

# Pitanja za pripremu prvog kolokvija iz kolegija Evolucijska Robotika

## 1. Umjetne neuronske mreže ?

- Neuronske se mreže dijele na biološke (prirodne), te umjetne neuronske mreže. Dok biološke predstavljaju biološki organizmi, mozak ljudi i životinja te visoka složenost i paralelizam, umjetne neuronske mreže motivirane su biološkim neuronskim mrežama. Za sada su dosta primitivne imitacije bioloških mreža.
- Implementiraju se na digitalnim računalima opće namjene ili pomoću specijaliziranih sklopova (analognih, digitalnih, hibridnih).
- Umjetna neuronska mreža je masivno paralelni distribuirani procesor koji je dobar za pamćenje iskustvenog znanja. Slična je mozgu u dva aspekta: znanje se stiče kroz proces učenja, te se međusobne veze između neurona koriste za spremanje znanja.

## 2. Biološki neuron?

- Ljudski mozak sastavljen je od oko  $10^{11}$  neurona kojih ima više od 100 vrsta i koji su shodno svojoj funkciji raspoređeni prema točno definiranom rasporedu. Svaki je neuron u prosjeku povezan s  $10^4$  drugih neurona. Četiri su osnovna dijela neurona:
  - tijelo stanice (soma),
  - skup dendrita (ogranaka),
  - aksona (dugačke cjevčice koje prenose električke poruke) i
  - niza završnih članaka
- Tijelo stanice sadrži informaciju predstavljenu električkim potencijalom između unutrašnjeg i vanjskog dijela stanice (oko  $-70$  mV u neutralnom stanju).
- Na sinapsama, spojnom sredstvu dvaju neurona kojim su pokriveni dendriti, primaju se informacije od drugih neurona u vidu post-sinaptičkog potencijala koji utječe na potencijal stanice povećavajući (hiperpolarizacija) ili smanjujući ga (depolarizacija).
- U tijelu stanice sumiraju se post-sinaptički potencijali tisuća susjednih neurona, u ovisnosti o vremenu dolaska ulaznih informacija.
- Ako ukupni napon pređe određeni prag, neuron "pali" i generira tzv. akcijski potencijal u trajanju od 1 ms. Kada se informacija akcijskim potencijalom prenese do završnih članaka, onda oni, ovisno o veličini potencijala, proizvode i otpuštaju kemikalije, tzv. neurotransmitere.
- To zatim ponovno inicira niz opisanih događaja u daljnjim neuronima. Propagacija impulsa očigledno je jednosmjerna
- Funkcionalnost biološkog neurona imitira **McCulloch-Pitts model** umjetnog neurona, tzv. **Threshold Logic Unit (TLU)**
- **Model** koristi slijedeću analogiju:
  - signali su opisani numeričkim iznosom i na ulazu u neuron množe se težinskim faktorom koji opisuje jakost sinapse;
  - signali pomnoženi težinskim faktorima zatim se sumiraju analogno sumiranju potencijala u tijelu stanice;
  - ako je dobiveni iznos iznad definirana praga, neuron daje izlazni signal.

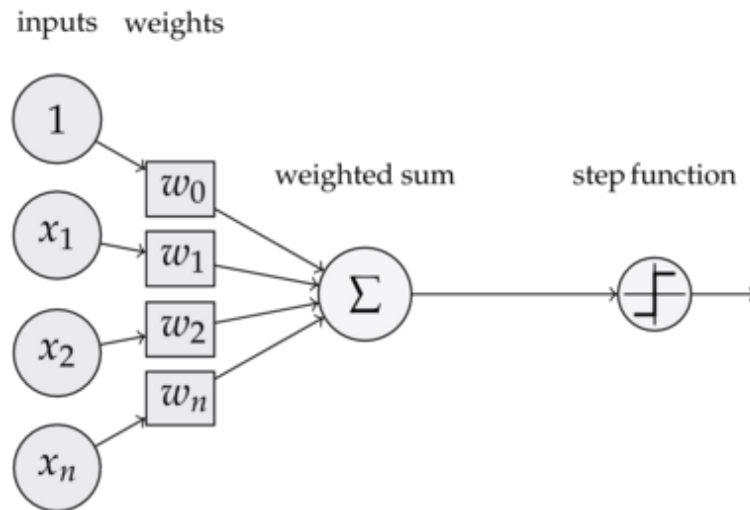
### 3. Definicije i osobitosti umjetne neuronske mreže ?

- Umjetna neuronska mreža u širem je smislu riječi umjetna replika ljudskog mozga kojom se nastoji simulirati postupak učenja.
- To je paradigma kojom su implementirani pojednostavljeni modeli što sačinjavaju biološku neuronsku mrežu.
- Analogija s pravim biološkim uzorom zapravo je dosta klimava jer uz mnoga učinjena pojednostavljena postoje još mnogi fenomeni živčanog sustava koji nisu modelirani umjetnim neuronskim mrežama kao što postoje i karakteristike umjetnih neuronskih mreža koje se ne slažu s onima bioloških sustava.
- **Neuronska mreža jest skup međusobno povezanih jednostavnih procesnih elemenata, jedinica ili čvorova, čija se funkcionalnost temelji na biološkom neuronu**
- Pri tome je **obradbena moć mreže pohranjena u snazi veza između pojedinih neurona tj. težinama** do kojih se dolazi postupkom prilagodbe odnosno učenjem iz skupa podataka za učenje.
- Neuronska mreža obrađuje podatke distribuiranim paralelnim radom svojih čvorova. Neke **osobitosti neuronskih mreža** naspram konvencionalnih (simboličkih) načina obrade podataka su slijedeće:
  - Vrlo su dobre u procjeni nelinearnih odnosa uzoraka.
  - Mogu raditi s nejasnim ili manjkavim podacima tipičnim za podatke iz različitih senzora, poput kamera i mikrofona, i u njima raspoznavati uzorke.
  - Robusne su na pogreške u podacima, za razliku od konvencionalnih metoda koje pretpostavljaju normalnu raspodjelu obilježja u ulaznim podacima.
  - Stvaraju vlastite odnose između podataka koji nisu zadani na eksplicitan simbolički način.
  - Mogu raditi s velikim brojem varijabli ili parametara.
  - Prilagodljive su okolini.
  - Moguća je jednostavna VLSI implementacija.
  - Sposobne su formirati znanje učeći iz iskustva (tj. primjera)
  - Neuronske mreže odlično rješavaju probleme klasifikacije i predviđanja, odnosno općenito sve probleme kod kojih postoji odnos između prediktorskih (ulaznih) i zavisnih (izlaznih) varijabli, bez obzira na visoku složenost te veze (nelinearnost).
- Danas se **neuronske mreže primjenjuju** u mnogim segmentima života poput medicine, bankarstva, strojarstva, geologije, fizike itd., najčešće za slijedeće zadatke:
  - raspoznavanje uzoraka,
  - obrada slike,
  - obrada govora,
  - problemi optimizacije,
  - nelinearno upravljanje,
  - obrada nepreciznih i nekompletnih podataka,
  - simulacije i sl.

#### 4. Postupci učenja mreže ?

- Jednostavnije neuronske mreže moguće je konstruirati tako da obavljaju određeni zadatak. Ovo će redom biti moguće **za mreže koje se sastoje od TLU perceptrona**, i koje obavljaju unaprijed zadanu logičku funkciju, jer u tom slučaju možemo pratiti što i kako točno mreža radi.
- **U slučaju** kada se koriste **složenije prijenosne funkcije**, poput sigmoidalne, ili dopušta rad s realnim brojevima, tipično se **gubi nadzor nad načinom kako mreža obrađuje podatke**.
- U tom slučaju uobičajeno se da se definira arhitektura mreže, i prije postupka obrade podatka obavi **postupak učenja ili treniranja**. Za razliku od konvencionalnih tehnika obrade podataka gdje je postupak obrade potrebno analitički razložiti na određeni broj algoritamskih koraka, kod ovog tipa neuronskih mreža takav algoritam ne postoji.
- Znanje o obradi podataka, tj. znanje o izlazu kao funkciji ulaza, pohranjeno je implicitno u težinama veza između neurona. Te se težine postupno prilagođavaju kroz postupak učenja neuronske mreže sve do trenutka kada je izlaz iz mreže, provjeren na skupu podataka za testiranje, zadovoljavajući.
- Pod **postupkom učenja** kod neuronskih mreža podrazumijevamo **iterativan postupak predočavanja ulaznih primjera** (uzoraka, iskustva) i eventualno **očekivana izlaza**.
- Ovisno o tome da li nam je u postupku učenja á priori znan izlaz iz mreže, pa ga pri učenju mreže koristimo uz svaki ulazni primjer, ili nam je točan izlaz nepoznat, razlikujemo **dva načina učenja**:
  - **učenje s učiteljem** – učenje mreže provodi se primjerima u obliku para (ulaz, izlaz),
  - **učenje bez učitelja** – mreža uči bez poznavanja izlaza.
- **Skup primjera za učenje često se dijeli na tri odvojena skupa**:
  - **skup za učenje, skup za testiranje i skup za provjeru** (validaciju).
  - Primjeri iz **prvog skupa** služe za **učenje u užem smislu** (podešavanje težinskih faktora).
  - Pomoću primjera iz **drugog skupa** vrši se tijekom učenja **provjera rada mreže s trenutnim težinskim faktorima** kako bi se postupak učenja zaustavio u trenutku degradacije performanse mreže.
  - **Umjetnu neuronsku mrežu moguće je, naime, pretrenirati** - nakon određenog broja iteracija mreža gubi svojstvo generalizacije i postaje stručnjak za obradu podatka iz skupa primjera za učenje dok preostale podatke obrađuje loše.
  - Stalnim praćenjem izlaza iz mreže dobivenog pomoću primjera iz skupa za testiranje moguće je otkriti iteraciju u kojoj dobiveni izlaz najmanje odstupa od željenog. **Točnost i preciznost obrade podataka** moguće je na posljeticu provjeriti nad **trećim skupom** primjera – **skupom za provjeru**.
- Uz pojam učenja umjetne neuronske mreže vezani su pojmovi **iteracije i epohe**.
- Pod **iteracijom** podrazumijevamo korak u algoritmu postupka za učenje u kojem se odvija podešavanje težinskih faktora
- **epoha** jedno predstavljanje cjelokupnog skupa za učenje.
- Ovisno o broju primjera predočenih mreži za trajanje jedne iteracije, razlikujemo:
  - **pojedinačno učenje** – u jednoj iteraciji predočavamo samo jedan primjer za učenje (tj. kod svakog primjera za učenje vrši se prilagodba težinskih faktora),
  - **grupno učenje** – u jednoj iteraciji predočavamo sve primjere za učenje (tj. iteracije se podudaraju s epohama)..

## 5. Što je perceptron (skica perceptrona i objašnjenje) ?



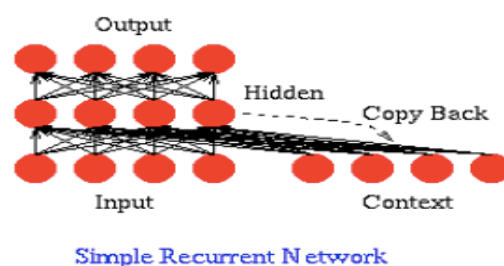
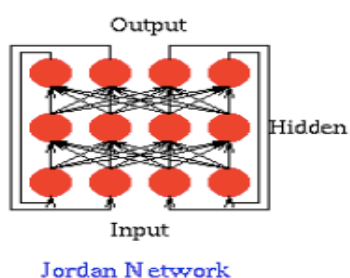
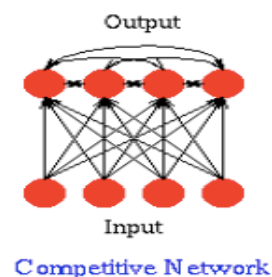
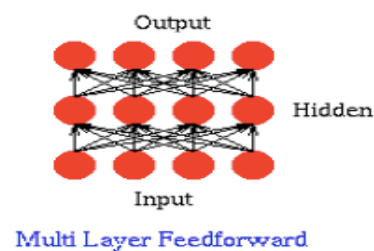
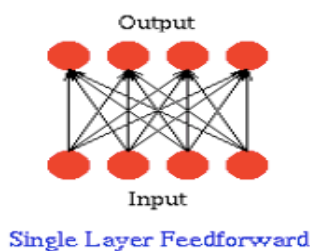
- Perceptron je najjednostavniji model neuronske mreže, 1 neuron
- Ulazni podaci  $x_1, x_2, \dots, x_n$ , s različitim težinskim faktorima  $w_1, w_2, \dots, w_n$  koji se množe i sumiraju prije dolaska na aktivacijsku funkciju.

$$s = \sum_{i=0}^n w_i x_i$$

- Nakon sumiranja svih ulaznih podataka s njihovim težinskim faktorima, suma  $s$  prolazi kroz step funkciju  $f$  ( npr. Heviside step function)

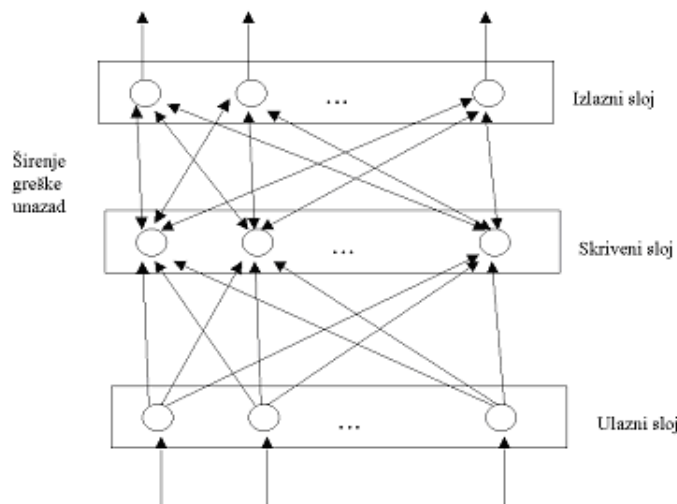
$$f(s) = \begin{cases} 1 & \text{if } s \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

## 6. Tipovi neuronskih mreža ?



## 7. Što je backpropagation ?

- Ova mreža propagira input kroz mrežu od ulaznog do izlaznog sloja, a zatim **određuje grešku i tu grešku propagira unazad sve do ulaznog sloja ugrađujući je u formulu za učenje**. Standardni algoritam mreže "širenje unatrag" uključuje **optimizaciju greške koristeći deterministički algoritam** gradijentnog opadanja.
- Glavni nedostatak ovog algoritma je **problem čestog pronalaženja lokalnog umjesto globalnog minimuma greške**, stoga novija istraživanja uključuju njegovo unapređivanje nekim drugim determinističkim (npr. metode drugoga reda) ili stohastičkim metodama (npr. simulirano kaljenje).
- Strukturu mreže čine ulazni sloj, izlazni sloja i najmanje jedan skriveni sloj, s vezom unaprijed. Tipična arhitektura "širenje unatrag" prikazana je na donjoj slici (zbog jasnoće je prikazan samo jedan skriveni sloj):



- **Tok podataka kroz mrežu** može se ukratko opisati u nekoliko koraka:
  - od ulaznog sloja prema skrivenom sloju: **ulazni sloj učitava podatke** iz ulaznog vektora  $X$ , i šalje ih u prvi skriveni sloj,
  - **u skrivenom sloju**: jedinice u skrivenom sloju primaju vagani ulaz i prenose ga u naredni skriveni ili u izlazni sloj koristeći prijenosnu funkciju,
  - kako informacije putuju kroz mrežu, računaju se **sumirani ulazi i izlazi** za svaku jedinicu obrade,
  - **u izlaznom sloju**: za svaku jedinicu obrade, računa se skalirana lokalna greška koja se upotrebljava u određivanju povećanja ili smanjenja težina,
  - **propagiranje unazad** od izlaznog sloja do skrivenih slojeva: skalirana lokalna greška, te povećanje ili smanjenje težina računa se za svaki sloj unazad, počevši od sloja neposredno ispod izlaznog sve do prvog skrivenog sloja, i težine se podešavaju.
- Mreža "širenje unatrag" je **univerzalni algoritam primjenjiv na probleme previđanja**, gdje je potrebno predvidjeti vrijednost jedne ili više izlaznih varijabli, no moguće ga je koristiti i za probleme klasifikacije, gdje se ulazni vektor raspoređuje u jednu od klasa zadanih na izlazu.
- Mreža "širenje unatrag" ne preporuča se za upotrebu na nestacionarnim podacima, ili za slučajeve kada podaci u sebi skrivaju više, u osnovi različitih, problema. Rješenje za takve probleme može se pronaći u upotrebi nekoliko neuronskih mreža od kojih će svaka rješavati pojedini problem zasebno, ili u izboru nekog drugog algoritma.

## 8. Što je evolucijska robotika, a što evolucijsko računarstvo ?

- **Evolucijsko računarstvo** je skup algoritama za globalnu optimizaciju inspirirana biološkom evolucijom.
- **Evolucijska robotika** je metodologija koja koristi evolucijsko računarstvo u razvoju upravljačkih sustava za **autonome robote**.
- Radi na razvoju robota koji se mogu adaptirati svom radnom okruženju kroz proces koji je sličan prirodnoj evoluciji
- Primjenjuje se pri razvoju upravljačkog sustava robota.
- Za razliku od umjetne inteligencije **ER** daje veću važnost interakciji mozga, tijela i okoline što je ključno za nastanak i razvoj inteligentnog adaptivnog ponašanja i kognitivne procese.

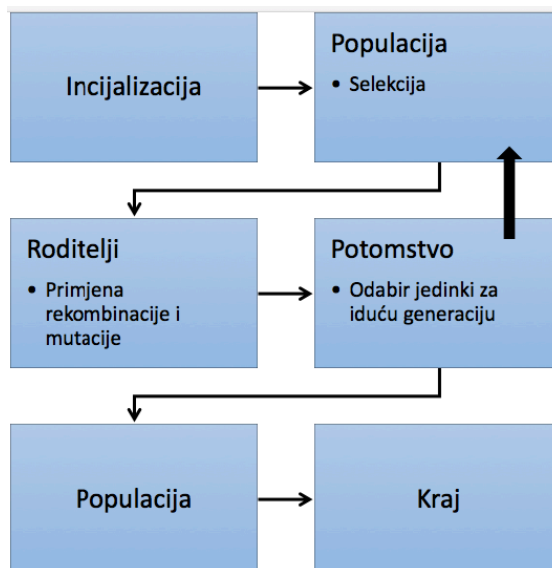
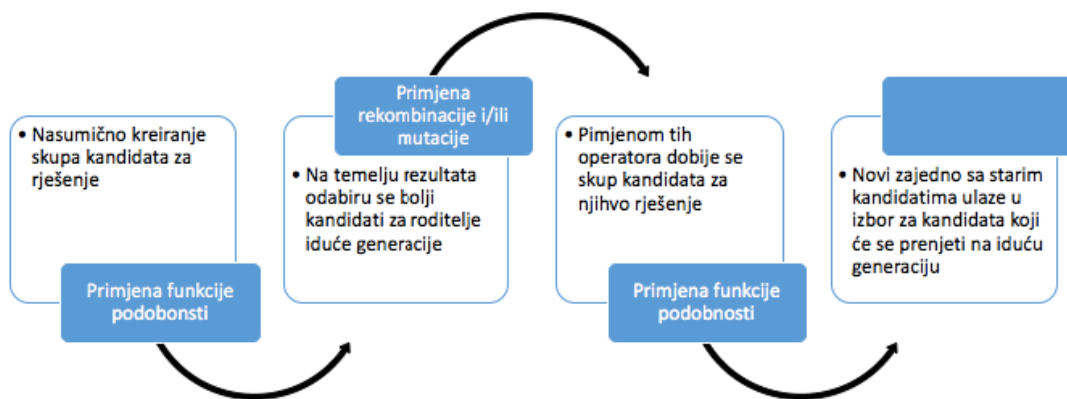
## 9. Što je evolucijski algoritam?

- **Evolucijski algoritmi** posebna su vrsta algoritama koja je **inspirirana procesom evolucije**.
- **Glavna ideja** evolucijskih algoritama je koristeći **metodu pokušaja i pogrešaka, simulirati proces evolucije** te ga primijeniti **na rješavanje problema**.
- U teoriji evolucije, neku okolinu nastanjuje populacija kojima je cilj preživjeti i razmnožavati se.
- Podobnost (engl. fitness) tih jedinki govori nam koliko je pojedina jedinka uspješna u ispunjavanju tih ciljeva, dakle predstavlja šansu jedinke da preživi dovoljno dugo kako bi se razmnožavala.
- U kontekstu rješavanja problema jedinke izjednačavamo s kandidatima za rješenje.
- **Kvaliteta** tih potencijalnih **rješenja koja dovoljno dobro aproksimiraju rješenje** problema, možemo iskoristiti kako bi odlučili **s kojom vjerojatnošću** će određeni kandidat za rješavanje sudjelovati u konstrukciji idućih kandidata
- Za danu funkciju podobnosti koju trebamo maksimizirati, nasumično kreiramo skup kandidata za rješenje te na njih primijenimo funkciju.
- **Na temelju rezultata**, možemo odabrati **bolje kandidate kao roditelje** sljedeće generacije i na njih **primijeniti rekombinaciju i/ili mutaciju**
- Primjenom tih operatora dobijemo skup kandidata za rješenje za koje opet izračunamo njihovu podobnost.
- Nakon toga, novi zajedno sa starim kandidatima ulaze u izbor za kandidata koji će se prenijeti na sljedeću generaciju.
- Taj **izbor** najčešće se zasniva **na temelju podobnosti** (ulogu može igrati i starost kandidata). Opisani **proces ponavljamo** sve dok ne pronađemo **dovoljno dobrog** kandidata za rješenje.

## 10. Komponente evolucijskog algoritma ?

- Komponente, procedure i operatori koji moraju biti zadani u Evolucijskom algoritmu kako bi bio dobro definiran su:
  - Reprerentacija jedinki
  - Funkcija podobnosti
  - Populacija
  - Mehanizam odabira roditelja
  - Operatori mutacije i rekombinacije
  - Mehanizam odabira jedinki koje će preživjeti do sljedeće generacije.
  - Način inicijalizacije populacije
  - Uvjet zaustavljanja

## 11. Shematski prikaz evolucijskog algoritma?



## 12. Što je fenotip, a što genotip?

- Moramo definirati vezu između prostora u kojem se nalazi problem i prostora u kojem će EA tražiti rješenje.
- Kandidate za rješenje **unutar prostora u kojem se nalazi problem** zovemo **fenotipi**, dok njihove **reprezentante u EA** zovemo **genotipi**.
- Reprezentacija je **početni korak** u kojem specificiramo **preslikavanje iz skupa fenotipa u skup genotipa** koji ih reprezentiraju.
- To preslikavanje **mora biti invertibilno**; za svaki genotip mora postojati najviše jedan fenotip. Kao trivijalan primjer možemo promatrati problem optimizacije na N.
- Tada je N skup fenotipa, dok kao primjer genotipa možemo promatrati binarni zapis broja.
- Važno je napomenuti da prostor u kojem se nalaze fenotipi može biti bitno različit od prostora u kojem se nalaze genotipi.
- EA rješava problem u prostoru genotipa; rjesenje problema u fenotipskom prostoru dobivamo dekodiranjem genotipa nakon kraja EA.

### 13. Što je funkcija podobnosti ?

- Funkcijom podobnosti **reprezentiramo uvjete na koje se jedinka mora prilagoditi.**
- Ona omogućuje selekciju, točnije pomoću nje možemo **ocijeniti podobnost neke jedinke i definirati poboljšanje.**
- To je funkcija koja svakom **genotipu pridružuje mjeru njegove kvalitete.**
- Matematički gledano, ta funkcija je kompozicija neke mjere kvalitete i inverza reprezentacije.
- Npr. Želimo li maksimizirati  $x^2$  na  $N$ , kvalitetu genotipa 10010 definiramo kao kvadrat odgovarajućeg fenotipa  $18^2 = 324$ .

### 14. Mehanizam odabira roditelja?

- Uloga mehanizma odabira roditelja je da, **među svim raspoloživim jedinkama,** omogući **odabir "boljih" roditelja** (naravno, boljih u kontekstu bolje podobnosti).
- Zajedno s mehanizmom odabira jedinki koje će preživjeti do sljedeće generacije, odabir roditelja **uzrok je poboljšavanja svojstava populacije kroz generacije.**
- Napomenimo da je odabir roditelja najčešće **probabilistički**, pa iako pojedinci s boljom podobnosti imaju bitno veće šanse da budu odabrani, **postoji i šansa da se za roditelje odaberu i manje kvalitetne jedinke.**
- Iako na prvi pogled nije jasna prednost takvog pristupa, **kad bi uvijek uzimali samo najkvalitetnije pojedince, algoritam bi bio previše pohlepan i postojala bi velika šansa da zapnemo u lokalnom optimumu**

### 15. Operatori varijacije ? Što je mutacija, a što rekombinacija ?

- Operatori varijacije **stvaraju nove jedinke iz već postojećih.**
- U odgovarajućem fenotipskom prostoru to je ekvivalentno generiranju novih kandidata za rješenje. Iz perspektive "generiraj i testiraj" algoritama, operatori varijacije odgovaraju "generiraj" koraku.
- Operatore varijacije u EA **dijelimo na 2 tipa**, ovisno o tome jesu li **unarni ili binarni.**
- **Operator mutacije**
  - **Unarni operator varijacije nazivamo operator mutacije. Ulazni podatak operatora mutacije je jedan genotip, a izlazni je modificirani genotip. Promjena između ta dva genotipa je obično mala.**
  - Mutacija je uvijek **stohastički operator**; genotip potomka je rezultat niza nasumičnih promjena. Važno je napomenuti da svaki unarni operator ne smatramo nužno mutacijom, ukoliko ne ispunjava i uvjete nepristranosti i nasumičnosti. Zbog toga se mnogi heuristički operatori ne smatraju mutacijama u strogom smislu riječi.
  - Generiranje potomka možemo promatrati i kao pomak na neku drugu točku u genotipskom prostoru. Iz te perspektive, operator mutacije ima i teorijsku ulogu, naime, on nam jamči da je prostor povezan
  - Postoje teoremi koji nam jamče da će EA naći globalni optimum (uz dovoljno vremena), uz uvjet povezanosti prostora.
  - Taj zahtjev trivijalno zadovoljava operator mutacije definiran tako da u svakom koraku može, uz pozitivnu vjerojatnost, skočiti u bilo koju točku prostora.



- **Operator Rekombinacije:**
  - **Binarni operator varijacije** nazivamo operator rekombinacije
  - **spaja genetske informacije dva** (ili više) roditelja **u jedan** ili više potomaka.
  - operator rekombinacije je **stohastički**; **izbor dijelova** koji će svaki roditelj prenijeti na potomke te način spajanja tih dijelova je **nasumičan**
  - Razlog zbog čega je rekombinacija poželjna je jednostavan; **rezultat kombiniranja genetskog materijala više jedinki s različitim, ali poželjnim karakteristikama, je potomak koji bi mogao objedinjavati te karakteristike.**
  - U EA, stvara se **određeni broj** potomaka **nasumičnom rekombinacijom** te se nadamo da će neki od njih imati bolje karakteristike od svojih roditelja.

## 16. Mehanizam odabira jedinki koje preživljavaju do sljedeće generacije

- Ovaj mehanizam **omogućuje** nam **odabir jedinki** koje će **preživjeti**, uzimajući **u obzir** njihovu **podobnost** ili neka druga svojstva.
- Po tome je **sličan mehanizmu odabira roditelja**, no obavlja se u drugoj fazi evolucijskog ciklusa.
- **veličina populacije** je skoro uvijek **konstantna**, pa nam je potreban neki mehanizam koji će **odlučivati koje** jedinke **će opstati** do sljedeće generacije.
- Za razliku od mehanizma odabira roditelja, kao i operatora mutacije i rekombinacije koji su stohastički, **ovaj operator je deterministički.**
- Najčešće se odabire najboljih  $n$  jedinki ili se selekcija vrši s obzirom na dob

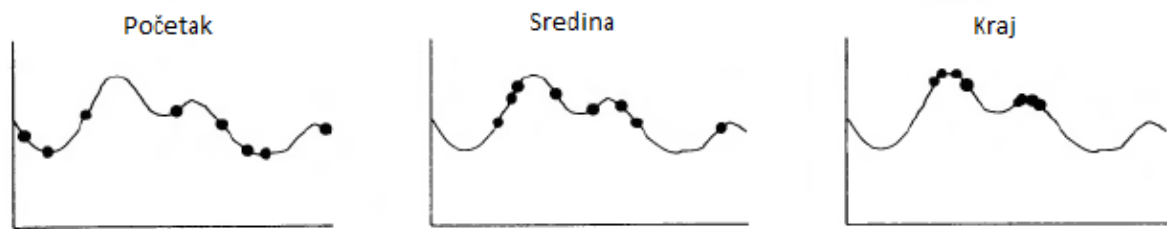
## 17. Načini inicijalizacije populacije?

- Inicijalizacija je u većini implementacija EA relativno jednostavna.
- **Prva populacija** često je samo **skup nasumično odabranih jedinki**. No, možemo i iskoristiti neku **heurističku metodu** koja bi nam, ovisno o problemu dala kvalitetniju inicijalnu populaciju.
- **Heuristika** – iskustvena pravila o prirodi problema i osobinama cilja čija je svrha pretraživanje brže usmjeriti ka cilju
  - Heuristička funkcija  $h: S \rightarrow R^+$  pridjeljuje svakom stanju  $s \in S$  procjenu udaljenosti od tog stanja do ciljnog stanja
  - Što je vrijednost  $h(s)$  manja, to je čvor  $s$  bliži ciljnome stanju. Ako je  $s$  ciljno stanje  $h(s) = 0$

## 18. Uvjet zaustavljanja?

- Ukoliko se bavimo problemom za koji nam je **poznat globalni optimum**, tada je najbolje kao **uvjet zaustavljanja** uzeti **dovoljno dobru aproksimaciju** tog optimuma (dakle približavanje optimumu na manje od nekog  $\epsilon$ )
- Međutim, kako su EA stohastički, uglavnom nije zajamčeno postizanje globalnog optimuma, pa se može dogoditi da taj uvjet nikad neće biti zadovoljen.
- Dakle, potrebno je dodati još jedan uvjet zaustavljanja koji sigurno zaustavlja algoritam. Obično se koriste neke od sljedećih opcija:
  1. Prekoračeno vrijeme izvršavanja.
  2. Prekoračen neki broj evaluacija podobnosti.
  3. Za neki broj generacija (ili evaluacija podobnosti), poboljšanje podobnosti je manje od neke vrijednosti.
  4. Raznolikost populacije pala je ispod neke vrijednosti. Uvjet zaustavljanja najčešće je zadan kao disjunkcija:
    - ili je postignut optimum,
    - ili je zadovoljen neki od uvjeta 1-4.

## 19. Rad Evolucijskog algoritma ?



- Pretpostavimo da želimo **maksimizirati** jednodimenzionalnu funkciju cilja.
- Prethodna slika prikazuje tri faze kroz koje prolazi evolucijski algoritam, konkretno kako je **populacija distribuirana** na **početku**, u **sredini** i na **kraju evolucije**.
  - U **prvoj fazi**, odmah nakon inicijalizacije, jedinke su **uniformno raspodijeljene** na cijelom **prostoru** pretraživanja. Nakon početka rada EA situacija se mijenja.
  - Zbog **mutacije** i **rekombinacije** populacija se postupno počinje "penjati" prema područjima s boljom podobnošću.
  - **Još kasnije** (blizu kraja pretraživanja, uz dobro zadane uvjete zaustavljanja), cijela **populacija koncentrirala se oko nekoliko lokalnih maksimuma**.
- Različite faze traženja rješenja često se **kategoriziraju s obzirom** na:
  - **proces istraživanja** (eng. exploration) tijekom čega se generiraju nove jedinke u još neistraženim dijelovima
  - **prostora iskorištavanja** (eng. exploitation) za vrijeme čega se pretraživanje koncentrira oko već postojećih dobrih rješenja.
- **Kod traženja rješenja evolucijskim algoritmima** moramo voditi računa o **balansu** te **dvije komponente**;
  - koncentriramo li se **previše na istraživanje**, naš će **algoritam** biti **neefikasan**,
  - dok će se u suprotnom slučaju **algoritam prebrzo fokusirati** na neki **mali dio prostora** što može dovesti do prerane konvergencije i zapinjanja u lokalnom optimumu.
- U većini slučajeva heuristika je neisplativa, jer će se ta razina podobnosti ionako postići već kroz nekoliko prvih generacija algoritma.

## 20. Dijelovi genetskog algoritma ?

- **Genetski algoritmi** su **podvrsta Evolucijskih** algoritama, koji su ujedno i **najčešće korištena** grana EA.
- Ne postoji jedan način za kreiranje GA; on nastaje **kombinacijom različitih operatora** koji su prikladni za rješavanje nekog konkretnog problema.
- Genetski algoritma sastoji se od:
  - Reprezentacija jedinki,
  - Operatori varijacije
    - Mutacija,
    - Rekombinacija,
  - Model populacije
  - Selekcija roditelja
  - Odabir jedinki koje preživljavaju smjenu generacija

## 21. Modeli populacije ?

- Model populacije jedan od bitnih dijelova svakog GA
- **Način na koji vršimo selekciju jedinku koje će preživjeti prelazi iz generacije u generaciju, baziran na njihovoj podobnosti.**
- Postoji jasna razlika između 2 modela GA, **generacijskog modela i modela stacionarnog stanja**
- **GENERACIJSKI MODEL**- U svakoj generaciji imamo populaciju veličine iz koje odabiremo roditelja (mogu se i ponavljati). Nakon toga, kreiramo  $\lambda(=\mu)$  potomaka pomoću operatora varijacije. **Nakon svake generacije, cijela se generacija zamijeni sa svojim potomstvom.**
- **MODEL STACIONARNOG STANJA** - cijela populacija se ne mijenja istovremeno. Umjesto toga mijenjamo  $\lambda(< \mu)$  jedinki iz prethodne generacije sa  $\lambda$  novih. Postotak zamijenjenih jedinki nazivamo generacijski jaz (eng. generation gap),  $\lambda/\mu$
- **operatori selekcije i zamjene** koji nam omogućuju gore opisan postupak
- **baziraju na podobnosti** jedinki, dakle za razliku od operatora varijacije uzimaju u obzir cijelu jedinku, umjesto samo neke gene.
- Kao posljedica toga operatori selekcije i zamjene nezavisni su o reprezentaciji problema. Također napomenimo da postoje 2 mjesta u evolucijskom ciklusu (vidi sliku 1.1) gdje možemo primijeniti te operatore: tijekom selekcije jedinki koje će se razmnožavati te tijekom selekcije jedinki koje će preživjeti smjenu generacija.

## 22. Selekcija roditelja ?

- Selekcija roditelja:
  - **Selekcija proporcionalna podobnosti** – u ovom načinu selekcije vjerojatnost da jedinka  $f_i$  bude odabrana za razmnožavanje iznosi

$$\frac{f_i}{\sum_{j=1}^{\mu} f_j} - \text{odnos podobnosti jedinke } i \text{ i podobnosti cijele populacije}$$

- **Selekcija po rang** – Ideja ove metode je selekcija jedinki bazirana na njihovom poretku u populaciji, umjesto na samoj vrijednosti podobnosti, prvo poredamo sve jedinke po vrijednosti podobnosti, a nakon toga im pridružimo neku vjerojatnost odabira
- **Turnirska selekcija**
  - Prethodne dvije metode zasnivaju se na podacima o cijeloj populaciji
  - Ako je populacija vrlo velika, doći do podataka o cijeloj populaciji može biti zahtjevno.
  - Ponekad nije moguće definirati podobnost kao vrijednost.
  - Turnirska selekcija ne zahtjeva znanje o cijeloj populaciji, dovoljno je imati relaciju koja može usporediti bilo koje 2 jedinke.

### 23. Odabir jedinki koje preživljavaju smjenu generacija ?

- Ovaj mehanizam odgovoran je za proces **odabira jedinki iz skupa od roditelja i djece koji će preći u sljedeću generaciju**. Glavne kategorije su mehanizmi koji biraju **na osnovu podobnosti** i oni koji biraju **na osnovu dobi**.
- Postoji:
  - Izbor na osnovu dobi,
  - Izbor na osnovu podobnosti (najgora jedinka – Genitor, najbolja jedinka – Elitizam)

### 24. Što je genitor, a što elitizam ?

- **GENITOR:**
  - **Najgorih  $\lambda$  jedinki odaberemo i zamijenimo.**
  - Iako ovakav način zamjene **može dovesti do jako brzog napretka podobnosti cijele populacije, riskiramo prebrzu konvergenciju prema jedinki s najvećom podobnosti.**
  - Zbog toga se često koristi kod velikih populacija te se ne dozvoljava dupliciranje jedinki.
- **ELITIZAM:**
  - **Cilj kod primjene elitizma je spriječiti gubitak najbolje jedinke u populaciji.**
  - To se postiže tako da se **u svakom trenutku prati trenutačno najkvalitetniji član te se njega uvijek ostavlja u populaciji.**
  - Konkretno, ako je on jedna od jedinki označenih za zamjenu, a među potomstvom nema jedinki s boljom kvalitetom, umjesto jednog od potomaka u populaciji ostavljamo tu, najkvalitetniju jedinku.

### 25. Što je planiranje putanje (eng. Motion Planing) ? Motion planing statement ?

- **Planiranje putanje** predstavlja **određivanje krivulja u prostoru** kroz koje robot mora proći pri obavljanju svog zadatka, pri čemu od početne do krajnje točke treba izbjegavati prepreke i objekte.
- **problematika planiranja putanje** dijeli se na sljedeće probleme:
  - početni položaj i orijentacija robota
  - krajnji položaj i orijentacija robota
  - geometrijski opis robota
  - geometrijski opis radnog prostora

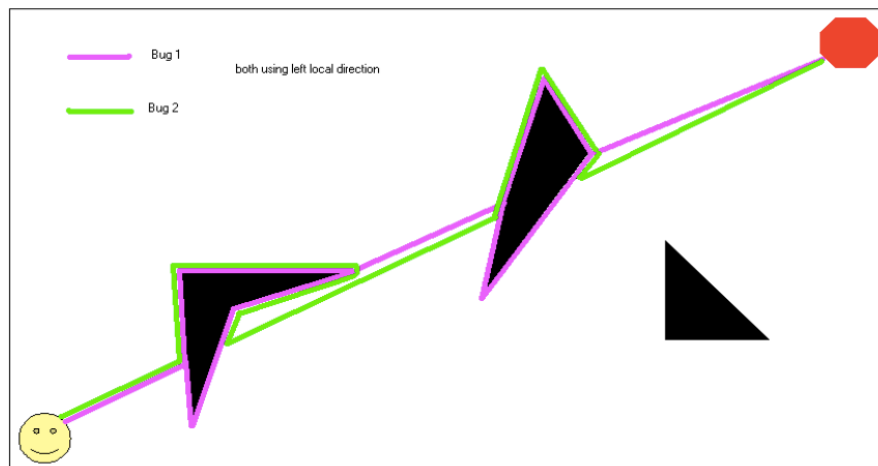
Motion planing statement

- Ako  $W$  predstavlja radni prostor robota, a  $C_i$  predstavlja prepreke, prostor kretanja robota  $FS$  definiran je kao

$$FS = W - ( \cup C_i )$$

Putanja  $c \in C^0$  je  $c : [0,1] \rightarrow FS$  pri čemu je  $c(0) q_{start}$  a  $c(1) q_{cilj}$

## 26. Start-Goal Algorithm: Lumelsky Bug Algorithms (skica i objašnjenje)?



- Nepoznate su prepreke, poznat je početni i krajnji položaj
- Jednostavni senzori dodira s pripadajućim enkoderima
- odabir smjera zakretanja u slučaju dolaska do prepreke, tj. lokalnog smjera
- kretanje je kao kod mrava:
- bug 1 - robot se kreće oko svake prepreke kojoj priđe, pri tome spremajući položaj točke najbliže cilju, te ponovo okruži prepreku kako bi nastavio prema cilju od točke najbliže cilju
- bug 2 - robot se kreće oko svake prepreke kojoj priđe sve dok ne može nastaviti predviđenu putanju za dolazak do cilja

## 27. Koordinirano istraživanje (eng. Coordinated Exploration) ?

- Problematika istraživanja prostora u robotici je jedan od fundamentalnih zadataka mobilne robotike. Primjene robota u istraživanju prostora su: istraživanje nepoznatih zatvorenih i otvorenih prostora, spašavanje, čišćenje...
- Korištenje skupine robota ima nekoliko prednosti nad korištenjem sustava s jednim robotom, npr. : kooperativni roboti imaju veću vjerojatnost za obavljanje zadatka znatno brže od jednog robota, kooperativna robotika dovodi do redundancije, čime sustav postaje otporniji na kvarove (jedan robot preuzme zadaću drugog), dolazi do spajanja preklapajućih informacija s više robota, čime se smanjuje i kompenzira utjecaj pogreške senzora
- Kao u slučaju samostalnog istraživanja prostora od strane jednog robota, cilj je minimizirati potrebno vrijeme za istraživanje.
- Glavna je zadaća u ovom pristupu istraživanju odabrati prikladne ciljeve za pojedinog robota, kako bi zajedno mogli simultano istraživati različite dijelove prostora.
- Kod koordiniranog je istraživanja potrebno uzeti u obzir cijenu dolaska do svake točke, odnosno cilja u prostoru, te korisnost te točke u samom istraživanju, odnosno benefit od dolaska do upravo te točke u prostoru.

## 28. Kooperativna manipulacija (eng. Cooperative Manipulation) ?

- Kooperativna manipulacija više robota na nekom objektu zahtjeva preciznu kinematičku "uigranost" robota i end-effectora kako bi se izbjegle nepotrebna naprezanja, sile i rad manipulatora.
- Manipulacija se smatra ispravnom ukoliko su željeni pokreti objekta i sile na istom precizno predviđene i nadzirane.
- Različite zadaće u manipulaciji objektom zahtijevaju kooperaciju više heterogenih robota. Međutim, kooperativna je manipulacija relativno nestabilna u slučaju nepredviđenih promjena u kinematici.
- Za kooperativnu je manipulaciju bitna ponovljivost svakog člana sustava, jer se neispravnosću jednog parametra kvar širi na ostale dijelove sustava.
  
- Kooperativna manipulacija u Swarm robotici predstavlja rješavanje relativno jednostavnih manipulacijskih zadataka od strane više jednostavnih robota.
- Takvi su roboti jednostavne arhitekture, prilagođeni za dani zadatak u sustavu, bez potrebe za end-effectorom.
- Glavna je zadaća transport objekta na određenu lokaciju, pri čemu više robota okruži predmet i pomiče ga željenom putanjom, ili jedan robot vrši pomak u prvu točku, drugi u sljedeću točku itd.
- Ovakav robotski sustav ne zahtjeva GPS, već se oslanja isključivo na ugrađene senzore za orijentaciju, pomak i sl.

## 29. Reconfigurable Robots ?

- Rekonfigurabilni roboti su modularni roboti sastavljeni od nekoliko identičnih dijelova, koji mogu preslagivanjem postići morfologiju prikladnu za potrebe zadatka
- izgrađeni su od modula, koji su zapravo manji samostalni roboti s jednostavnim funkcijama, pri čemu sadrže osnovne komponente potrebne za rad samog robota, odnosno senzore, aktuatora, baterija te procesor (logiku).
- uz to, potrebna je komunikacija s ostalim modulima, te sučelje za spajanje sa susjednim modulima
- aktuatori u modulu dozvoljavaju razmjerno kretanje sa susjednim modulom, ili za pomicanje susjednog modula
- na taj se način postiže fleksibilnost u morfologiji robota, tj. robotskog sustava, pri čemu sama morfologija ovisi o potrebama i zadaći
- rekonfigurabilni roboti posjeduju svojstvo redundancije, koje se koristi u svrhu robusnosti
- kvar u hardwareu ili softwareu na jednom modulu robota neće zaustaviti rad samog robota kao sustava
- ostali moduli preuzet će zadaću modula s greškom, kompenzirati nedostatak istog, te na taj način povećati robusnost sustava
- robusnost sustava opada kvarom svakog dodatnog modula, te mu same performanse također postupno opadaju
- ovakvi su roboti svestrani budući da im se morfologija mijenja tijekom rada prema potrebi, te su znatno jeftiniji s obzirom na mogućnost rada u različitim izvedbama, nego zasebni roboti za svaku zadaću, tj. zasebnu morfologiju
- primjeri: ATRON, CONRO