v generaciji $t+1$, v OCS avtomatsko popravljeno, saj C_{t-1} še naprej sledi isti poti. Posledično bo omejitev v generaciji $t+1$ bolj stroga ali bolj sproščena glede na dejansko izmerjeno sorodstvo populacije (v primerjavi z načrtovanim).

Selekcija z optimalnimi prispevki skuša maksimirati genetski napredek v naslednji generaciji (G_{t+1}) in hkrati omejiti skupno sorodstvo na C_{t+1} . Pri tem so potrebne dodatne omejitve, ki izhajajo iz praktičnega vidika parjenja: zagotoviti moramo, da so prispevki vseh posameznikov ne-negativni, da je vsota moških kot tudi ženskih posameznikov $\frac{1}{2}$ ter da je vsota prispevkov vseh posameznikov enaka 1 (Woolliams in sod., 2015).

Prispevke lahko optimiziramo na dva načina. Prvi način vključuje optimizacijo prispevkov z namenom: i) maksimiranja genetskega napredka z omejitvijo stopnje skupnega sorodstva med izbranimi posamezniki; ali ii) samo minimiziranja skupnega sorodstva. Tej optimizaciji prispevkov lahko sledi še ena optimizacija, in sicer za dodelitev partnerjev za parjenje za minimiziranje inbridinga posameznih parjenj, kar prepreči preveliko depresijo zaradi inbridinga v potomcih (Gorjanc in Hickey, 2018). Obe optimizaciji lahko rešimo deterministično z Lagrangeovimi multiplikatorji (Meuwissen, 1997), linearnim programiranjem (Toro in Perez-Enciso, 1990), konveksnim programiranjem (Pong-Wong in Woolliams, 2007; Yamashita in sod., 2015) ali hevrističnimi metodami optimizacije (Gorjanc in Hickey, 2018). Drugi način vključuje vzajemno optimizacijo prispevkov in določitve partnerjev za parjenje preko optimizacije načrta parjenja (Kinghorn in Shepherd, 1999). Ta skupna optimizacija ni rešljiva z determinističnimi metodami, pač pa mora biti rešena s stohastičnimi ali hevrističnimi metodami.

1.2 RAZVOJ REŠITVE ZA SELEKCIJO Z OPTIMALNIMI PRISPEVKI

Glavni cilj selekcije z optimalnimi prispevki je maksimirati genetski napredek, vendar ob tem ohraniti genetsko variabilnosti. Algoritmi za optimizacijo problema so se razvijali od preprostih linearnih modelov, preko linearnega celoštevilskega programiranja, linearnega programiranja s kvadratno omejitvijo, ki je bila sprva rešena z Lagrangeovimi multiplikatorji, kasneje pa s

konveksnim programiranjem, in sicer semidefinitnim programiranjem in stožčastem programiranjem drugega reda, za rešitev problema pa so raziskovalci implementirali tudi hevristične algoritme. Algoritmi se med seboj razlikujejo v optimizacijskem postopku, številu ciljev za optimizacijo, potrebnem računalniškem času, spominu ... Podrobneje bom razdelala stožčasto programiranje in hevrističen evolucijski algoritem, ki sta implementirana tudi v prosto dostopnih programih za optimizacijo prispevkov v selekciji.

1.2. 1 Deterministični algoritmi

Lindgren je prvi predlagal linearno razporeditev (angl. *linear deployment*), v kateri so prispevki kandidatov sorazmerni njihovim PV-jem (Lindgren in sod., 1989). Ta pristop zagotavlja, da imajo posamezniki z višjimi PV-ji večje prispevke, ne nadzira pa sorodstva med posamezniki. Metoda posledično zagotovi uspeh selekcije le v primeru, da so rodovniki neodvisni in posamezniki med seboj nesorodni, saj pristop ne omeji prispevkov posameznikov (Yamashita in sod., 2015). Napredek se tako skozi generacije občutno zmanjša zaradi depresije zaradi inbridinga in povečanja sorodnosti populacije.

Toro in Pérez-Enciso, 1990, sta uporabilo celoštevilsko linearno programiranje za optimizacijo odbire staršev in partnerjev za parjenje. Zelena rešitev je v tem primeru matrika $\mathbf{X} = [x_{ij}]$, kjer je x_{ij} binarna spremenljivka (0 ali 1) in predstavlja, ali sta oče i in mati j izbrana za parjenje. Matrika je izbrana z namenom maksimiranja genetskega napredka. Meuwissen je prvi uvedel **kvadratno omejitev** za nadzor skupnega sorodstva (Meuwissen, 1997). Optimizacijski problem za OCS tako predstavlja naslednji set enačb (Pong-Wong in Woolliams., 2007):

$$\text{min:} \quad -c^T \mathbf{g} \quad (11a)$$

$$\text{omejitve:} \quad c^T \mathbf{A}c/2 \leq \theta \text{ oz. } c^T \mathbf{C}c \leq \theta \quad (11b)$$

$$c^T \mathbf{s} = 0.5 \quad (11c)$$

$$c^T \mathbf{d} = 0.5 \quad (11d)$$

$$c \geq 0 \quad (11e)$$

$$c \leq m \quad (11f)$$

Ciljna funkcija je minimizacijska funkcija, ki minimizira negativni genetski napredek, kar je ekvivalentno maksimiranju genetskega napredka (11a). S c označujemo prispevke posameznikov, \mathbf{g} pa označuje vektor napovedanih PV. Z omejitvijo za skupno sorodstvo v enačbi (11b) zagotovimo, da skupno sorodstvo v naslednji generaciji ne preseže prednastavljene meje θ . Omejitev lahko izrazimo z matriko sorodstva \mathbf{C} ali polovico le-te, ki je matrika \mathbf{A} . Z omejitvami v (11c) in (11d) zagotovimo, da je prispevek vsakega izmed spolov $\frac{1}{2}$ ter da je vsota prispevkov 1. Omejitev v (11e) zagotavlja, da so prispevki nenegativni – prispevek 0 pomeni, da posameznik ni bil izbran kot starš naslednje generacije. Z omejitvijo v (11f) nastavimo zgornjo mejo za prispevek posameznika in s tem preprečimo preveliko uporabo ene živali.

Meuwissen je problem rešil z metodo **Lagrangeovih multiplikatorjev** (Meuwissen, 1997). Čeprav se je ta metoda uporabljala v selekciji, pa ne zagotavlja vedno optimalne rešitve, saj se tekom

iteracij določene vrednosti za prispevke fiksirajo na zgornjo ali spodnjo dovoljeno vrednost (Pong-Wong in Woolliams, 2007). Pong-Wong in Woolliams, 2007, sta nato problem prevedla na problem **semidefinitnega programiranja**, Yamashita in sod., 2015, pa so problem rešili s **stožčastim programiranjem drugega reda**.

Tako semidefinitno kot tudi stožčasto programiranje drugega reda sta podrazreda stožčastega programiranja. Problemi stožčaste optimizacije pa so eni izmed najenostavnejših primerov konveksne optimizacije s posplošenimi neenakostmi. Konveksni optimizacijski problemi minimizirajo konveksno funkcijo preko konveksne množice. Temeljna lastnost konveksne optimizacije je dejstvo, da je vsak lokalni optimum tudi globalni optimum. Stožčasto programiranje pa optimizira linearno funkcijo (12a) z linearno omejitvijo (12c), čemu je dodana ena ali več stožčastih omejitev (omejitev neenakosti v (12b)). (Boyd, 2004).

$$\min: c^T x \quad (12a)$$

$$\text{omejitve: } Fx + g \preceq_K 0 \quad (12b)$$

$$Ax = b. \quad (12c)$$

1.2.1.1 Razvoj rešitve za OCS s semidefinitnim programiranjem

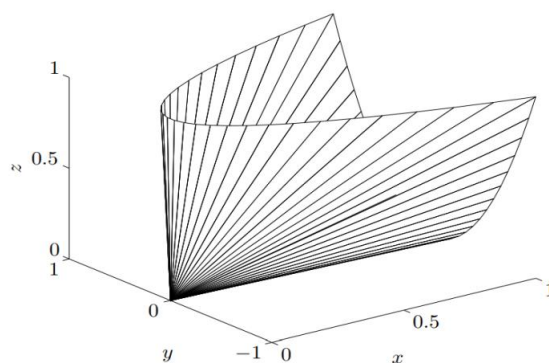
Semidefinitno programiranje (SDP) spada pod stožčasto programiranje, ki minimizira linearno ciljno funkcijo (13a) z omejitvijo linearne neenakosti (13b). Neenakost je linearna matrična neenakost (angl. *linear matrix inequality*, LMI), ki si lahko predstavljamo tudi kot presek stožca semidefinitne pozitivne matrike (slika 4) z afinim prostorom, kar imenujemo spektrahedron (Ramana, 1995). Tako je SDP optimiziranje linearne funkcije preko spektrahedrona. Standardna oblika SDP je tako:

$$\min: c^T x \quad (13a)$$

$$\text{omejitve: } x_1 F_1 + \dots + x_n F_n + G \preceq 0 \quad \text{oz. } F(x) \geq 0 \quad (13b)$$

$$Ax = b, \quad (13c)$$

; kjer so $G, F_1, \dots, F_n \in \mathbf{S}^k$ in $A \in \mathbf{R}^{p \times n}$. Če so matrike diagonalne matrike, je problem zreduciran na linearen program (Boyd, 2004).



Slika 4: Omejitev s pozitivnim semidefinitnim stožcem v S^2 (vir: Boyd, 2004).

Probleme semidefinitnega programiranja lahko učinkovitost rešimo z npr. metodami notranje točke. Slednja išče optimalno rešitev v strogo možnem prostoru parametrov. Glavni korak za

izvedbo semidefinitnega programiranja je tako dokaz, da je problem konveksen in prevod problema na standardno obliko (Boyd, 2004).

Pong-Wong in Woolliams, 2007, sta izkoristila dejstvo, da je matrika sorodstva \mathbf{C} - in posledično tudi matrika \mathbf{A} - vedno pozitivno definitna, zaradi česar sta lahko omejitve za skupno sorodstvo prevedla na pozitiven semidefinitni pogoj s simetrično matriko. Pri semidefinitnem programiranju moramo kvadratne omejitve moramo najprej prevesti na standardno formo oz. raven linearne matrične neenakosti. Pong-Wong in Woolliams sta za pretvorbo kvadratne omejitve uporabila Schurov komplement (14a), vse omejitve pa zapisala v obliki diagonalne bločne matrike, kjer je vsaka izmed omejitev zajeta v diagonalnih blokih (14b). SDP najde optimalno rešitev - v nasprotju z Lagrangeovimi multiplikatorji - vendar pa zahteva veliko računalniškega časa. Zato so Yamashita in sod., 2015, predlagali problem prevesti na stožčastega problem programiranja drugega reda.

$$c^T \mathbf{A} c / 2 \leq \theta \quad \rightarrow \quad \begin{bmatrix} \mathbf{A}^{-1} & \mathbf{c} \\ \mathbf{c}^T & 2\theta \end{bmatrix} \geq \mathbf{0} \quad (14a)$$

$$\mathbf{Y} = \begin{bmatrix} \begin{bmatrix} \mathbf{A}^{-1} & \mathbf{c} \\ \mathbf{c}^T & 2\theta \end{bmatrix} & & & & & \\ & \begin{bmatrix} \mathbf{c}^T \mathbf{s} - 0.5 \end{bmatrix} & & & & \\ & & \begin{bmatrix} -\mathbf{c}^T \mathbf{s} + 0.5 \end{bmatrix} & & & \\ & & & \begin{bmatrix} \mathbf{c}^T \mathbf{d} - 0.5 \end{bmatrix} & & \\ & & & & \begin{bmatrix} -\mathbf{c}^T \mathbf{d} + 0.5 \end{bmatrix} & \\ & & & & & \text{diag}(\mathbf{c}) \\ & & & & & & \text{diag}(\mathbf{m} - \mathbf{c}) \end{bmatrix} \geq \mathbf{0} \quad (14b)$$

1.2.1.2 Razvoj rešitve za OCS s stožčastim programiranjem drugega reda

Če je omejitev stožčastega programiranja v obliki Lorentzovega stožca (tudi stožec drugega reda ali »kornet«) (slika 5), potem govorimo o stožčastem programiranju drugega reda. Stožčasto programiranje drugega reda tako optimizira linearno funkcijo (15a) preko preseka afinnega prostora (15c) in stožca drugega reda (15b).

$$\min: c^T x \quad (15a)$$

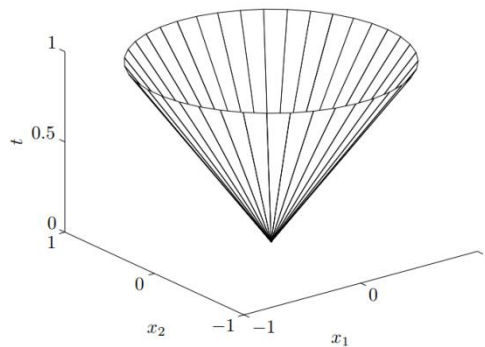
$$\text{omejitve: } -(A_i x + b_i, c_i^T x + d_i) \preceq_{K_i} 0, \quad i = 1, \dots, m \quad (15b)$$

$$F x = g, \quad (15c)$$

; kjer je $\mathbf{x} \in \mathbf{R}^n$ spremenljivka, ki jo optimiziramo, $A_i \in \mathbf{R}^{n_i \times n}$ in $F \in \mathbf{R}^{p \times n}$,

$$K_i = \{(y, t) \in \mathbf{R}^{n_i+1} \mid \|y\|_2 \leq t\}, \quad (15d)$$

pa je stožec drugega reda v \mathbf{R}^{n_i+1} (Boyd, 2004). Linearne programe, konveksne kvadratne programe in konveksne kvadratne programe s kvadratnimi omejitvami lahko vse formuliramo kot probleme stožčastega programiranja drugega reda.



Slika 5: Omejitev s stožcem drugega reda v R^3 (vir: Boyd, 2004).

Yamashita in sod., 2015, so izkoristili dejstvo, da je matrika A pozitivna definitna, kar omogoča, da kvadratno omejitev za skupno sorodstvo izrazimo tudi kot omejitev stožca drugega reda. Problem v enačbah (11) so tako prevedli na standardno formo stožčastega programiranja. Najprej nam pozitivna definitnost omogoči Cholesky faktorizacijo matrike A . S tem pridobimo zgornjo trikotno matriko, $A = U^T U$, omejitev za skupno sorodstvo iz (11b) pa prevedemo na:

$$c^T A c / 2 \leq \theta \leftrightarrow c^T U^T U c \leq 2\theta \leftrightarrow \|Uc\| \leq \sqrt{2\theta} \leftrightarrow \begin{pmatrix} \sqrt{2\theta} \\ Uc \end{pmatrix} \in \mathcal{K}^{1+m} \quad (16)$$

1.2.2 Hevristični algoritmi

V nasprotju z natančnimi metodami, ki vedno najdejo optimalno rešitev, hevristični algoritmi ne morejo garantirati optimalne rešitve, vendar pa bodo na podlagi izkušenj ali presoje zelo verjetno podali dobro rešitev. Hevristične algoritme uporabimo, ko želimo zreducirati iskanje pri reševanju problema ali ko želimo znotraj omejenega računalniškega časa pridobiti sprejemljivo rešitev. Pogosto jih imenujemo tudi pravilo palca, izobraženo ugibanj, intuitivna presoja ali preprosto zdrava pamet (Pearl, 1984). Hevristika je popularna tudi v operacijskih raziskavah, saj: a) velikokrat je malo verjetno, da bomo našli optimalno rešitev zaradi računalniške zahtevnosti problemov; b) če so problemi slabo definirani ali pa podatki netočni, najverjetneje najdena »optimalna« rešitev ne bo optimalna za prave podatke - v takšnih primerih je bolje pridobiti robustno rešitev, ki je blizu optimalni; c) kadar uporabnik potrebuje več različnih rešitev za končno odločitev, še posebno v situacijah, kjer mora biti več kriterijev uravnoteženih s človeškim faktorjem (Aickelin in Clark, 2011).

Uporaba hevristike je uporabna v naslednjih primerih (Zanakis in Evans, 1981):

- netočni in pomanjkljivi podatki – ocena parametrov modela na podlagi le-teh lahko vsebuje večjo napako kot suboptimalna rešitev dobre hevristike;
- uporabljamo poenostavljen model – le-ta je že tako ali tako netočna predstavitev problema, zaradi česar je »prava« rešitev zgolj akademska;
- zanesljiva natančna (angl. *exact*) metoda ni na voljo;
- naturna metoda je na voljo, vendar je računalniško neprivlačna – ali ima velike zahteve za računalniški čas ali pa spomin;

- e) za izboljšanje izvedbe optimizacije, npr. za pridobitev začetne rešitve ali kot vodilo za zmanjšanje števila možnih rešitev;
- f) ponavljajoča se pogosta potreba po reševanju istega problema, pri čemer pomaga hitrost heuristike;
- g) heuristična rešitev je morda dovolj dobra – če je boljša od trenutne;
- h) heuristika je enostavna, zato ji uporabnik razume in ji zaupa;
- i) kot naprava za učenje – pridobivanje vpogleda v kompleksne probleme;
- j) ostale omejitve sredstev – čas in denar, znanje ...

Glavna lastnosti heurističnih optimizacijskih algoritmov je, da začnejo s poljubno rešitvijo, potem pa preko iteracij po nekem pravilu proizvajajo nove rešitve, ki jih ovrednotijo in sčasoma podajo najboljšo rešitev, ki so jo našli tekom iskanja. Iterativni proces je ponavadi ustavljen, ko: i) dano število iteracij ne uspemo izboljšati rešitve (algoritem konvergira); ii) ko je najdena rešitev dovolj dobra; iii) ko proces doseže dovoljen računalniški čas; ali iv) ko nek interni parameter konča izvajanje procesa (Maringer, 2005).

Skupina heurističnih algoritmov je raznolika v tipu uporabljenih metod. Prav tako obstajajo različne razvrstitve heurističnih algoritmov, ki delijo algoritme glede na različne parametre. Ena izmed bolj uporabnih je delitev glede na število sočasno uporabljenih rešitev. Tako jih delimo na trajektorne metode, ki uporabljajo eno rešitev, in populacijske rešitve, ki delujejo nad populacijo rešitev. Za trajektorne metode je značilno, da pri iskanju opišejo tirnico v iskalnem prostoru, iskalni proces populacijskih algoritmov pa opisuje evolucijo množice točk v iskalnem prostoru. Ker bomo v nalogi uporabili algoritem, ki spada med populacijske metode, se bomo podrobneje osredotočili na to skupino. Med najbolj raziskanimi populacijskimi algoritmi sta optimizacija s kolonijami mravelj (angl. *Ant-Colony Optimisation*) in evolucijsko računanje (angl. *Evolutionary computation*). Oba posnemata fenomena iz narave: medtem ko prvi posnema orientacijo mravelj na podlagi feromonov, drugi posnema proces naravne selekcije (Korošec, 2004).

1.2.2.1 Razvoj rešitve za OCS z evolucijskim algoritmom

Evolucijsko računanje posnema koncept evolucije oz. naravne selekcije – imamo populacijo osebkov, v kateri okoljski pritiski ustvarijo naravno selekcijo (preživetje najmočnejšega), kar povzroči naraščanje fitnesa populacije. Najprej naključno ustvarimo populacijo osebkov ter definiramo t.i. fitnes funkcijo, ki je ciljna funkcija, s katero bomo ocenili fitnes osebkov. Glede na fitnes najboljše osebkke odberemo kot starše prihodnje generacije, ki jo ustvarimo z mutacijo in / ali rekombinacijo dveh staršev. V naslednjem koraku tako starše kot tudi potomce ovrednotimo s fitnes funkcijo in le najboljše prenesemo v naslednjo generacijo. Vsaka iteracija algoritma ustvari novo generacijo osebkov, vsak osebek pa predstavlja eno možno rešitev. Potek evolucijskega algoritma lahko predstavimo z naslednjo shemo:

inicializiraj populacijo (naključen proces)

osebki (kandidatne rešitve)

ovrednoti vse osebkke (s fitnes funkcijo)

DOKLER ne STOP delaj:

izberi starše iz populacije

ustvari potomce z mutacijo in rekombinacijo staršev

ovrednoti novorojene potomce

zamenjaj nekatere izmed staršev s potomci

Uspeh evolucijskih algoritmov je tako močno odvisen od upravljanja populacije. V splošnem sta v evolucijskem algoritmu dve gonilni sili: selekcija, ki si prizadeva za kakovost in zmanjšuje genetsko variabilnost populacije, in variacija – implementirana kot mutacija in rekombinacija - ki si prizadeva za novosti in povečuje genetsko variabilnost. Zato je uravnoteženje teh dveh sil ključno za uspeh evolucijskega algoritma (Eiben in Schoenauer, 2005).

Gorjanc in Hickey, 2018, sta za rešitev OCS uporabila evolucijski algoritem, in sicer algoritem diferencialne evolucije. Slednji vsebuje naslednje korake (Zamuda in sod., 2008):

- 1) **inicializacija** začetne populacije, kjer naključno definiramo N število osebkov oz. rešitev \mathbf{X}_i^G :

$$\mathbf{X}_{i,G} = \{x_0, x_1, \dots, x_j\}; 0 \leq i < N, 0 \leq j < D, G = 0 \quad (17)$$

; kjer je x_j j -ta neznanka v problemu, N je število osebkov znotraj populacije, D je dimenzija problema, G je generacija.

- 2) **mutacija**: za vsak osebek $\mathbf{x}_{i,G}$, ki ga imenujemo tudi **ciljni vektor** (angl. *target vector*), naključno izberemo dodatne tri osebkke $r1$, $r2$ in $r3$, tako, da so indeksi $r1$, $r2$ in $r3$ različni od i in različni med sabo. Iz njih ustvarimo t.i. **mutiran vektor** (angl. *donor vector*):

$$\mathbf{v}_{i,G} = \mathbf{x}_{r1,G} + F \times (\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G}) \quad (18)$$

; kjer $\mathbf{x}_{r1,G}$ imenujemo **osnovni vektor** (angl. *base vector*), izraz $(\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G})$ pa diferenčni vektor (angl. *difference vector*). F je mutacijski faktor in zavzema vrednosti $[0, 2]$. Slednji nam uteži diferenčni vektor, ki se zatem imenuje utežen diferenčni vektor (angl. *weighted difference vector*).

- 3) **rekombinacija**: z rekombinacijo ciljnega in mutiranega vektorja ustvarimo t.i. **poskusni vektor** (angl. *trial vector*) $u_{i,j,G+1}$:

$$u_{i,j,G+1} = \begin{cases} v_{i,j,G+1} & \text{rand}(0, 1) \leq CR \text{ ali } j = j_{rand} \\ x_{i,j,G} & \text{sicer} \end{cases} \quad (19)$$

; kjer je $\text{rand}_{j,i} \sim U[0, 1]$, j_{rand} pa naključno število iz $[1, 2, \dots, D]$. j_{rand} zagotovi, da $v_{i,G+1} \neq x_{i,G}$.

- 4) **selekcija**: na tej točki ciljni in poskusni vektor ocenimo s fitnes funkcijo f . Če je poskusni vektor boljši od ciljnega, potem preizkusni vektor v populaciji zamenja ciljnega.

$$\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1} & \text{if } f(\mathbf{u}_{i,G+1}) < f(\mathbf{x}_{i,G}) \\ \mathbf{x}_{i,G} & \text{sicer} \end{cases} \quad (20)$$

Vsak osebek je ena rešitev oz. vektor prispevkov vseh živali v optimizaciji. V kontekstu OCS lahko optimiziramo samo korak odbire in prispevkov ali pa (tudi) korak parjenja. Cilj optimizacije je lahko samo minimizacija sorodstva, samo maksimiranje napredka ali pa uravnoteženje obeh ciljev. V vsaki iteraciji optimizacije evlucijski algoritem za vsako rešitev glede na cilj izračuna, kakšen bi bil genetski napredek in / ali skupno sorodstvo populacije, ki bi jo ustvarili glede na predlagane prispevke in / ali parjenja. S ciljno funkcijo te vrednosti ovrednotimo ali pa združimo glede na naš cilj in tako določimo fitnes rešitev. Rešitve oz. osebk, ki našemu cilju ne ustrezajo – npr. presežejo maksimalno skupno sorodstvo - so penalizirane in tako niso odbrani kot starši, niti ne vstopijo v naslednjo generacijo.

1.3 RAZISKOVALNI PROBLEM

Genomska selekcija omogoča večji genetski napredek zaradi skrajšanega generacijskega intervala in večje točnosti zgodnjih selekcijskih odločitev (Schaeffer, 2006). Posledica krajšega generacijskega intervala je hitrejšo obračanje generacij. Ohranjanje oz. optimalno upravljanje z genetsko variabilnostjo postane tako še pomembnejše za trajnostno upravljanje s populacijo in dolgoročno doseganje genetskega napredka. Pri tem še vedno ostaja vprašanje, na kakšno vrednost omejiti skupno sorodstvo oz. kako uravnotežiti ohranjanje genetske variance relativno na doseganje genetskega napredka, da bo scenarij učinkovitejši kot scenarij brez optimiziranja genetskih prispevkov.

Čeprav genomska selekcija omogoča večji genetski napredek, je njena uvedba predvsem v majhnih populacijah otežena zaradi majhnega števila živali za referenčno populacijo in omejenih sredstev. V Sloveniji se nekatere pasemske organizacije vključujejo v mednarodna združenja za skupno napoved genomskih plemenskih vrednosti, druge pa takšne možnosti nimajo. Zato je vprašanje tudi, ali bi z OCS lahko dosegli večjo učinkovitost in genetski napredek slovenske genomskega selekcijskega scenarija in s tem omogočili mednarodno konkurenčnost slovenske selekcije.

Kot predstavljeno obstaja za optimizacijo prispevkov v selekciji veliko različnih algoritmov, ki se razlikujejo v načinu optimizacije, računalniškem času, spominu in možnosti nastavitve ciljev optimizacije. Kot že raziskano tudi za druge primere problemov v genomski selekciji, kot npr. napoved gPV, se lahko rezultati različnih algoritmov razlikujejo. Zato nas je zanimalo, kako primerljivi so algoritmi – specifično deterministično stožčasto programiranje in hevristični evlucijski algoritem – med seboj.

1.4 RAZISKOVALNI CILJI

Splošni cilj naloge je analiza vpliva uporabe optimizacije prispevkov na genetski napredek in učinkovitost selekcije v simulirani populaciji slovenske rjave pasme. To bo doseženo skozi naslednje cilje:

- razvoj orodja za stohastično simulacijo specifičnega rejskega programa s prekrivajočimi generacijami, ki omogoča tudi optimizacijo prispevkov staršev;
- analiza slovenskih rejских programov;
- priprava alternativnih scenarijev selekcije z optimalnimi prispevki (OCS);
- primerjava slovenskega genomskega scenarija in OCS scenarijev: primerjava genetskega napredka, genske variance in učinkovitost selekcije;
- primerjava OCS scenarijev glede na tarčne stopinje: primerjava genetskega napredka, genske variance in učinkovitost selekcije;
- primerjava OCS scenarijev, ki uporabijo determinističen ali hevrističen algoritem za optimizacijo: primerjava genetskega napredka, genske variance in učinkovitost selekcije.

2 METODE IN MATERIALI DELA

Najprej smo razvili orodje za stohastično simulacijo, ki nam je omogočalo nastavitve vseh selekcijskih parametrov in s tem simulacijo slovenske populacije govedi. Testirali smo 11 scenarijev z ali brez optimiziranja prispevkov živali, od tega je en scenarij posnemal genomsko selekcijo v slovenski populaciji s trenutnim režimom odbire živali. Preostalih deset scenarijev je implementiralo optimizacijo prispevkov očetov s: z) petimi različnimi relativnimi težami za maksimiranje genetskega napredka in minimiziranje skupnega sorodstva; ii) dvema različnima optimizacijskima algoritmoma. Testirane scenarije smo primerjali na podlagi doseženega genetskega napredka, izgubljene genske variance in dosežene učinkovitosti selekcije.

2.1 PRIPRAVA ORODJA IN PARAMETROV ZA SIMULACIJO

Najprej smo razvili simulator populacije govedi v selekciji. Uporabnik lahko nastavi vse selekcijske parametre, kar mu omogoča selekcijo specifične populacije govedi. Simulator je zgrajen kot Python ogrodje, ki povezuje programe za izvedbo vseh korakov enega kroga selekcije:

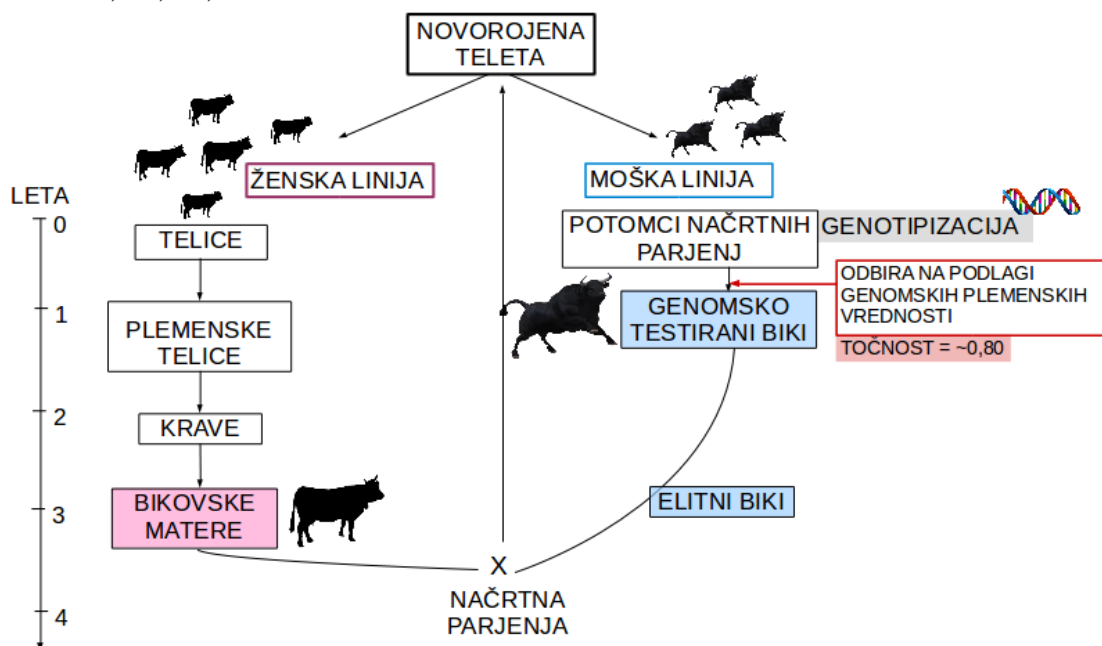
1. AlphaSim (Faux in sod., 2016) za stohastično simulacijo rodovnika in genoma živali,
2. blupf90 (Misztal in sod., 2002) za ocenitev genomskih plemenskih vrednosti,
3. R program, ki uporabi R-ov paket optiSel (Wellman, 2017) ali AlphaMate (Gorjanc in Hickey, 2018) za optimizacijo prispevkov in
4. Python program za določitev staršev nove generacije potomcev.

V naši študiji so bili vsi parametri povzeti po slovenski populaciji rjave pasme govedi, ki šteje ~30.000 aktivnih živali, od tega 10.000 krav. Simulirani genomski podatki vključujejo 10 kromosomov, iz katerih je bilo 10.000 mest izbranih za vzročna mesta za lastnost s heritabiliteto 0,25. Da smo dosegli primerno začetno strukturo populacije, smo najprej smo simulirali 20 generacij selekcijo z naključno odbiro staršev, nato pa še 20 generacij klasične selekcije (selektivna odbira). Temu je sledila simulacija 20 generacij za vsakega izmed testiranih scenarijev.

2.2 TESTIRANI SCENARIJI

V raziskavi smo analizirali deset scenarijev selekcije z ali brez optimizacije prispevkov. Vsi testirani scenariji so implementirali genomsko selekcijsko shemo, v kateri je bila genomska informacija uporabljena le za odbiro očetov, matere pa so bile odbrane po klasični selekcijski

shemi (slika 6). Izmed vseh korakov v selekcijski shemi smo optimizirali le korak odbire in prispevkov v selekcijski poti očetov očetov in očetov mater. V scenariju brez optimizacije prispevkov so bili parametri odbire očetov – število očetov, leta v uporabi in prispevki - povzeti po slovenskih rejских programih (SLO_selekcija). Število odbranih očetov in leta v uporabi so bili stalni in prednastavljeni, prispevki pa so bili le navzgor omejeni. V scenariju slovenske genomske selekcije so bili očetje odbrani le na podlagi njihovih PV, brez obzira na sorodstvo z ostalimi živalmi (selektivna odbira). Preostalih deset scenarijev je vključevalo korak optimizacije prispevkov očetov v skladu z OCS. Število izbranih očetov, leta v uporabi in prispevki tako niso bili stalni. V teh scenarijih so bili očetje odbrani na podlagi njihovih PV in sorodstva z ostalimi živalmi. V optimiziranih scenarijih smo ustvarili pet različnih uravnoveženj maksimiranja genetskega napredka in nadzora skupnega sorodstva, kar smo izrazili s t.i. tarčnimi stopinjami. Poleg tega smo za optimizacijo uporabili dva različna algoritma oz. programa. Skupno smo tako ustvarili deset OCS scenarijev: z vsakim programom, tako optiSel kot AlphaMate, smo testirali smo pet tarčnih stopinj, in sicer 15, 30, 45, 60 in 75.



Slika 6: Testirana genomska selekcija shema. Genomska informacija je bila uporabljena je za odbiro očetov.

2.3 OPTIMIZACIJA PRISPEVKOV

2.3.1 DETERMINISTIČNA OPTIMIZACIJA S STOŽČASTIM PROGRAMIRANJEM DRUGEGA REDA

V OCS scenarijih smo optimizirali le korak odbire in prispevkov očetov očetov in očetov mater, ki so bili odbrani na podlagi genomskega testiranja (slika 6). Prispevke smo optimizirali v vsakem izmed 20 krogov selekcije. Prispevkov mater nismo optimizirali, prav tako nismo optimizirali določitve partnerjev za parjenje. Med potencialne očete smo vključili vse selekcijske kandidate ter že odbrane genomsko testirane bike iz obdobja zadnjih petih let. Da bi skrajšali optimizacijski čas, smo v optimizacijo vključili naključno izbran vzorec 20 % ženskih živali in nato izbrane očete

sorazmerno uporabili na celotni populaciji. Za optimizacijo prispevkov očetov smo uporabili: i) deterministično stožčasto programiranje drugega reda, ki je implementirano v R-ovem paket optiSel (Wellman, 2017); ii) hevristično optimizacijo z evolucijskim algoritmom diferencialne evolucije, ki je implementiran v programu AlphaMate (Gorjanc in Hickey, 2018).

Pri obeh programih so bile želena tarča optimizacije t.i. trigonometrične stopinje, ki opisujejo razmerje med genetskim napredkom in izgubo genetske variabilnosti (slika 7). Podrobneje - merijo kot med rezultatom selekcije s ciljem maksimiranja genetskega napredka brez omejitve za sorodstvo (0 stopinj) in selekcije s ciljem minimiziranja sorodstva brez obzira na genetski napredek (90 stopinj). Manjše stopinje tako pomenijo, da dajemo relativno večji poudarek maksimiranju genetskega napredka, večje pa, da dajemo prednost minimiziranju skupnega sorodstva populacije (Gorjanc in Hickey, 2018). Trigonometrične stopinje lahko izračunamo iz maksimalnega odstotka napredka, ki je normalizirana vrednost za genetski napredek, ali minimalnega odstotka sorodstva, ki je normalizirana vrednost za skupno sorodstvo. Trigonometrične stopinje tako izračunamo kot:

$$\cos(\text{maksimalni odstotek napredka} / 100) \times 180/\pi \quad (21a)$$

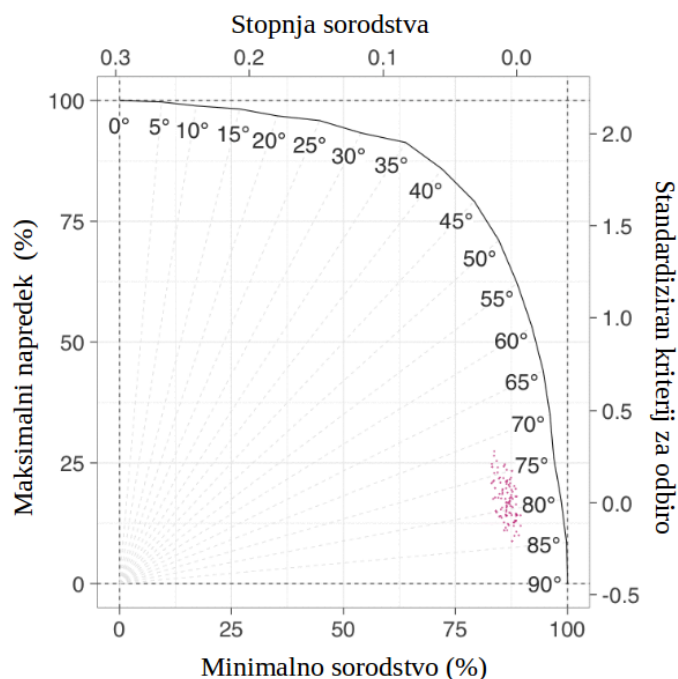
ali:

$$\sin(\text{minimalen odstotek sorodstva} / 100) \times 180/\pi. \quad (21b)$$

V vsakem krogu selekcije je optimizacija potekala v treh stopnjah:

- 1) optimizacija za maksimiranje genetskega napredka, iz česar smo pridobili maksimalen odstotek napredka;
- 2) optimizacija za minimiziranje sorodstva, iz česar smo pridobili minimalen odstotek sorodstva;
- 3) optimizacija za tarčne stopinje, pri čemer je šlo za maksimiranje genetskega napredka glede na omejitve sorodstva izračunanega na podlagi 1), 2) in tarčnih stopinj.

AlphaMate v vsaki iteraciji algoritma s ciljno funkcijo preveri vse rešitve v skladu z nastavljenimi stopinjami. Program izračuna skupno sorodstvo nastale populacije in genetski napredek, ciljna funkcija pa preveri, ali so stopinje pod dovoljenimi. Rešitve, ki tega pogoja ne izpolnjujejo, so penalizirane in tako izločene iz boja za vstop v naslednjo generacijo rešitev. OptiSel pa pred začetkom optimizacije na podlagi izračuna maksimalnega dovoljenega sorodstva iz 3) nastavi omejitev za sorodstvo in nato maksimira genetski napredki ob omenjeni omejitvi po enačbah v (11) oz. (19). V raziskavi smo simulirali pet scenarijev uravnotežene selekcije za vsakega izmed programov optiSel in AlphaMate s tarčnimi stopinjami 15, 30, 45, 60 in 75.



Slika 7: Prikaz trigonometričnih stopinj kot razmerje med maksimalnim genetskim napredkom in minimalnim sorodstvom (vir: Gorjanc in Hickey, 2018).

2. 4 PRIMERJALNI PARAMETRI

Po simulaciji smo testirane scenarije med seboj primerjali glede na dosežen letni genetski napredek, izgubo genetske variance in učinkovitost selekcije. Letni genetski napredek smo pridobili z regresijo povprečne PV na generacijo. Genetski napredek smo za vsak scenarij standardizirali tako, da je imel povprečje 0, izražen je pa bil v enotah standardnega odklona v prvi generaciji primerjave, t.j. generacija 40. Gensko varianco smo standardizirali tako, da je imela v prvi generaciji primerjave vrednost 1. Učinkovitost selekcije smo definirali kot naklon regresije genetskega napredka na izgubo genskega standardnega odklona. Tako genetski napredek kot tudi genski standardni odklon sta bila standardizirana z genskim standardnim odklonom v prvi generaciji primerjave. Učinkovitost selekcije tako izraža, kakšen genetski napredek bi dosegli (v enotah genskega standardnega odklona), če bi se vsa genetska variabilnost pretvorila v genetski napredek.

3 REZULTATI IN RAZPRAVA

Pri testiranih scenarijih smo primerjali genetski napredek, gensko varianco in učinkovitost selekcije po 20 generacijah selekcije. Najprej smo med seboj primerjali scenarij slovenske genomske selekcije in scenarije z optimiziranimi prispevki (OCS scenariji). Ugotovili smo, da lahko z optimizacijo prispevkov očetov v slovenski populaciji govedi dosežemo večji genetski napredek kot genomska selekcija s trenutnim režimom odbire, pri čemer se genetski napredek načeloma večja z manjšanjem tarčnih stopinj. Nasprotno temu smo ugotovili, da scenariji z večjimi stopinjami, t.i. večjo težo na minimiziranju skupnega sorodstva, izgubijo manj genetske variabilnosti, prav tako pa je pri ohranjanju genetske variabilnosti uspešen tudi slovenski genomski selekcijski scenarij. Pri

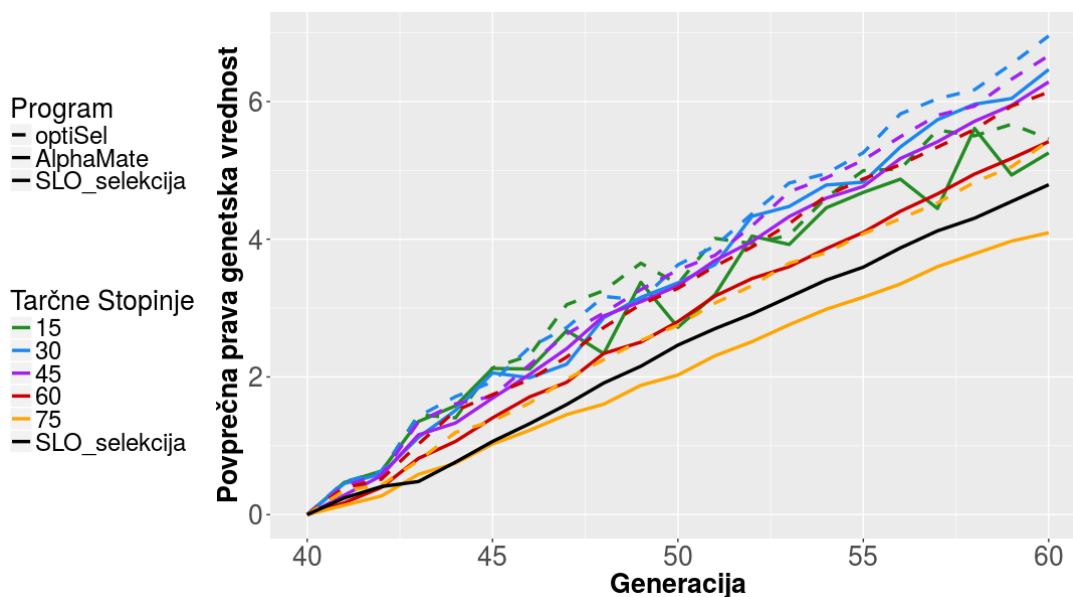
analizi učinkovitosti smo ugotovili, da scenariji z največjim genetskim napredkom niso najbolj učinkoviti ter da lahko z optimizacijo prispevkov očetov marginalno izboljšamo učinkovitost slovenske genomske selekcije. Nato smo med seboj primerjali še rezultate dveh optimizacijskih programov, in sicer optiSel (deterministična optimizacija) in AlphaMate (hevristična optimizacija). Ugotovili smo, da AlphaMate pri istih tarčnih stopinjah prinaša manj genetskega napredka, vendar pa ohrani več genetske variabilnosti kot optiSel. Ker je učinkovitost pod močnih vplivov genetske variabilnosti, so AlphaMate scenariji tudi bolj učinkoviti.

3.1 GENETSKI NAPREDEK

Rezultati za genetski napredek so prikazani na sliki 8. V generaciji 60 je genetski napredek testiranih scenarijev med 4,09 in 6,95 enot. Če najprej primerjamo rezultate slovenske genomske selekcije in selekcije z optimalnimi prispevki s slovensko genomsko selekcijo, vidimo, da lahko z optimizacijo prispevkov za 45 % povečamo genetski napredek. Genetski trend slovenske genomske selekcije je na sliki 8 prikazan s črno i prikazuje potencial ob obstoječem režimu odbire, t.j. 0,247 enot genetskega napredka letno (4,79 v generaciji 60). Z optimizacijo prispevkov očetov lahko genetski napredek povečamo za 9,7 do 45,1 %.

Pri primerjavi OCS scenarijev z različnimi tarčnimi stopinjami smo ugotovili, da genetski napredek narašča z manjšanjem tarčnih stopinj. Slednje je bilo pričakovano, saj se z manjšanjem stopinj večja relativna teža za maksimiranje genetskega napredka pred minimiziranjem skupnega sorodstva. Posledično izmed OCS scenarijev najmanj genetskega napredka prinaša AlphaMate75, in sicer 0,213 (standardni odklon = 0,0122) enot letnega genetskega napredka, v generaciji 60 pa doseže 4,09 enot. Največ napredka prinaša optiSel30, in sicer 0,340 enot (6,95 enot v generaciji 60) kar je 60 % več kot AlphaMate75. Čeprav je teža za genetski napredek največja pri 15 stopinjah, pa ta scenarija – ne optiSel15 niti AlphaMate15 – ne dosežeta največjega genetskega napredka. Navkljub začetni superiornosti teh scenarijev, predvsem optiSel15, začne letni genetski napredek po približno 10 generacijah padati (pri AlphaMate15 še prej), scenarija pa izgubita relativno prednost pred ostalimi. Razlog za to je velika izguba genetske variabilnosti, ki onemogoča dolgoročni genetski napredek. Slednje je posledica močnega pritiska na odbiro najboljših posameznikov, zato je v tema scenarijema število očetov zelo majhno. Medtem, ko je pri tarči 75 stopinj povprečno število očetov na generacijo 32 v optiSel in 25 v AlphaMate scenariju, to pri 15 stopinjah znaša le 4 v optiSel in 3 v AlphaMate scenariju. V določeni generaciji so v optiSel15 in AlphaMate15 tako vsi novorojeni osebki potomci nekaj očetov, kar povzroči velika nihanja v povprečni genetski vrednostmi med generacijami - standardni odklon je 0,0282 v optiSel15 in 0,0390 v AlphaMate15. Ker pa so očetje odbrani na podlagi plemenske vrednosti, točnost katere pa je ~0,6, lahko odbira ni dejansko optimalna. Zaradi velikega števila potomcev lahko tako že en napačno izbran oče povzroči padec povprečne genetske vrednosti populacije v določeni generaciji. Kot posledica manjšega števila očetov pri manjših stopinjah se standardni odklon povprečne genetske vrednosti večja s padajočimi stopinjami in narašča od 0,0163 in 0,0122 pri optiSel75 in AlphaMate75 do 0,0282 in 0,0390 pri optiSel15 in AlphaMate15.

Če nato primerjamo še oba programa, opazimo, da prihaja do razlik med rezultati, pridobljenih s stožčastim programiranjem (optiSel) in hevrističnim evolucijskim algoritmom (AlphaMate). Opazimo, da pri istih tarčnih stopinjah AlphaMate prinaša manjši genetski. Relativna prednost optiSel rešitev pred AlphaMate se načeloma manjša z manjšanjem stopinj in je največja pri največjih stopinjah, t.j. 75, in sicer 33 % (5,43 optiSel in 4,09 AlphaMate), najmanjša pa pri najmanjših stopinjah, t.e. 15, in sicer 4,0 % (5,46 optiSel in 5,25 AlphaMate).



Slika 8: Povprečen genetski napredek testiranih scenarijev po generacijah. Črtkana črta = optiSel15 – 75: selekcija z optiSel-optimiziranimi prispevki s tarčo 15 – 75 stopinj, polna črta = AlphaMate15 – 75: selekcija z AlphaMate-optimiziranimi prispevki s tarčo 15 – 75 stopinj, SLO_selekcija = slovenski genetski selekcijski scenarij s trenutnim režimom odbire.

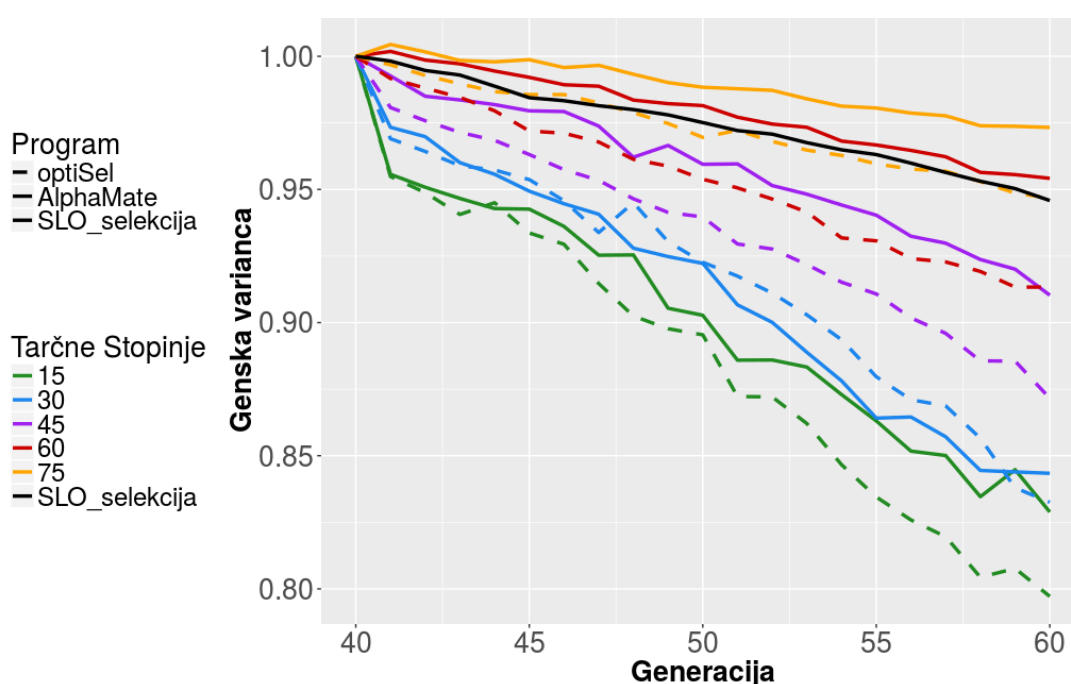
3.2 GENETSKA VARIABILNOST

Rezultati za izgubo genetske variabilnosti, merjene kot genska varianca, so prikazani na sliki 9. Tekom 20 generacij simulacije testirani scenariji izgubijo med 2,7 % in 20,3 % prvotne genske variance. Če zopet najprej primerjamo slovenski genetski scenarij z OCS scenariji, vidimo, da je slovenski scenarij relativno uspešen pri ohranjanju genetske variabilnosti skozi generacije. Slovenski scenarij izgubi 5,4 % prvotne genske variance in je primerljiv z najboljšim optiSel scenarijem (optiSel75), ki izgubi 5,3 % genske variance in drugim najboljšim AlphaMate scenarijem (AlphaMate 60), ki izgubi 4,6 % prvotne genske variance.

V skladu s pričakovanji imajo scenariji z večjimi tarčnimi stopinjami – ki pomenijo večjo težo na minimiziranju skupnega sorodstva – manjšo izgubo genske variance. Manjše stopinje pomenijo manjšo težo na ohranjanju genetske variabilnosti v optimizaciji, kar povzroči večjo izgubo genske variance. Tako znotraj optiSel in AlphaMate scenarijev najmanj genske variance izgubita scenarija s 75 stopinjami, in sicer izgubi optiSel75 izgubi 5,3 % prvotne genske variance, AlphaMate75 pa 2,7 %. Izguba variance nato monotonno narašča s padanjem stopinj - največ genske variance tako izgubita scenarija s 15 stopinjami, in sicer optiSel15 v 20 generacijah izgubi 20,3 % prvotne genske

variance, kar je 3,6-krat več kot optiSel75, AlphaMate15 pa izgubi 17,1 % genske variance, kar je 6,3-krat več kot AlphaMate75.

Pri primerjanju različnih algoritmov za optimizacijo, ki sta implementirana v optiSel in AlphaMate, tudi tukaj opazimo, da med njima prihaja do razlik. Tukaj vidimo obratno zgodbo kot pri doseganju genetskega napredka, saj je AlphaMate pri vseh tarčnih stopinjah bolj uspešen pri ohranjanju genetske variabilnosti kot optiSel - slednje pomeni, da je AlphaMate bolj konservativen. Dva AlphaMate scenarija sta bolj uspešna od optiSel scenarijev, in sicer AlphaMate75 in AlphaMate60, ki izgubita 2,7 % in 4,6 % prvotne genske variance. Slednja scenarija sta tudi edina bolj uspešna pri ohranjanju genske variance kot slovenski genomski scenarij.



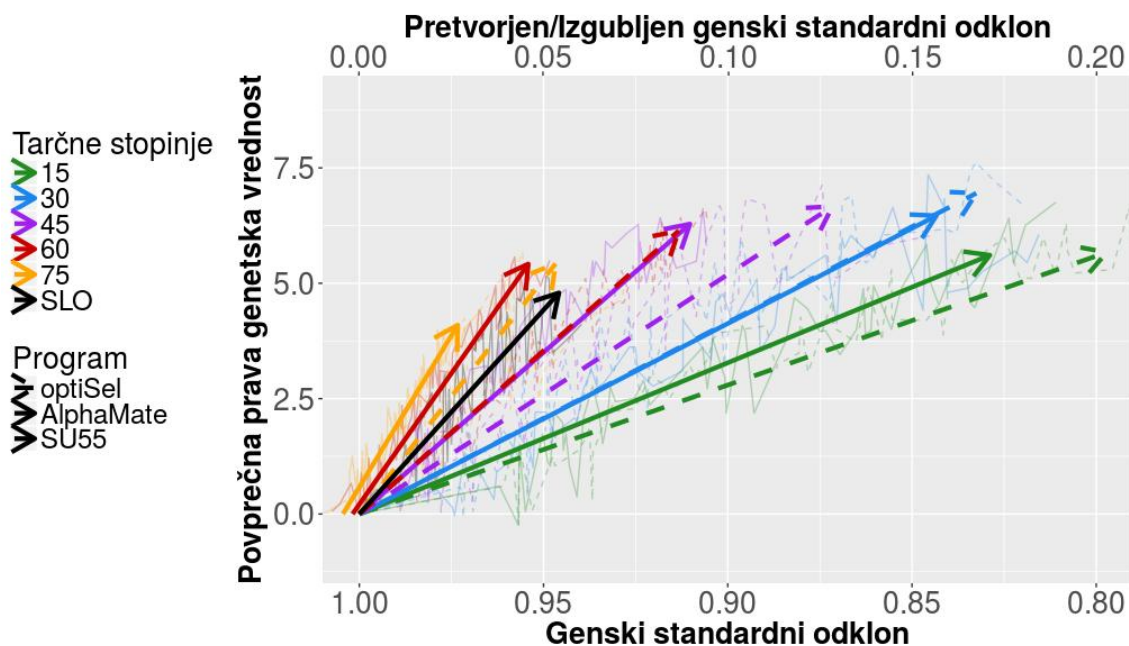
Slika 9: Povprečna genska varianca testiranih scenarijev po generacijah. Črtkana črta = optiSel15 – 75: selekcija z optiSel-optimiziranimi prispevki s tarčo 15 – 75 stopinj, polna črta = AlphaMate15 – 75: selekcija z AlphaMate-optimiziranimi prispevki s tarčo 15 – 75 stopinj, SLO_selekcija = slovenski genomski selekcijski scenarij s trenutnim režimom odbire.

3.3 UČINKOVITOST SELEKCIJE

Na podlagi genetskega napredka in izgube genske variance – oz. genskega standardnega odklona – smo izračunali tudi učinkovitost selekcije za vse scenarije (slika 10). Učinkovitost scenarijev se giblje med 16 in 61 enot. Vidimo, da lahko z optimizacijo prispevkov očetov izboljšamo učinkovitost slovenske genomske selekcije, in sicer do 33 %. Učinkovitost slovenske genomske selekcije je 46 enot, boljšo učinkovitost pa dosežejo scenariji AlphaMate75, AlphaMate60 in optiSel75 z učinkovitostjo 61, 52 in 51 enot. Opazimo tudi, da čeprav sta bila scenarija optiSel75 in slovenski genomski selekcijski scenarij po izgubi genske variance zelo primerljiva, pa se po učinkovitostih razlikujeta, in sicer ima slovenski genomski scenarij 10 % manjšo učinkovitost. Razlika je posledica dejstva, da navkljub isti izgubi genetske variabilnosti, scenarij optiSel75 doseže večji genetski napredek.

Iz slike 10 je razvidno, da učinkovitost OCS scenarijev narašča z naraščanjem stopinj. Pri tem opazimo, da vrstni red scenarijev tesno sledi vrstnemu redu scenarijev na podlagi genske variance – t.j. scenariji, ki izgubijo manj genske variance, so bolj učinkoviti. V nasprotju s tem pa vrstni red ne sledi vrstnemu redu na podlagi genetskega napredka. Slednje nakazuje, da na razlike v učinkovitosti scenarijev vplivajo predvsem razlike v izgubi genske variance. Če podrobneje pogledamo rezultate, vidimo, da so razlike v izgubi genske variance (2,7 – 20,3 %) med scenariji večje kot pa razlike v genetskem napredku (4,09 do 6.95 enot) – posledično je izguba genske variance gonilo za selekcijsko učinkovitost. Najbolj učinkovit je tako scenarij AlphaMate75, ki doseže učinkovitost 61 enot. Najmanj učinkovit je scenarij optiSel15, ki doseže učinkovitost 16 enot, kar je 3,8-krat manj kot najbolj učinkovit scenarij. Majhna učinkovitost scenarijev z večjimi tarčnimi stopinjami je posledica velike izgube genske variance oz. standardnega odklona. Prav tako vidimo, da čeprav je optiSel30 imel največji genetski napredek, pa ni med najbolj učinkovitimi scenariji, saj porabi tudi relativno veliko genske variance oz. standardnega odklona. Nasprotno temu je AlphaMate75, ki je dosegel najmanjši genetski napredek, hkrati tudi najbolj učinkovit.

Če nazadnje primerjamo še optiSel in AlphaMate scenarije, vidimo, da AlphaMate scenariji pri istih tarčnih stopinjah dosegajo večjo – ali enako - učinkovitost selekcije. Slednje sledi ugotovitvi, da na učinkovitost v tej simulaciji bolj kot genetski napredek vpliva izguba genske variance. Ker so AlphaMate scenariji bolj konservativni in izgubijo manj genske variance, je posledično tudi njihova učinkovitost večja. Do največjih razhajanj med programoma prihaja pri 60 stopinjah, kjer optiSel doseže učinkovitost 34 enot, AlphaMate pa 53 % več, in sicer 52 enot. Programa sta najbolj usklajena pri 30 tarčnih stopinjah, saj optiSel30 doseže 23, AlphaMate30 pa 22 enot.



Slika 10: Povprečna učinkovitost selekcije testiranih scenarijev. Črtkana črta = optiSel15 – 75: selekcija z optiSel-optimiziranimi prispevki s tarčo 15 – 75 stopinj, polna črta = AlphaMate15 – 75: selekcija z AlphaMate-optimiziranimi prispevki s tarčo 15 – 75 stopinj, SLO_selekcija = slovenski genomski selekcijski scenarij s trenutnim režimom odbire.

4 ZAKLJUČKI

Selekcija stremi h genetskemu izboljšanju populacije. Poleg maksimiranja genetskega napredka je v za trajnostni uspeh selekcije in dolgoročni genetski napredek ključno tudi ohranjanje genetske variabilnosti za lastnosti v populaciji. Slednje je še posebno pomembno v genomski selekcije, ki zaradi načina napovedi PV omogoča zgodnjo odbiro in hitrejšo obračanje generacij. Selekcija z optimalnimi prispevki združuje cilj maksimiranja genetskega napredka z omejitvijo skupnega sorodstva populacije. Razvoj rešitve za selekcijo z optimalnimi prispevki se je razvijal od linearnega celoštevilskega programiranja, preko rešitve z Lagrangeovimi multiplikatorji, do semidefinitnega programiranja, stožčastega programiranja drugega reda in nazadnje hevrističnih algoritmov.

V naši raziskavi smo želeli analizirati uspeh selekcije z optimalnimi prispevki v slovenski populaciji govedi in rezultate primerjati z uspehom slovenske genomske selekcije s trenutnim realnim režimom odbire. Poleg tega smo želeli primerjati rezultate dveh različnih optimizacijskih algoritmov: determinističnega in hevrističnega. Razvoj orodja za stohastično simulacijo nam je omogočal nastavitve vseh selekcijskih parametrov in s tem simulacijo genomske selekcije v slovenski populaciji govedi. Poleg scenarija, ki posnema slovensko genomsko selekcijo, smo testirali deset uravnoteženih scenarijih z različnim ciljem, t.e. tarčne stopinja, in različnim optimizacijskim algoritmom. V vseh scenarijih smo optimizirali le korak prispevkov očetov očetov in očetov mater. Poleg tega smo optimizacijo izvedli tudi z dvema različnima algoritmoma, in sicer determinističnim stožčastim programiranjem implementiranim v optiSel programu in hevrističnim evolucijskim algoritmom implementiranim v AlphaMate.

Ugotovili smo, da lahko z optimizacijo prispevkov očetov povečamo genetski napredek slovenske genomske selekcije za 45 %. Kot pričakovano smo ugotovili, da večja teža na maksimiranju genetskega napredka, i.e. manjše tarčne stopinje, načeloma prinaša večji genetski napredek. Izjema je OCS scenarij z najmanjšo težo na ohranjanju genetske variabilnosti, ki začne sčasoma izgubljati genetski napredek zaradi izgube genetske variabilnosti. To potrjuje, da za dolgoročni uspeh selekcije ni pomembna le odbira najboljših živali za večji genetski napredek, ampak je pomembna tudi sorodnost teh živali, ki zagotavlja ohranjanje genetske variabilnosti v populaciji. Ugotovili smo tudi, da prihaja do razlik med optimizacijskima algoritmoma, in sicer smo z determinističnim programiranjem v optiSel scenarijih pri istih tarčnih stopinjah dosegli večji genetski napredek kot z AlphaMate.

Pri primerjavi izgube genske variance scenarijev smo ugotovili, da je slovenski genomski scenarij relativno uspešen pri ohranjanju genetske variabilnosti v populaciji, saj so bili rezultati primerljivi z OCS scenariji, ki ohranijo največ genske variance. Rezultati so pokazali tudi, da je izguba genske variance večja pri manjših tarčnih stopinjah. Slednje je pričakovano, saj manjše stopinja pomenijo večjo težo na ohranjanje genetske variabilnosti. Pri primerjavi algoritmov smo ugotovili, da hevrističen evolucijski algoritem, implementiran v AlphaMate, ohrani več genske variance pri istih tarčnih stopinjah in je tako bolj konservativen algoritem.

Na koncu smo rezultate za genetski napredek in gensko varianco povezali v merilo učinkovitosti selekcije. Ugotovili smo, da so lahko z optimizacijo prispevkov za do 33 % izboljšamo učinkovitost slovenske genomske selekcije. Učinkovitost OCS scenarijev je v splošnem naraščala z naraščanjem stopinj, kar je posledica manjše izgube genske variance. Ugotovili smo tudi, da razvrstitev scenarijev po učinkovitost tesno sledi razvrstitvi glede na izgubo genske variance. Slednje nakazuje, da imajo v naši simulaciji razlike v izgubi genske variance večji vpliv na selekcijsko učinkovitost kot pa razlike v genetskem napredku. Ugotovili smo tudi, da scenariji z največjim genetskim napredkom niso najbolj učinkoviti, ampak je najbolj učinkovit scenarij z najmanjšim genetskim napredkom, saj izgubi najmanj genske variance.

V raziskavi smo pokazali možnost uporabe OCS v slovenski populaciji govedi. Slednja lahko prinese kar za 45 % večji genetski napredek, in za 33 % izboljša učinkovitost selekcije. Tako na genetski napredek, gensko varianco kot tudi učinkovitost selekcije lahko vplivamo s uravnoteženjem selekcijskih ciljev, kar izrazimo s t.i. tarčnimi stopinjami. Odločitev o uravnoteženju je v rokah rejских organizacij – kakšno tveganje so pripravljene sprejeti v smislu izgube genetske variabilnosti oz. kakšno »ceno« so pripravljene plačati za doseg genetskega napredka. Vprašanje je tudi v dolgoročnosti njihovih ciljev – medtem ko genetski napredek predstavlja merilo za kratkoročni uspeh selekcije (praktično le v naslednji generaciji), pa učinkovitost selekcije poda informacijo o dolgoročnem uspehu selekcije. V naši simulaciji je bil najbolj učinkovit scenarij z najmanjšim genetskim napredkom. Navkljub največji učinkovitosti pa morda takšen scenarij ni sprejemljiv za rejske organizacije, saj zaradi majhnega genetskega napredka ne omogoča konkurenčnosti selekcije, hkrati pa je morda tudi preveč konservativen za v smislu ohranjanja genetske variance za določeno populacijo.

Ugotovili smo tudi, da prihaja do razlik med optimizacijskima algoritmoma pri istih tarčnih stopinjah. Slednje je tudi posledica razlik v preverjanju dosege cilja. AlphaMate namreč v vsaki iteraciji algoritma s ciljno funkcijo preveri vse rešitve, ali so njihove stopinje pod dovoljenimi. Rešitve, ki tega pogoja ne izpolnjujejo, so penalizirane in tako izločene iz boja za vstop v naslednjo generacijo rešitev. OptiSel pa pred začetkom optimizacije na podlagi maksimalnega napredka in minimalnega sorodstva izračuna maksimalno dovoljeno sorodstvo in nato maksimira genetski napredki ob omejitvi za maksimalno stopnjo sorodstva. Tako na koncu optimizacije prihaja do razlik v realiziranih tarčnih stopinjah med AlphaMate in optiSel scenariji, ki so potencialni vzrok razlik v genetskem napredku in učinkovitosti.

Rezultati te raziskave bodo lahko v pomoč slovenskim rejским organizacijam pri pripravi rejских programov. V prihodnje bi lahko raziskavo nadgradili z optimizacijo korak parjenja oz. določitve partnerja za parjenje, za kar pa bi za tako veliko populacijo potrebovali večje računalniške kapacitete.

5 VIRI IN LITERATURA

- Aickelin U. in Clark A. 2011. Heuristic optimisation. *Journal of the Operational Research Society*, 62, 2: 251 - 252.
- Boyd S., Vandenberghe L. 2004. *Convex Optimization*. New York, ZDA, Cambridge University Press: 716 str.
- Dattorro J. 2005. *Convex Optimization & Euclidian Distance Geometry*. Palo Alto, Kalifornija, ZDA, Meboo Publishing USA: 690 str.
- Eiben A. in Schoenauer M. 2005. Evolutionary Computing. *Information Processing Letters*. 82, 1: 1-6.
- Falconer D.S., Mackay, T.F.C. 1996. *Introduction to Quantitative Genetics*. 4. izdaja. Harlow, UK, Longman: 464 str.
- Faux A.-M., Gorjanc G., Gaynor R.C., Battagin M., Edwards S.M., Wilson D.L., Hearne, S.J., Gonen S., Hickey J.M. 2016. AlphaSim: Software for Breeding Program Simulation. *Plant Genome*, 9, 3.
- Gorjanc G., Hickey J.M. 2018. AlphaMate: a program for optimising selection, maintenance of diversity, and mate allocation in breeding programs. *bioRxiv* 250837.
- Grundy B., Villanueva B., Woolliams J.A. 1998. Dynamic selection procedures for constrained inbreeding and their consequences for pedigree development. *Genetics Research*, 72, 2: 159–168.
- James J.W., McBride G. 1958. The spread of genes by natural and artificial selection in closed poultry flock. *Journal of Genetics*, 56, 55–62.
- Kinghorn B.P., Shepherd R.K. 1999. RK: Mate selection for the tactical implementation of breeding programs. *Proceedings of the Association for the Advancement of Animal Breeding and Genetics*, 13: 130 – 133.
- Korošec P. 2004. Magistrska naloga: Metahevristično reševanje optimizacijskega problema s kolonijami mravelj. Ljubljana, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko: 80 str.
- Lindgren D., Libby W. S., Bondesson F. L. 1989. Deployment to plantations of numbers and proportions of clones with special emphasis on maximizing gain at a constant diversity. *Theoretical and Applied Genetics*, 77, 6: 825–831.
- Lush J.L. 1945. *Animal Breeding Plans*. 3. izdaja, Ames, Iowa, ZDA, Iowa State University Press: 442 str.
- Maringer D. 2005. *Portfolio Management with Heuristic Optimization*. New York, ZDA, Springer-Verlag New York: 38 – 76.
- Meuwissen T. H. E. 1997. Maximizing the response of selection with a predefined rate of inbreeding. *Journal of Animal Science*, 75, 4: 934–940.

- Meuwissen T.H.E., Hayes B.J., Goddard M.E. 2001. Prediction of Total Genetic Value Using Genome-Wide Dense Marker Maps. *Genetics*, 157, 4: 1819–1829.
- Misztal I., Tsuruta S., Strabel T., Auvray B., Druet T., Lee D.H. 2002. BLUPF90 and re-lated programs (BGF90). V: Proc. 7th World Congress on Genetics Applied to Livestock Production. WCGALP, Montpellier, France, 19-23 avg. 2002. Montpellier, WCGALP: 1-2.
- Pearl J. 1984. Heuristics: Intelligent search strategies for computer problem solving. Boston, Massachusetts, ZDA, Addison-Wesley Longman Publishing Company: 382 str.
- Pong-Wong R., Woolliams J.A. 2007. Optimisation of contribution of candidate parents to maximise genetic gain and restricting inbreeding using semidefinite programming. *Genetics Selection Evolution*, 39, 1: 3-25.
- Pryce J.E., Daetwyler H.D. 2012. Designing dairy cattle breeding schemes under genomic selection: a review of international research. *Animal Production Science*, 52: 107–114
- Ramana M. in Goldman A. J. 1995. Some Geometric Results in Semidefinite Programming. *Journal of Global Optimization*, 7, 1: 33–50.
- Rendel J.M., Robertson A. 1950. Estimation of genetic gain in milk yield by selection in a closed herd of dairy cattle. *Journal of Genetics*, 50, 1: 1–8.
- Schaeffer L. R., 2006. Strategy for applying genome-wide selection in dairy cattle. *Journal of Animal Breeding and Genetics*, 123, 4: 218-223.
- Toro M., Pérez-Enciso M. 1990. Optimization of selection response under restricted inbreeding. *Genetics Selection Evolution*, 22, 1: 93 - 107.
- Wellmann R. 2017. optiSel: optimum contribution selection and population genetics. R package version 0.7.1. <https://cran.r-project.org/web/packages/optiSel/optiSel.pdf>.
- Woolliams J.A., Berg P., Dagnachew B.S., Meuwissen T.H.E. 2015. Genetic contributions and their optimization. *Journal of Animal Breeding and Genetics*, 132, 2: 89–99.
- Woolliams J.A., Bijma P., Villanueva, B. 1999. Expected Genetic Contributions and Their Impact on Gene Flow and Genetic Gain. *Genetics* 153, 2: 1009–1020.
- Wray N.R., Thompson R. 1990. Prediction of rates of inbreeding in selected populations. *Genetics Research*. 55, 1: 41–54.
- Wright S. 1921. Systems of Mating. II. the Effects of Inbreeding on the Genetic Composition of a Population. *Genetics* 6, 2: 124–143.
- Yamashita M., Mullin T. M., Safarina, S. 2015. An efficient second-order cone programming approach for optimal selection in tree breeding. arXiv:1506.04487v2.
- Zamuda A., Brest J., Bošković B., Žumer V. 2008. Diferencialna evolucija za večkriterijsko optimizacijo s samoprilagajanjem in z lokalnim preiskovanjem SQP. V: Zbornik 17. mednarodne elektrotehniške in računalniške konference ERK, Portorož, Slovenija, 29. sep. - 1. okt. 2008. Zajc B., Trost A. (ur.). Ljubljana : IEEE Region 8, Slovenska sekcija IEEE:103-106.

Obšteter, J. Selekcija z optimalnimi prispevki v slovenskih rejских programih za govedo.
Ljubljana, Univ. v Ljubljani, Biotehniška fakulteta, 2018

Zanakis S. H. in Evans, J. R. 1981. Heuristic "Optimization": Why, When, and How to Use It. Interfaces, 11, 5: 84-91.