

Optimizacijske metode

Splošno

Optimizacija je postopek, pri katerem želimo določiti parametre kompleksnega sistema, tako da bi sistem deloval kar najboljše, oziroma da bi deloval optimalno. Beseda "sistem" je uporabljena za zelo širok krog pojmov: za sintezo spojine, za kemijsko reakcijo, za analizo metodo, za lastnost izdelka, za kemijski proces in podobno. Parametri, ki jih želimo z optimizacijo določiti, so vrednosti fizikalno-kemijskih spremenljivk x_i , ki sistem opisujejo. Optimizacijski postopek omogoči, da iz katereregakoli možnega stanja sistema $X_s = (x_{s1}, x_{s2}, \dots, x_{sm})$ s pomočjo cenilne funkcije $f(X_s)$ izračunamo naslednje stanje sistema $X_{s+1} = (x_{s+1,1}, x_{s+1,2}, \dots, x_{s+1,m})$, ki je glede na rezultat cenilne funkcije $f(X_{s+1})$ predvidoma boljši kot X_s . V primeru, da $f(X_{s+1})$ ni boljši od $f(X_s)$, mora optimizacijski postopek natančno predpisati vse potrebne korake, s katerimi bodisi ugotovimo, da smo dosegli optimum, ali pa dobimo boljši X_{s+1} . Hkrati moramo poznati tudi kriterije, ki so potrebni za to, da optimizacijo ustavimo.

Poleg dobrega poznavanja omejitev celotnega sistema (območij vseh neodvisnih spremenljivk x_i), je najpomembnejši korak optimizacijskega postopka določitev kvalitetne cenilne funkcije. Cenilni funkciji pravimo tudi "optimizacijski kriterij" (OK) ali "fitness-function" (FF). Cenilna funkcija je lahko enostavna, odvisna od velikosti ene same lastnosti, lahko pa je zelo komplicirana nelinearna funkcija večjega števila lastnosti. Enostavna cenilna funkcija je lahko numerična vrednost lastnosti izdelka, ki se da preprosto izmeriti, kot je npr. odstotek absorpcije UV žarkov zaščitne kreme ali hitrost strjevanja lepila. Sestavljena cenilna funkcija mora upoštevati več različnih lastnosti izdelka (ali sistema), ki velikokrat delujejo druga proti drugi. Zato je postavitve dobre cenilne funkcije, ki pravilno razvršča izdelke po kvaliteti z eno samo realno vrednostjo, lahko zelo težavna naloga.

Da za vsak izdelek dobimo numerično vrednost cenilne funkcije, moramo opraviti določeno število meritev ali pa imeti na voljo matematični model (funkcijo), s katerim lastnosti izdelka (sistema) napovemo. V primeru, da model za izračun cenilne funkcije imamo [$OK = f(x_1, x_2, \dots, x_m)$], se da optimizacijo avtomatizirati. V nasprotnem primeru moramo vse eksperimente, ki jih optimizacijski potopek predpiše, narediti, nato izmeriti lastnosti izdelka in na koncu izračunati cenilno funkcijo. Tak način optimizacije je zelo zamuden. Hkrati je treba vedeti, da tudi modelov, ki bi dobro opisovali sisteme ali izdelke, ni lahko dobiti.

Za optimizacijo je najbolj pomembna merska natančnost merjenja lastnosti, ki jih zajema cenilna funkcija. Razpršenost rezultatov (standardni odklik) določa kriterij za ustavitve optimizacijskega postopka. Čim bolj natančne so meritve lastnosti izdelkov in meritve njegovih spremenljivk, tem bližje optimumu lahko pridemo.

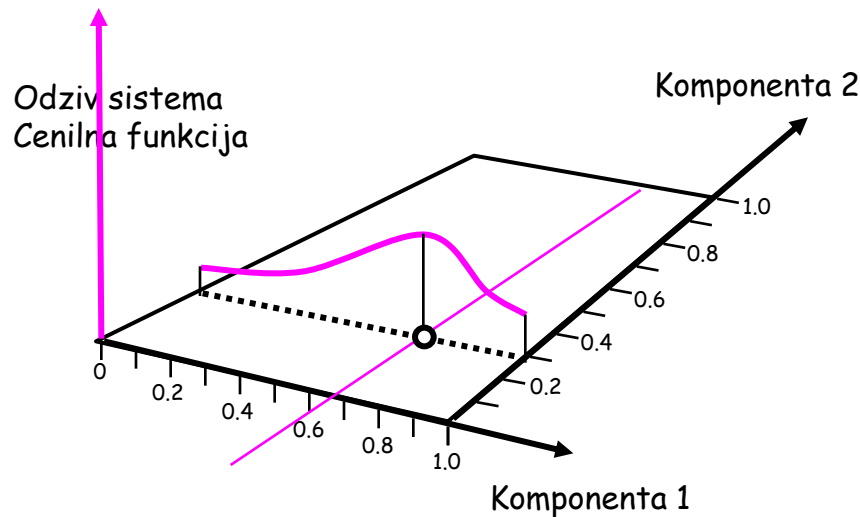
V nadaljevanju bomo predpostavljali, da rezultate cenilne funkcije za vsak izdelek poznamo (so izmerjeni ali izračunani). V primeru (stran 6), ki ga bomo obdelali bolj podrobno, je prikazan način, kako s kombinacijo meritev več lastnosti pridemo do primerne cenilne funkcije oziroma primernega optimizacijskega kriterija.

Optimizacija z metodo posameznih spremenljivk po načelu "eno-po-eno" ali "one-at-the time"

To je optimizacija, pri kateri držimo vse spremenljivke *razen ene* konstantne, in je v praksi zelo razširjena. Izbrano spremenljivko spreminjamo toliko časa, da dobimo najboljši rezultata in potem pri tej vrednosti prve pričnemo spreminjati drugo spremenljivko. Ta izredno priljubljeni način optimizacije je v osnovi slab. Vodi kvečejmu do lokalnih izboljšanj in lahko do optimuma na širšem področju pripelje zgolj slučajno. Verjetnost, da se to zgodi, je izredno majhna in močno pada z naraščajočim številom spremenljivk. Primer take "optimizacije" je prikazan spodaj.

Optimiziramo enostaven dvo-komponenti sistem.

Koncentracije na sliki so podane v deležih med maksimalnima in minimalnima možnima koncentracijama obeh komponent.



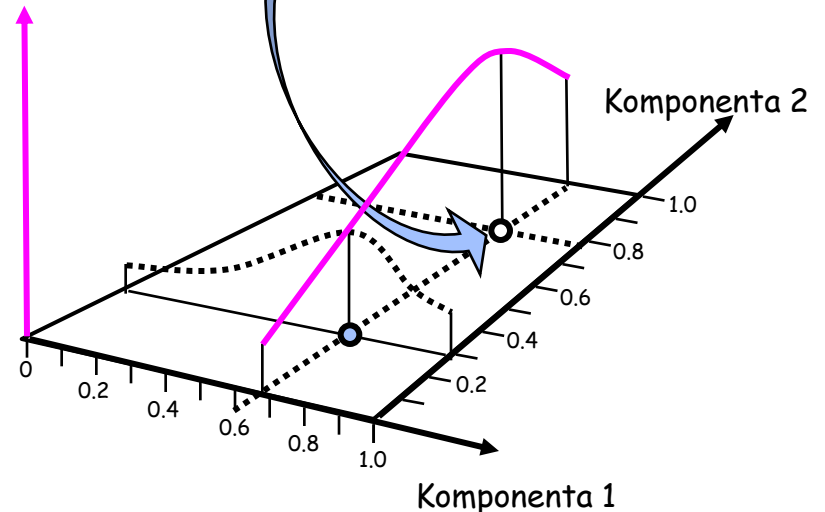
1. faza

Koncentracijo komponente 2 držimo pri vrednosti 0.3 in delamo poskuse z vsemi koncentracijami komponente 1. Najboljši odgovor dobimo pri koncentraciji 0.67 komponente 1.

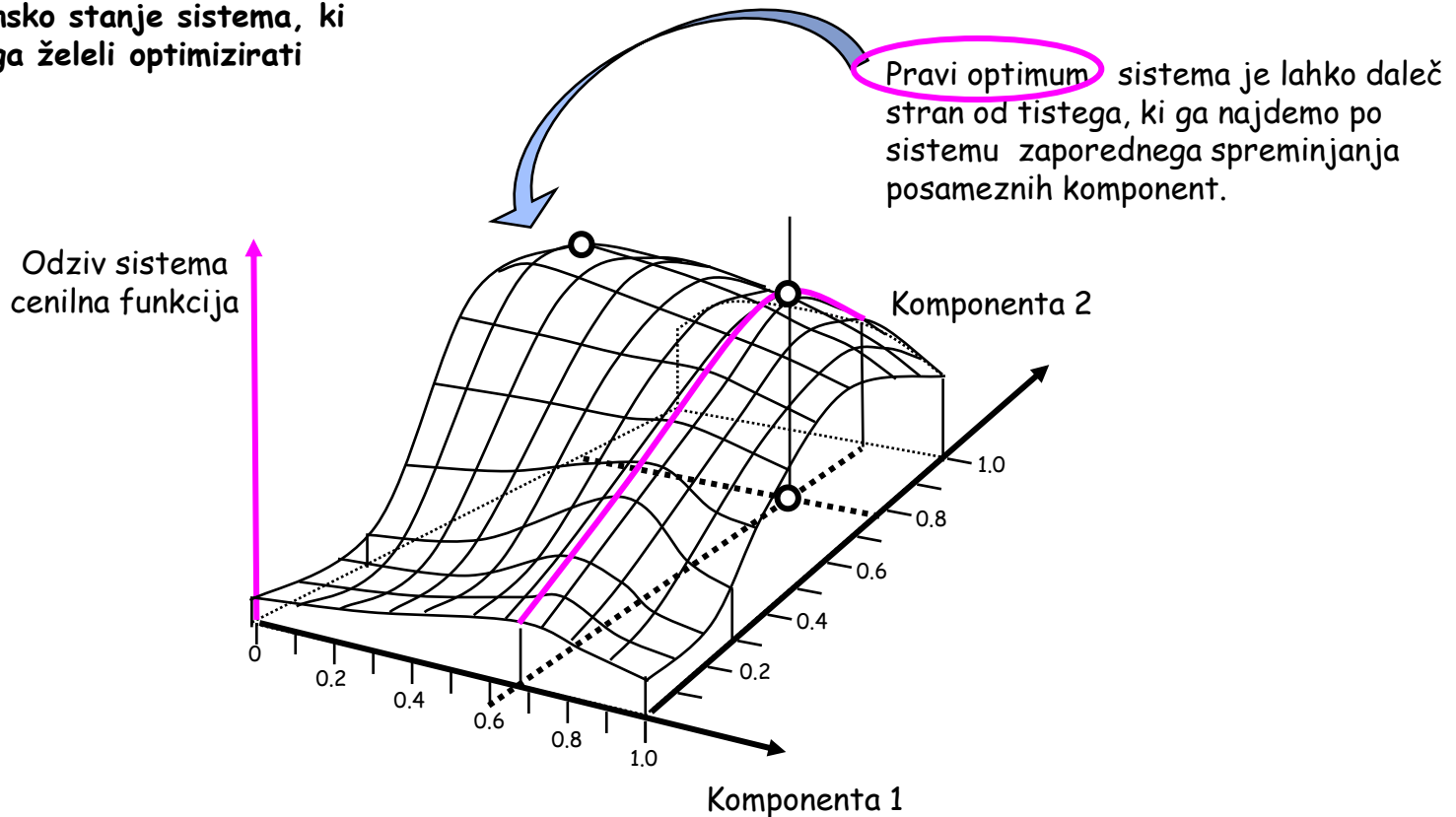
2. faza

Koncentracijo komponente 1 držimo pri vrednosti 0.67 in delamo poskuse z vsemi možnimi koncentracijami komponente 2. Najboljši odgovor dobimo pri koncentraciji 0.75 od maksimalnega možnega deleža komponente 1 v mešanici.

Sklepamo, da je optimum pri koncentracijah 0.67 za komponento 1 in 0.75 za komponento 2.



Dejansko stanje sistema, ki smo ga želeli optimizirati



Že v dvo-komponentnem sistemu nujno zgrešimo pravi optimum, če za eno od obeh komponent v začetku ne izberemo pravilne vrednosti, kar se lahko zgodi zgolj naključno. V primerih, ko imamo več, recimo, m spremenljivk x_i , bi morali pravilno uganiti $m-1$ vrednosti. Šele v tem primeru bi s spreminjanjem zadnje spremenljivke zanesljivo prišli do optimuma. S spreminjanjem ene in držanjem drugih $m-1$ spremenljivk nespremenjenih lahko obvladujemo le zelo ozko lokalno področje v sorazmeroma majhnih tolerančnih intervalih vhodnih spremenljivk x_i . Zato so majhne izboljšave s tako optimizacijo načeloma sicer možne, a le pri dobro poznanih in v že "skoraj optimiziranih" sistemih (nearly optimised systems). Če želimo poiskati optimum v novem in neznanem sistemu, ki je opisan z m -spremenljivkami, potrebujemo hkratno optimizacijo vseh spremenljivk.

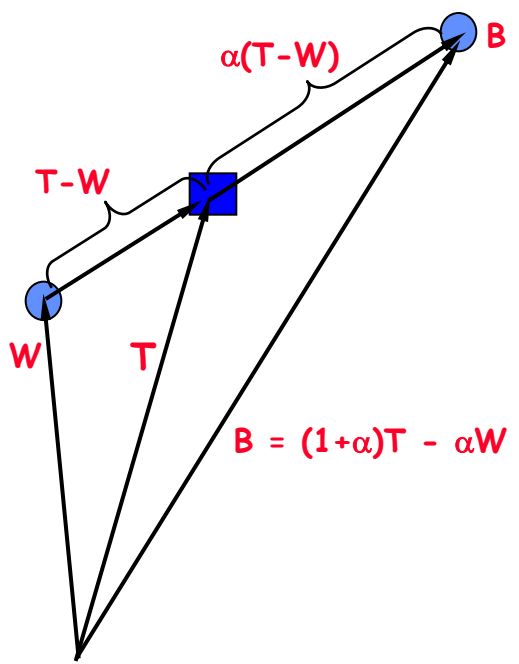
Optimizacija z metodo SIMPLEX

Simplex je najenostavnejša in zelo splošna optimizacijska metoda. Največkrat se uporablja za napovedovanje eksperimentov, ki po najhitrejši poti vodijo k optimumu. Metoda je uporabna tudi (in predvsem takrat), ko nimamo modelne funkcije. Navadno jo uporabljamo pri sorazmeroma majhnem številu spremenljivk: nekako med 2 do 20.

Glavna ideja optimizacije simplex je sledeča: Pričnemo s peščico naključno izbranih (in izvedenih) eksperimentov. S cenilno funkcijo izberemo najslabši eksperiment. Najslabši eksperiment zrcalim preko težišča vseh ostalih eksperimentov. S tem dobimo novo točko v m -dimenzionalnem prostoru. Ker ležita nova točka in najslabša točka vsaka na svoji strani težišča sistema, bi morala biti nova točka boljša. Izračun simetrične točke pomeni izračunati vse spremenljivke x_i za novi eksperiment, ki naj bi ležal v nasprotnem delu prostora kot najslabša točka.

Optimizacijo simplex sestavljajo naslednji koraki:

1. Za sistem, ki je opisan z m spremenljivkami x_i , ($i=1, \dots, m$) moramo z generatorjem naključnih števil določiti $m+1$ eksperimentov, jih izvesti in s cenilno funkcijo oceniti (kvantitativno opredeliti kvaliteto).
2. Glede na rezultat cenilne funkcije ali optimizacijskega kriterija (OK) razvrstimo vse eksperimente in določimo najslabšega.
3. Izračunamo težišče preostalih točk (brez najslabšega eksperimenta) in najslabši eksperiment preslikamo (zrcalimo) preko težišča. S tem dobimo nov eksperiment.
4. Novi eksperiment izvedemo in ga ocenimo s cenilno funkcijo.
5. Pri oceni imamo tri možnosti:
 - a) Novi eksperiment je dal najboljšo vrednost od vseh prejšnjih eksperimentov. Naslednji novi eksperiment izračunamo v podaljšku iste smeri, ki smo jo že izbrali ob prejšnji preslikavi. Postopek nadaljujemo pri koraku št. 4.
 - b) Novi eksperiment je nekje med drugim najslabšim in najboljšim rezultatom. Nadaljujemo pri koraku št. 2.
 - c) Novi eksperiment je najslabši od vseh. Kot novega "najslabšega" vzamemo drugega najslabšega in nadaljujemo pri koraku št. 3. V tem primeru moramo paziti na sledeče: Lahko se zgodi, da preslikave vseh točk (eksperimentov) simplexa dajo slabše rezultate od vseh preostalih eksperimentov. To pomeni, da smo zelo blizu optimuma in takrat moramo simplex zmanjšati. Hkrati ocenimo, glede na izračun napake meritev in cenilne funkcije, če je nadaljevanje optimizacije smiselno ali ne. Če se vse vrednosti spremenljivk novegega izračunanega eksperimenta nahajajo znotraj intervala zaupanja prejšnjih eksperimentov, potem je nadaljevanje nesmiselno. Isto velja za oceno cenilne funkcije, ki ima lahko tudi precejšnji standardni odklik. Zaradi tega je izredno pomembno, da so meritve kvalitete izdelkov ali sistemov, ki jih želimo optimizirati, izmerjene kar se da natančno.



B objekt z najboljšim (the best) odgovorom
T Težišče preostalih točk (brez najslabše)
W objekt z najslabšim (the worst) odgovorom

$$B = T + \alpha(T - W) = T + \alpha T - \alpha W = (1 + \alpha)T - \alpha W$$

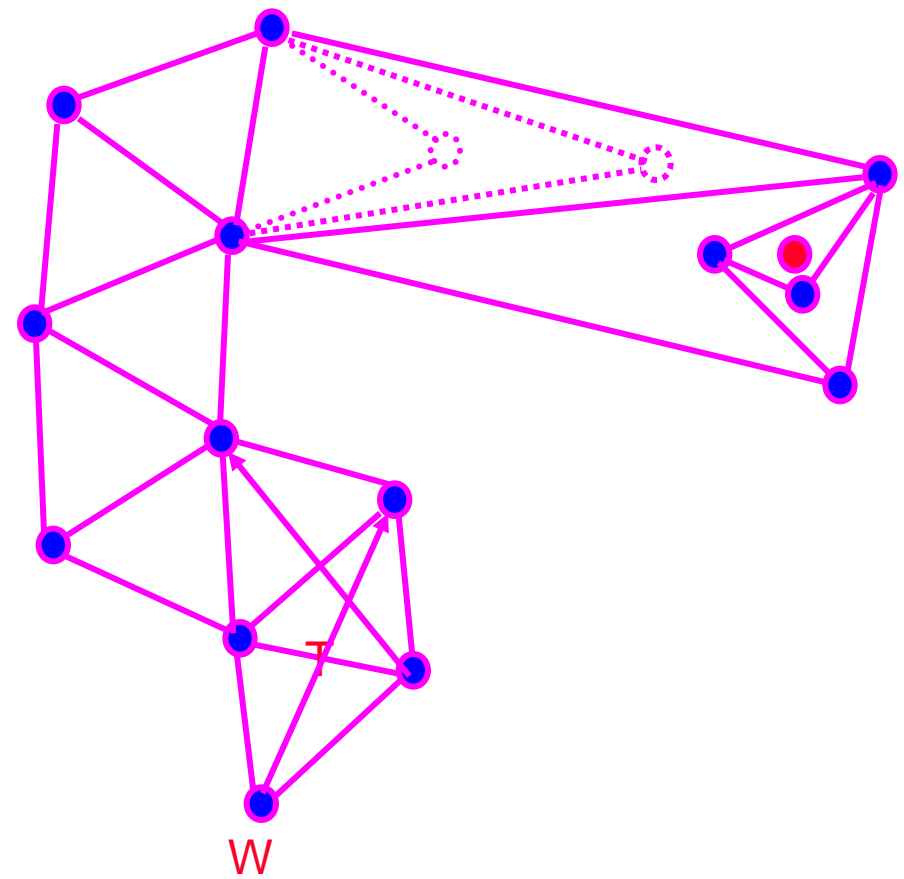
V primeru čistega zracaljenja (običajni koraki simplexa) je $\alpha = 1$ in naslednji eksperiment izračunamo po enčbi:

$$B = 2T - W$$

Težišče T je povprečna vrednost komponent vseh eksperimentov, brez najslabšega:

$$T = \frac{1}{m} \sum_{i=1}^{m+1} x_i \quad (\text{brez najslabšega})$$

Primer optimizacije dvokomponentnega sistema. Simplex ima 3 točke



Ena od stvari, na katere moramo paziti pri izračunu novih eksperimentov, je, da nove vrednosti ne padejo iz območij, kjer so posamezne spremenljivke sploh možne ali dovoljene. Če se to pri izračunu zgodi, je treba vrednost take spremenljivke postaviti na tisto mejno vrednost, ki je bila presežena.

Realni primer optimizacije gradbeniškega kita za tesnjenje vzdanih elementov

Optimizacija recepture gradbeniškega kita.

Gre za optimizacijo kita, ki se uporablja za tesnenje na stikih med posameznimi elementi kot so vzdane betonske plošče ali za stike betonska plošča/okenski okvir in podobno. Receptura za izdelavo kita vsebuje 9 spremenljivk. Optimizacija je izvedena glede na vsebnost sestavin (ne na postopek priprave). Ker imajo sestavine zelo različno ceno, gre pri optimizaciji v prvi vrsti za iskanje izdelka, ki mora dosegati s standardi predpisano kvaliteto (ali pa imeti še boljše) in hkrati vsebovati čim manj najdražje komponente.

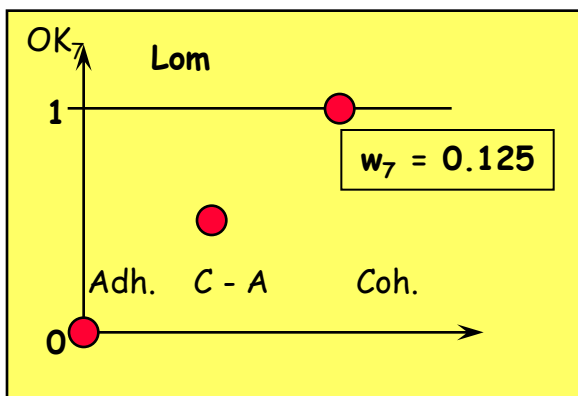
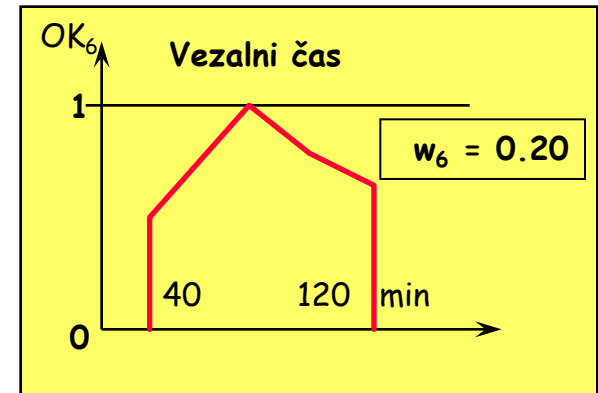
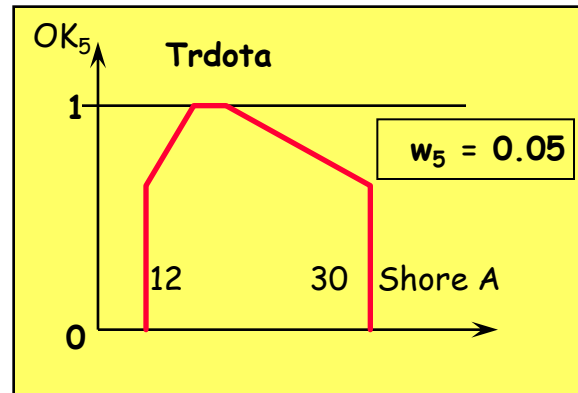
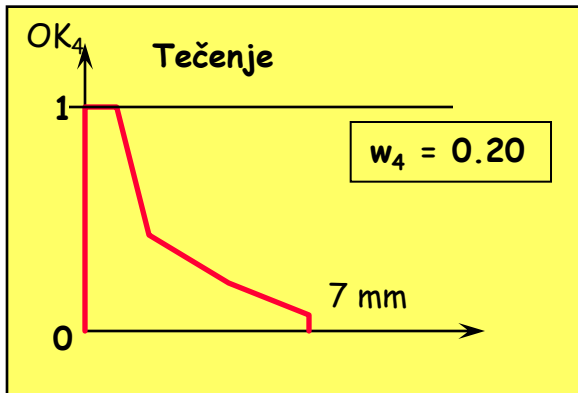
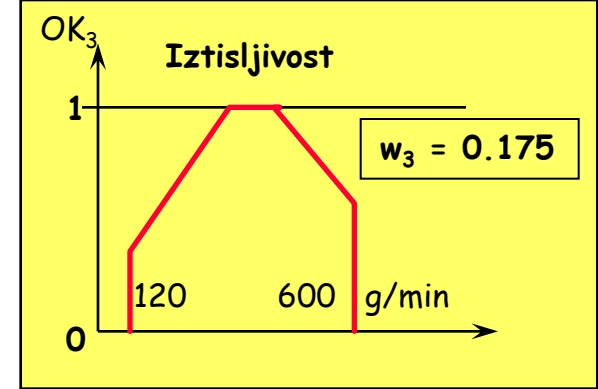
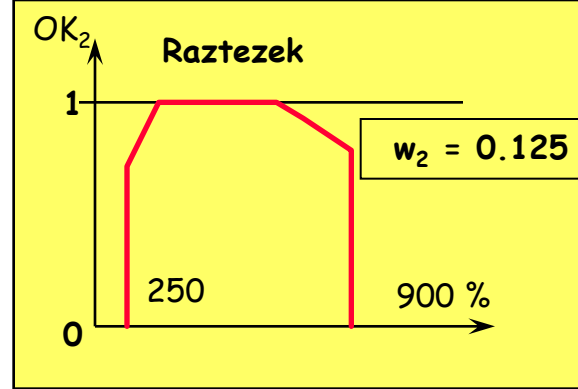
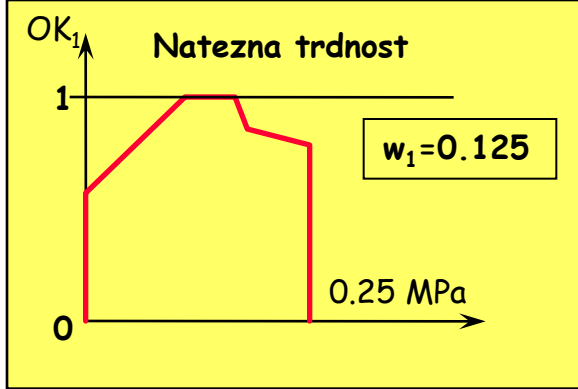
9 spremenljivk:

• polimer	160 - 360 g
• plastifikator	80 - 250 g
• adheziv	8 - 80 g
• pigment	40 - 120 g
• zgoščevalno sred.	1.6 - 32 g
• reg. vulkan.	0.24 - 0.48 g
• polnilo 1	80 - 100 g
• polnilo 2	80 - 200 g
• trdilec	0.4 - 2 g

7 lastnosti:

• strength (natezna trdnost)	[Mpa]
• expansion (raztezek)	[%]
• extrusion (iztisljivost)	[g/min]
• slide (tečenje)	[mm pri 30 °C]
• hardness (trdota)	[Shore A]
• breakability (lom)	[3 razredi]
• binding time (vezanje)	[min]

Lastnosti, ki določajo optimum izdelka, je sedem in vse so po svojem učinkovanju in s tem definiranju izdelka precej različne. Izrazito takšni sta iztisljivost iz tub oziroma brizgalk in pa tečenje kita, ko je le-ta že iztisnjen ali nanešen na oba materiala, med katerima mora kit zatesniti razmike. Če je iztisljivost dobra (velika), je običajno tudi tečenje po nanešenem materialu veliko, kar je seveda slabo za delavni postopek, in obratno: če je tečenje po materialu (včasih pravimo tudi mazavost) dobro, potem je iztisljivost slaba (za iztisnjenje je potrebna velika sila).



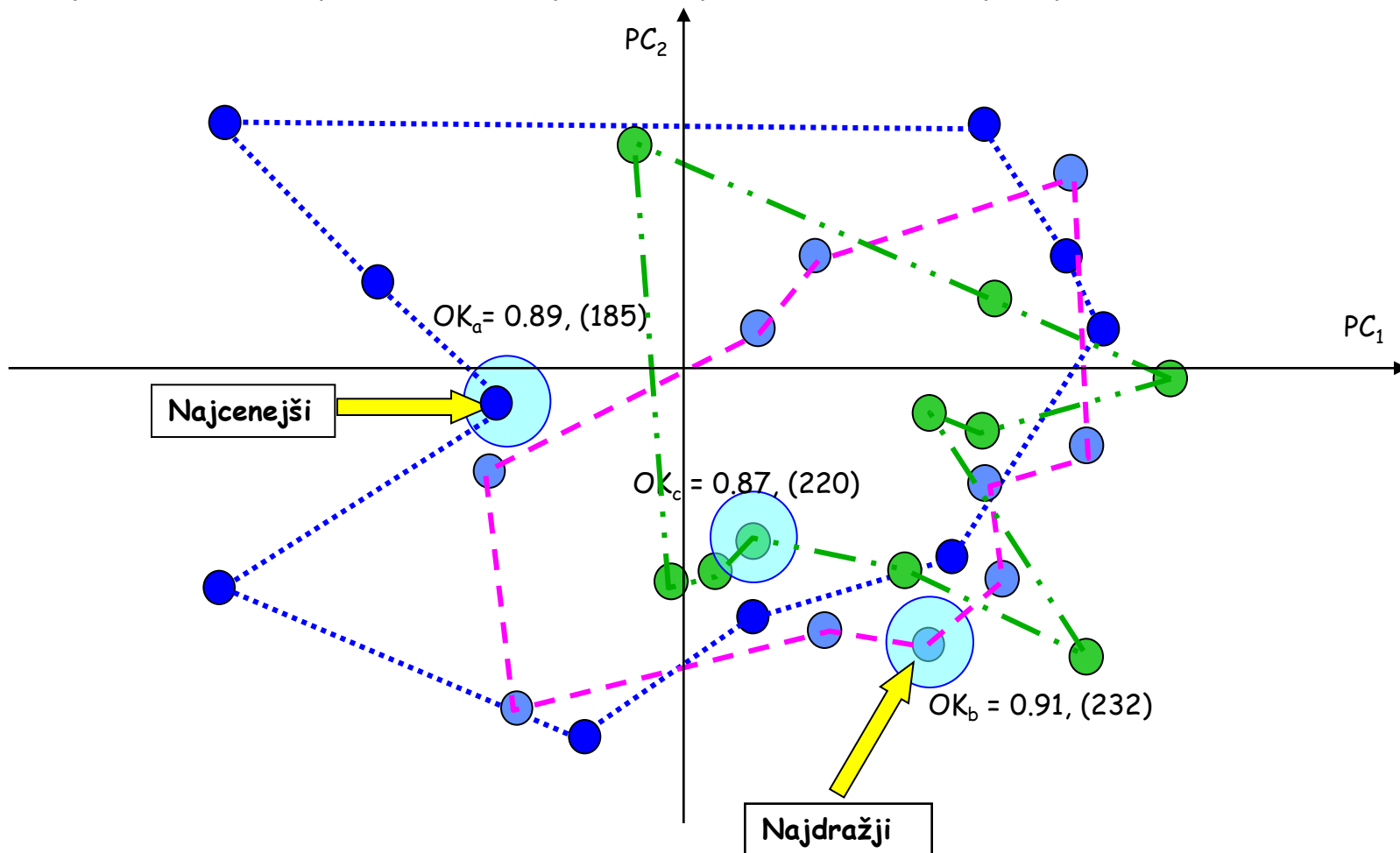
Vsi posamezni optimizacijski kriteriji OK_i so normalizirani. Vrednost $OK_i = 1$ pomeni, da je lastnost optimalna. $OK_i = 0$, pomeni, da je izdelek nesprejemljiv. Uteži w_i , za vsako lastnosti j povedo, kolikšen vpliv ima posamezna lastnost v skupnem optimizacijskem kriteriju $OK_{celoten}$. Določene so bile tako, da so najprej eksperti ocenili skupno kvaliteto $OK_{celoten}$ desetih izdelkov z ocenami od 1 do 10, potem so bile uteži w_i izračunane z MLR postopkom: 10 enačb za 7 uteži (glej poglavje o MLR). OK_{ij} je i -ta eksperimentalna vrednost OK_j .

$$OK_{1, celoten} = w_1 OK_{1,1} + w_2 OK_{1,2} + \dots + w_7 OK_{1,7} = \sum_j w_j OK_{ij},$$

... ..

$$OK_{10, celoten} = w_1 OK_{10,1} + w_2 OK_{10,2} + \dots + w_7 OK_{10,7}$$

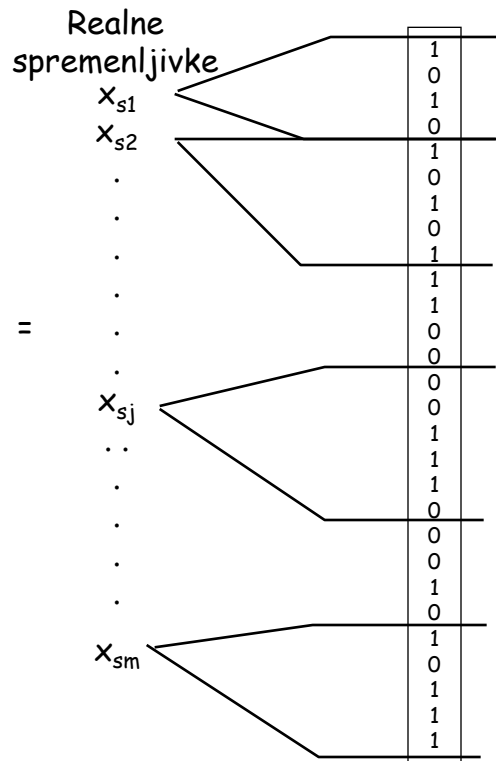
Ker je vsak izdelek opisan z devetimi spremenljivkami, je treba za izhodiščni simplex narediti 10 eksperimentov (10 izdelkov). Celoten postopek je bil voden iz treh povsem različnih izhodiščnih simplexov, to je skupaj iz 30 izdelkov in ustreznih 210 (7×30) meritev lastnosti. Na spodnji sliki so prikazani položaji končnih simplexov v 2d-prostoru obeh glavnih osi PC_1/PC_2 . Kvalitetni kriteriji $OK_{celoten}$ najboljših točk vseh treh simplexov so 0.89, 0.87 in 0.91. V oklepajih so poleg optimizacijskih kriterijev navedene tudi mase najdražje komponente (plastifikatorja) v gramih na 1000 g končnega izdelka. Pomemben rezultat je v tem, da ima drugi najboljši izdelek ($OK_a = 0.89$) *najmanj* najdražje komponente (185 g), kar predstavlja 25 % prihranek v primerjavi z najboljšim izdelkom ($OK_b = 0.91$). V primerjavi s standardno recepturo, ki je bila največkrat uporabljena v dejanski proizvodnji (a je imela nižji kvalitetni kriterij, kot navedeni trije izdelki), je bil prihranek pri najdražji komponenti približno 15 %.



Optimizacija z genetskim algoritmom je precejšnje nasprotje optimizaciji z metodo SIMPLEX. Prva je večinoma namenjena optimizaciji eksperimentalnega dela, saj najbolje deluje takrat, ko imamo sorazmeroma malo spremenljivk (manj kot 20) in ko imajo le-te realne vrednosti. Optimizacija z genetskim algoritmom (GA) je namenjena reševanju problemov pri katerih so objekti optimizacije (spektri, strukture, procesi, postopki) predstavljeni z velikim številom spremenljivk. GA je primeren tudi za primere, ko želimo iz velikega števila možnih spremenljivk izbrati tiste, ki so za dani primer najbolj relevantne.

Pri optimizaciji z genetskim algoritmom se je uveljavilo posebno izrazoslovje, ki je v tesni povezavi s pravo genetiko. Objekti optimizacije v kemiji, kot os spektri, večkomponentne analize, strukture, procesi in podobno [m -dimenzionalni vektorji $X_s = (x_{s1}, x_{s2}, \dots, x_{sm})$], so v genetskem algoritmu predstavljeni kot "kromosomi". Največkrat kot binarni vektorji: $X_s = (0000010010101010111101010101000111101010101)$

Kromosom z bitnimi predstavitvami realnih spremenljivk



Bitom, ki v kromosomu predstavljajo intervale realnih vrednosti spremenljivk, pravimo "geni". Za vsako spremenljivko x_j je treba definirati število bitov, s katerimi bo v postopku GA predstavljena. V statističnem jeziku pomeni to izbor števila nivojev vsake spremenljivke. Število nivojev pri GA je lahko samo potenca 2: $2^1 = 2$, $2^2 = 4$, $2^3 = 8$, $2^4 = 16$, $2^5 = 32$, $2^6 = 64$, $2^7 = 128$, $2^8 = 256$. Predstavitev vrednosti spremenljivke z več kot osmimi biti (več kot 256 nivojev) pride v poštev zelo redko, saj so spremenljivke večinoma merjene z natančnostjo okrog 1 %, za to pa zdostuje že 7 bitov (128 vrednosti). Primer: Spremenljivko pH v območju 5.4 - 7.7 lahko povsem zadovoljivo predstavimo s 16 nivoji (4 biti). Vrednosti pH so razmaknjene za: $0.153 = (7.7-5.4)/(16-1)$ pH. Posamezni nivoji pH (skupina štirih genov) so potem naslednji:

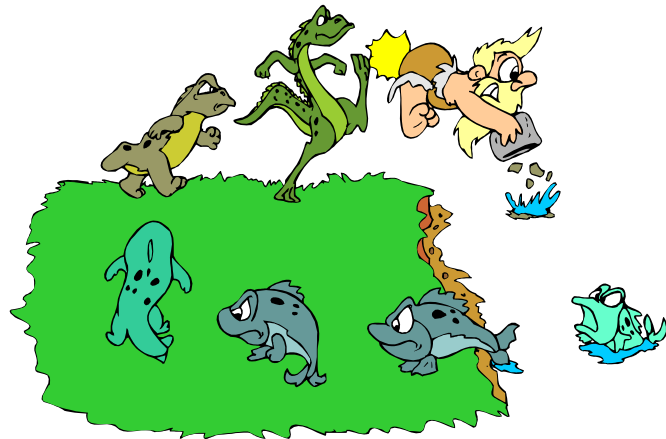
- 0000 = 5.40, 0001 = 5.55, 0010 = 5.71, 0011 = 5.86,
- 0100 = 6.01, 0101 = 6.17, 0110 = 6.32, 0111 = 6.48,
- 1000 = 6.63, 1001 = 6.78, 1010 = 6.94, 1011 = 7.09,
- 1100 = 7.24, 1101 = 7.40, 1110 = 7.55, 1111 = 7.70

Če želimo z optimizacijo ugotoviti, ali je boljša prisotnost ali odsotnost neke komponente, potem uporabimo samo en bit (0/1).

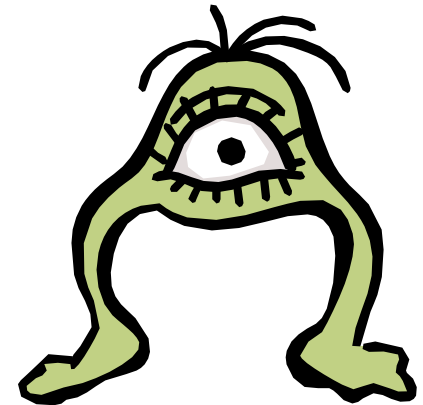
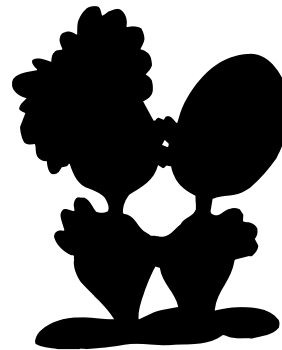
Poleg izrazov "osebek" ali kromosom vpeljuje genetski algoritem tudi pojme kot so "starši" (parents), "potomci" (offspring) in "generacija" ali "bazen" (pool). V dveh zaporednih generacijah je vedno najprej generacija staršev, ki ji sledi generacija potomcev. Potomci staršev prve generacije so starši potomcev tretje generaciji itd. Vsaka generacija vsebuje isto število osebkov (staršev ali potomcev). Z drugimi besedami: "pool" "kromosomov" je ves čas enakoštevilen. Ta lastnost ponazarja omejenost surovin v naravi, oziroma dejstvo, da se v stabilnih razmerah populacija zaradi omejitve "hrane" ne večja a tudi ne manjša, ker je "hrana" za določeno številčno stanje populacije zadostna. Optimizacija z genetskim algoritmom posnema postopek naravno selekcijo vedno uspešnejših potomcev. V naravni selekciji so trije mehanizmi, ki v vsaki generaciji vodijo do izboljšave:

1. Preživijo le najbolj prilagojeni osebki (osebki z najboljšimi kromosomi). Zato imajo več potomcev kot slabši,
2. S križanjem kromosomov pride do sprememb posameznih genov ali skupin genov in s tem do osebkov z boljšimi/slabšimi lastnostmi,
3. So sprememb genov in s tem do izboljšav/poslabšanja lastnosti pride tudi z dednimi mutacijami (naključnimi spremembami genov).

1. Preživetje najboljših

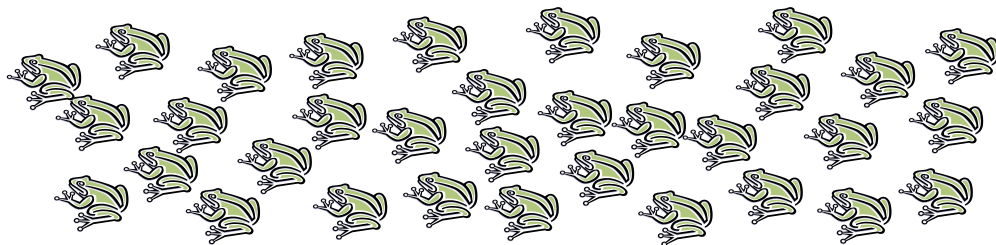


2. Izboljšave s križanjem



3. Izboljšave z mutacijami

1	0	0	0	1	0	0	0	1	1	1	0	1	1	1	0	1	0	1	0	0	1	0	1	1	0	1	0	1	0	0
0	0	1	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0
1	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1	1	1	0	1	1	1	0	1	0	0	1	1	1	0	1
0	1	0	0	0	0	0	0	1	0	1	1	0	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0
0	0	1	1	0	1	0	0	0	0	1	0	0	0	1	1	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	1	1	1	0	1	1	0	0	1	1	1	0	0	0
1	0	0	0	1	0	0	1	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	1	0	1	0	1	1	1	1	0	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0



Postopek optimizacije z genetskim algoritmom se začne s tem, da generiramo skupino N kromosomov z naključno postavljenimi geni. Vsak kromosom je binarna predstavitev vektorja $\mathbf{X}_s = (x_{1s}, x_{2s}, \dots, x_{sm})$ oziroma posameznih sprmenljivk.

V primeru na levi strani imamo 31 kromosomov (objektov), od katerih je vsak predstavljen z devetimi geni.

2.faza

najslabši



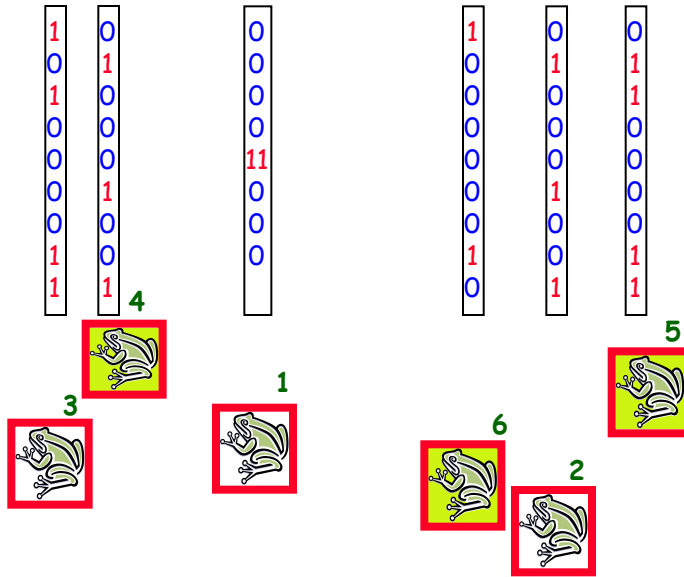
najboljši



V drugi fazi postopka je vsak kromosom (objekt \mathbf{X}_s) ovrednoten s cenilno funkcijo ali optimizacijskim kriterijem, ki ga izračunamo se na podlagi meritev ali na podlagi modela $OK = f(\mathbf{X}_s)$. Kot modele pri optimizacijskih kriterijih velikokrat uporabljamo umetne nevronske mreže.

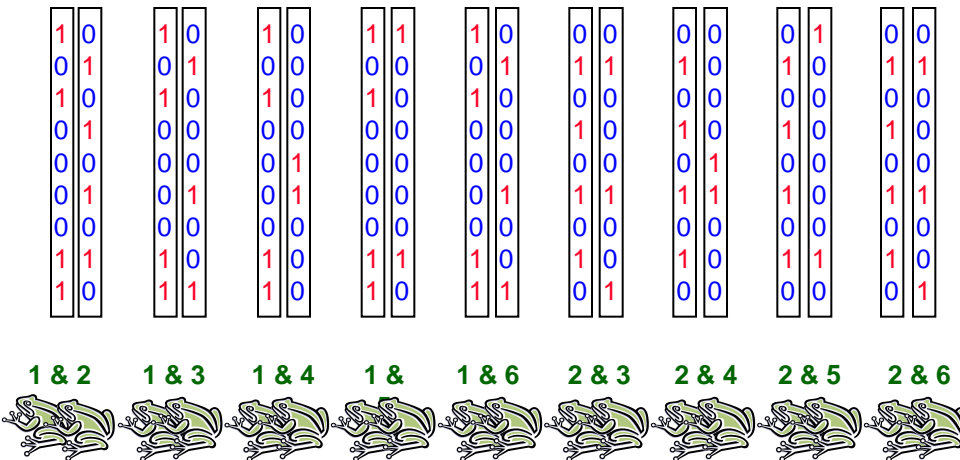
Vse kromosome razvrstimo po velikosti optimizacijskega kriterija OK in kot starše ohranimo le določen odstotek najboljših (10 - 50 %). Na levi strani so najboljši osebki označeni s krogom.

3. faza



1

4. faza

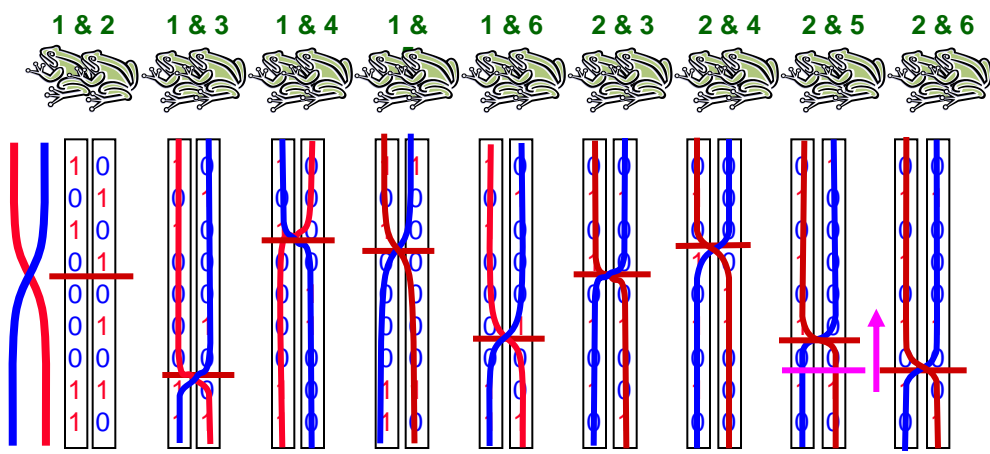


Izbrani kromosomi so dali najboljše rezultate pri vrednotenju s cenilno funkcijo. Število oziroma odstotek najbolj ocenjenih kromosomov stare populacije je eden glavnih parametrov optimizacijskega postopka. Odstotek kromosomov, ki naj v vsaki generaciji preživi in poskrbi za potomstvo, mora biti izbran, preden se optimizacijski postopek prične. Z določitvijo tega odstotka reguliramo različnost genske porazdelitve znotraj izbranega dela najboljših kromosomov. Če izberemo premajhen odstotek "preživelih", se lahko zgodi, da optimizacija prehitro konča v lokalnem optimumu. S prevelikim odstotkom preživelih, pa je lahko optimizacija prepočasna. Zaradi tega tudi pri optimizaciji z genetskim algoritmom ves postopek ponovimo večkrat z različnimi začetnimi parametri.

Generacijo potomcev pričnemo z izbiro *parov* staršev izmed kromosomov, ki smo jih obdržali. Za vsak par potomcev (članov naslednje generacije) je potreben par staršev. Pare lahko izbiramo naključno. Željeno je, da kromosom nastopa kot "starš" tem večkrat, čim višja je njegova vrednost cenilne funkcije. V primeru na levi strani je izbor staršev narejen tako, da so tvorjeni najprej vsi možni pari z najboljšim kromosomom, sledijo pari z drugim najboljšim itd. Ker ima vsak par staršev dva potomca, moramo postopek ponavljati (lahko tudi ponovno, od začetka), dokler ne dobimo $N/2$ parov.

V danem primeru, ko mora biti v vsaki generaciji 31 kromosomov, naredimo 15 parov staršev, zadnji kromosom pa dobimo tako, da prenesemo najboljši kromosom stare generacije v novo generacijo, pravimo "elitizem". Elitizem zagotovi, da bo najboljši kromosom naslednje generacije vsaj tako dober, kot najboljši iz prejšnje.

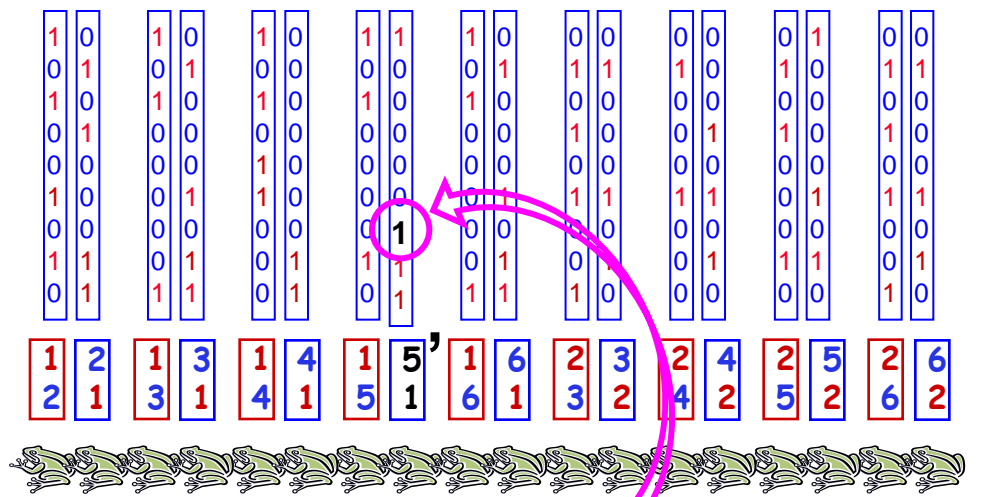
5. faza



Ko so pari staršev izbrani, se v izbranih parih prične križanje kromosomov. Mesto križanja v vsakem paru kromosomov je izbrano naključno. Križanje je v tem, da se vsebini kromosomov (genska zapisa lastnosti) pod mestom križanja v obeh kromosomih zamenjata. S tem dobimo dva povsem nova kromosoma, ki s svojima novima genskima zapisoma povsem natanko definirata dva nova potomca (objekta).

Pri križanju je treba paziti, da kromosoma obeh staršev pod mestom križanja nista enaka (glej predzadnji par kromosomov na levi). Če pod mestom križanja v obeh kromosomih ni razlik, potem pomikamo mejo križanja navzgor (puščica) toliko časa, da ne pridemo do prvih dveh različnih genov. Križanje lahko izvajamo tudi z dvema križnima mesti.

6. faza



Zadnja faza nastajanja nove generacije kromosomov je mutacija. Mutacija je naključno spreminjanje vrednosti posameznih genov - iz nič v ena ali obratno. Verjetnost mutacije je začetni parameter optimizacije GA in jo moramo prilagoditi dolžini kromosoma, se pravi številu genov v enem kromosomov. Verjetnost za spremembo genov mora biti zelo majhna, običajno manj kot 0.5 %. Če bi, npr.: prišlo do mutacije v vsakem kromosomu, se vpliva križanj skoraj ne bi zaznalo. Zato je treba izbiro tega parametra skrbno prizkusiti.

mutacija

Ko je generiranih N novih potomcev, se prične postopek ocenjevanja nove generacije. Postopek je končan, ko v novih generacijah ne pride več do izboljšav.