

Algoritmo genetikoak: Nola eta zergatik funtzionatzen dute?

Jose Ramon Zubizarreta

Lengoaia eta Sistema Informatikoak Saila. Informatika Fakultatea
E.H.U. 20080 DONOSTIA

Abstract

Genetic Algorithms are adaptive methods which may be used to solve search and optimisation problems. They are based on the genetic processes of biological organisms. Over many generations, natural populations evolve according to the principles of natural selection and "survival of the fittest". By mimicking this process, genetic algorithms are able to "evolve" solutions to real world problems, if they have been suitably encoded. The major advantage of this nature-based algorithm is its broad applicability, flexibility, ease of implementation, and the potential of finding near-optimal solutions.

Laburpena

Algoritmo genetikoak erruz erabiltzen dira bilaketa heuristiko eta optimizazio alorretan. Bilakera mekanismoan eta genetika naturalean sustraituta daude bere oinarriak. Belaunaldiz belaunaldi populazioa egokitzen joaten da gurutzaketa eta mutazio operatzaile genetikoak erabiliz. Funtzio objektibo batek ebazpena izan daitekeenaren egokitasuna neurtzen du. Prozesua bukatutzat ematen da ebazpen egokira heldzen bagara edo aurrez jarritako konputazio-denboraren mugak gainditu badira.

Algoritmo genetikoaren metodoak bilakaera-mekanismoan eta genetika naturalean sus-traituta daude. Naturaren bilakaeran espezie bakoitza ahal den ondoen egokitzen ahalegin-tzen da ingurune konplexu eta aldakorretara. Historian zehar espezie bakoitzak bereganatu duen “ezagumendu” guztia gordeta gelditzen da partaide bakoitzaren kromosometan.

Har dezagun adibide bat. Une batean untxi-populazio bat dugu, eta horietako batzuk azkarragoak eta argiagoak dira besteak baino. Untxi hauek beraz, errazago ihes egiten dute otsoengandik eta hauek ez dituzte irentsiko. Ondorioz, untxi azkar eta argi gehiago sortuko dituzte. Baina hala ere, zoria dela medio edo beste zerbaitegatik, untxi motel batzuek ere bizirik iraungo dute. Beraz, untxien ernalketa hasten denean material genetikoa gurutzatzen hasiko da, untxi azkarrak untxi azkarrekin gurutzatuko direlarik. Baina untxi motelak untxi argiekin ere gurutzatuko dira. Azkenean, untxi jaioberriak (batezbeste) azkarragoak eta argiagoak izango dira aurreko belaunaldi-dikoak baino; belaunaldi bakoitzean untxi azkar eta argi gehiagok bizirik irauten baitu. Garrantzizkoa da honakoa gogoratzeari: untxien kalterako, otsoek ere beren bilakaeran antzeko prozesua dutela eta haien atzaparretan eroriko den untxi gizari-joren bat beti egongo dela.

Egun algoritmo genetikoak erruz erabiltzen dira hainbat arlotan, batez ere optimizazio-problema ebazteko. Goldberg jaunak dioenez, algoritmo genetikoak oso erakargarriak dira, (a) problema zailak azkar eta era eraginkorrean ebazteko, (b) erraz delako algoritmo genetikoaren egungo simulazio eta ereduak lotzea, (d) hedagarriak direlako, eta (e) beste metodo batzuekin hibridizatzen errazak direlako.

Algoritmo genetikoak asko ugaltu dira azken urteotan; bilaketa heuristikoen alo-

rrean oso algoritmo interesgarriak suertatu baitira. *Simulated annealinga* eta estrategia ebolutiboekin batera, bilaketa naturalean eta prozesu fisikoetan oinarritzen da planteatzen diren problema ebazteko. Simulated annealinga termodinamikan oinarritzen da, annealinga interpretatuz prozedura optimizatzaile gisa optimo orokorrera hurbiltzeko. Estrategia ebolutiboak mutazioa erabiltzen du bilaketa-mekanismo gisa bilaketa espazioko eskualde egokienetara bideratzeko. Algoritmo genetikoek problema emaitza onenak aurkitzeko gurutzaketa erabiltzen dute mekanismo probabilistikoa gisa. Informazioa trukatzeko, bestalde, estrategia ebolutiboetan bilaketa burutzeko oinarritzko mekanismoa mutazioa da.

Algoritmo genetikoak: nola funtzionatzen dute?

Algoritmo genetikoek hitz egiten dugunean, maiz agertuko zaizkigu biologiarekin zerikusia duten hitz-formak, hala nola kromosoma, mutazioa, populazioa, kromosomen arteko gurutzaketa, gurasoak, seme-alabak eta abar. Ondoren, hitz hauen esanahia zein den argitzen saiatuko gara.

Algoritmo genetikoaren implementatzeko ondorengo urratsak eman behar ditugu:

1. Ebatzi nahi dugun problemari bere funtzio objektiboa atxikiko diogu. Funtzio honek ebazpena izan daitekeenaren ego-kitasuna neurtuko digu. Adibidez, pentsa dezagun $f(x)=15-x$, $x \in \{0,1,2,\dots,15\}=A$ funtzioa maximizatu $f(x^*) = \max f(x)$, $x \in A$ nahi dugula.
2. Murriztapen batzuen arabera hautagai izan daitezkeen ebazpenen populazioa hasiarazi egiten da. Normalean, ebazpen posible bakoitza x bektore gisa —kromo-

soma deituko diogu hemendik aurrera—kodeatzen da. Kromosomaren elementu bakoitzari **gene** deitzen zaio. Gene horren balioari berriz **alelo** eta betetzen duen tokiari **locus**. Funtzio bat tarte batean optimizatu nahi dugunean, kromosomen neurriak tarteko puntu adierazgarri guztiak kodetu egiten dituela ziurtatu behar dugu. Adibidez, m neurriko kromosoma alfabeto bitarreko ($\Omega = \{0, 1\}$) elementuz osatua badugu, 2^m balio kodetu ditzakegu. Gure goiko adibidean alfabeto bitarra erabiliz 4 neurriko kromosomarekin nahikoa izango zaigu.

3. Orain artekoa kontuan hartuta, populazioaren kromosoma bakoitza era egokian kodetuko da ebalua dadin. Ondoren egokitasun-balioa esleituko diogu funtzio objektiboaren arabera. Pentsa dezagun gure kasuan populazioa 4 kromosomaz osatua izango dela, $P=4$. Lehen zikloan ondorengo egoeran egon gaitezke:
 $P^0 = \{x_1^0, x_2^0, x_3^0, x_4^0\}$ x_i bakoitza ausazko aukeraketa eginez sortzen da.

biltzen da. Aukeraketa hau egiteko Otik 1era bitarteko ausazko zenbaki errealak sortzen dira gurutzatuak izango diren gurasoak aukeratzeko. Probabilitate metatuek kromosoma bakoitza noiz aukera daitekeen adierazten dute. Adibidez, pentsa dezagun 0.712 zenbakia sortu dugula, zenbaki honek hirugarren kromosoma aukeratu behar dugula esaten digu; zenbaki hau probabilitate metatuen 0.457 eta 0.828 tartean baitago. Gure adibiderako ondorengo ausazko zenbakiak sortu ditugu:

	zenbakia		kromosoma
1.-	0.712	→	$x_3^1 = 0010$
2.-	0.919	→	$x_4^1 = 1001$
3.-	0.317	→	$x_2^1 = 1000$
4.-	0.787	→	$x_3^1 = 0010$
5.-	0.517	→	$x_3^1 = 0010$
6.-	0.144	→	$x_1^1 = 0110$
7.-	0.018	→	$x_1^1 = 0110$
8.-	0.667	→	$x_3^1 = 0010$

Kromosoma	Alfabeto bitarra	Balio hamartarra	Egokitasun-maila	Probabilitatea	Probabilitate metatua
x_1^0	0110	6	9	$9/35 = 0.257$	0.257
x_2^0	1000	8	7	$7/35 = 0.200$	0.457
x_3^0	0010	2	13	$13/35 = 0.371$	0.828
x_4^0	1001	9	6	$6/35 = 0.172$	1.000
			35		

4. Kromosoma bakoitzari p_i , $i=1, \dots, P$ birsorkuntza-probabilitatea esleituko diogu. Probabilitate honek populazioaren beste kromosomekiko kromosomaren egokitasun errelatiboa zein den adierazten digu. Askotan hau egiteko erruleta-gurpilararen aukeraketa (*roulette wheel selection*) erabiltzen da. Aukeraketa hau egiteko Otik 1era bitarteko ausazko zenbaki errealak sortzen dira gurutzatuak izango diren gurasoak aukeratzeko. Probabilitate metatuek kromosoma bakoitza noiz aukera daitekeen adierazten dute. Adibidez, pentsa dezagun 0.712 zenbakia sortu dugula, zenbaki honek hirugarren kromosoma aukeratu behar dugula esaten digu; zenbaki hau probabilitate metatuen 0.457 eta 0.828 tartean baitago. Gure adibiderako ondorengo ausazko zenbakiak sortu ditugu:
5. Ondoko probabilitateak p_i , $i=1, \dots, P$, kontuan izanda kromosoma-populazio berria sortzen da, dugun populaziotik stringak (kromosomak) probabilistikoki aukeratu. Aukeraketako kromosomek seme-alabak sortzen dituzte gurutzaketa eta mutazio operatzaile genetikoak erabiliz.

Gurutzaketa operatzailea kromosoma “guraso” biri aplikatzen zaie, beste bi kromosoma seme edo alaba sortzeko. Horretarako ausazko posizio bat aukeratu da, kromosomaren neurrietan sartzen dena, eta bi gurasoen posizio honen aurreko eta atzeko zatiak trukatu egiten dira bi seme (kromosoma) berri sortuz. Adibidez, gurutzaketa burutzeko 0010 eta 1001 kromosomak aukeratu badira eta ausazko gurutzaketa-posizioa bi bada, mutaziorik gabe, ondorengo bi semeak sortzen zaizkigu: 0001 eta 1010, hau da $x=1$ eta $x=10$. Gure adibidean, sortzen zaizkigun bi semeetatik lehenengoak aukeratuz, ondorengo semeak sortuko zaizkigu:

berrien egokitasun-maila lortuz eta zikloa errepikatuz.

Algoritmo genetikoak: zergatik funtzionatzen dute?

Algoritmo genetikoak nola funtzionatzen duten ikusi ondoren, zergatik funtzionatzen duten aztertzen saiatuko gara ondorengo lerroetan.

Algoritmo genetikoak emaitza egokiak lortzea nola hurbiltzen den ulertzeko, eskemek nola funtzionatzen duten aztertu behar da. Eskema, kromosomen arteko antzekotasunak aztertzeko erabiltzen den adierazpidea dela esan dezakegu. Geneen alfabetoan ikur komodinak (*) sartuz sortzen ditugu eske-

Urrats	Guraso 1	Guraso 2	Gurutzaketa Posizioa	Mutazioa	Semea	
					String	f(x)
1	3	4	2	ez	0001	14
2	2	3	3	ez	1000	7
3	3	1	2	ez	0010	13
4	1	3	1	bai	0011	12
						46

Mutazioak kromosoma sortu berrien bit bat edo beste aldatzen du, probabilitate baten arabera kromosoma berria edo berriak sortuz. Bit-mutazioak ez du kromosoma-kopurua aldatzen; gurutzaketa bidez sortutako kromosomaren bat edo beste aldatzen du baizik. Jarraitzen ari garen adibidean ondorengo gurutzaketa- (P_g) eta mutazio- (P_m)-probabilitateak erabiliko ditugu, $P_g=0.9$ eta $P_m=0.01$.

6. Prozesua bukatutzat ematen da ebazpen egokira heltzen bagara edo aurrez jarritako konputazio-denboraren mugak gainditu badira. Bestela prozesua hirugarren urratsera bideratuko dugu, kromosoma

mak. Adibidez **110 eskemak, azken hiru posizioetan 110 balioak dituzten bost neurriko stringak adierazteko balio digu, hau da, 00110, 01110, 10110, 11110. Beraz, argi ikusten da eskema batek n ikur komodin baditu 2^n string adierazteko balio digula.

Eskemek bi propietate garrantzitsu dituzte: **ordena** eta **neurria**. Oso interesgarriak dira ondoren aztertuko dugun ernalketa-formula definitzeko eta neurri batean, eskema batzuk beste batzuk baino interesgarriago zergatik diren eta ondorengo belaunaldietan zergatik iraungo duten jakiteko.

E eskemaren ordenak ($o(E)$) eskema horretan azaltzen diren 0 eta 1 biten kopu-

rua neurtzen du. Ordena honek eskemaren espezialitatea neurtzen du. Adibidez, $E_1=11^{***}$ eta $E_2=01^{**}11k$ ondorengo ordenak dituzte: $o(E_1)=2$ eta $o(E_2)=4$. Gero ikusiko dugunez, eskemaren ordena oso beharrezkoa izango dugu eskemak bizirik irauteko duen probabilitatea neurtzeko mutazioa aplikatzen denean.

E eskemaren neurria ($n(E)$) eskema horretako lehen eta azken bitaren, hau da, ikur komodinaren eta ez komodinaren, arteko distantzia da. Gure aurreko adibidea hartuz $n(E_1)=3-2=1$ eta $n(E_2)=6-1=5$. Neurri honek informazioaren trinkotasuna neurtzen du. Gero ikusiko dugunez, eskemaren neurria oso beharrezkoa izango dugu eskemak bizirik irauteko duen probabilitatea neurtzeko gurutzaketa aplikatzen denean.

Dakigunez algoritmo genetikoaren prozesu nagusiak urrats garrantzitsu bi burutzen ditu sekuentzialki: *populazioa aukeratu eta ondoren gurutzaketa eta mutazio eragiketak aplikatu*. Pentsa dezagun gure sistemak 20 kromosoma izango dituela ziklo bakoitzean populazioa osatzeko eta kromosoma bakoitzak 10 ($m=10$) neurtzen duela. Populazio honetako eskema (E) bakoitzak, ondorengo egokitasun-neurria izango du t zikloan: $ebal(E,t) = \sum ebal(k_{ij})/p$, p adierazten digu dugun populazioan eskemak eskatzen dituen murriztapenak zenbat kromosomak betetzen duten, ikur komodinak eta ez komodinak kontuan hartuz. $ebal(k_{ij})$ kromosoma bakoitzaren egokitasun-maila izango da. Beraz $ebal(E,t)$ eskemaren batezbesteko egokitasun-neurria adierazten du.

$\zeta(E,t)$ t zikloan E eskemak eskatzen dituen murriztapenak betetzen dituen kromosoma-kopurua zein den adierazten badigu, $\zeta(E_i,2)=4k$ honakoa adieraziko du: 4 kromosoma daudela populazioan bigarren zikloan, E_i eskemak inposatzen dituen murriztapenak betetzen dituztenak.

Beraz $\zeta(E,t+1)$ lortzeko, hau da, $t+1$ zikloan E eskema errespetatuz jakiteko zenbat kromosoma pasatuko diren, ondorengo formula ebatzi behar dugu:

$$\zeta(E,t+1) = \zeta(E,t) \cdot 20 \cdot ebal(E,t) / F(t)$$

non $F(t) = \sum ebal(k_i)$ eta 20ak populazio-kopurua adierazten duten, edo beste era batera jarrita:

$$(1) \zeta(E,t+1) = \zeta(E,t) \cdot ebal(E,t) / F'(t)$$

non $F'(t) = F(t) / 20$

Azter dezagun adibide batekin hau guztia: Pentsa dezagun E_0 eskemak $\zeta(E_0,t)=3$, hau da, hiru kromosoma daude populazioan E_0 eskatzen dituen murriztapenak betetzen dituztenak. Gainera, jakin badakigu, $ebal(E_0,t) = (27.3167+30.0602+23.8672)/3 = 27.0814$, hau da, E_0 eskemaren egokitasun-maila $ebal(E_0,t)$ dela.

Bestalde, populazio osoaren batezbesteko egokitasun-maila ondorengoa da: $F'(t) = \sum ebal(k_i) / 20 = 387.77 / 20 = 19.388$. Balio hauekin eskemaren E_0 egokitasun-ratioa zein den aterako dugu,

$$ebal(E_0,t) / F'(t) = 1.3968$$

Orain (1) aplikatuz $\zeta(E_0,t+1) = 3.1.3968 = 4.19$, hau da, $t+1$ zikloan E_0 eskemak dituen murriztapenak 4 edo 5 kromosomak beteko dituztela espero dugu, $t+2$ zikloan $3.1.3968^2=5.85$ kromosoma izango direlarik.

Beraz, intuizioz esan dezakegu E_0 eskemak bilaketa espazio egokia definitzen duela algoritmo genetikoaren ondorengo zikloetarako.

Baina prozesu honek guztiak, interesgarria izanik, aukeraketa-urratsa besterik ez du osatzen, hau da, kromosoma batzuk kopiatzen ditu tarteko populazioa sortzeko.

Beraz, bigarren urratsera pasatu beharko dugu kromosoma berriak populazioan sartzeko eta hau egiteko gurutzaketa eta mutazioa erabiliko ditugu.

Lehen esan dugunez, eskemaren neurria oso garrantzitsua da eskemak bizirik iraungo duen ala ez jakiteko. Probabilitateak erabiliz E eskemaren bizirik ez ateratzeko probabilitatea ondorengoa dela esan dezakegu: $p_{\downarrow}(E) = n(E)/(m-1)$ eta, beraz, bizirik irautekoa: $p_{\uparrow}(E) = 1-n(E)/(m-1)$. Gurutzaketa-probabilitatea $-p_g$ kontuan izanik, azken probabilitate hau honela geldituko litzateke: $p_{\uparrow}(E) = 1-p_g \cdot n(E)/(m-1)$

Adibidez, pentsa dezagun arestian aztertu dugun E_0 eskemak ondorengo ezaugarriak dituela: $p_g=0.25$ $n(E_0)=5$. Hau jakinda, E_0 eskemak ondorengo probabilitatea du bizirik irauteko:

$$p_{\uparrow}(E_0) = 1 - 0.25 \cdot 5 / (10-1) = 0.8611$$

Une honetan $\zeta(E,t+1)$ aldatu egiten zaigu, beste itxura hau hartzen duelarik:

$$\zeta(E,t+1) \geq \zeta(E,t) \cdot (\text{ebal}(E,t)/F'(t)) [1-p_g \cdot n(E)/(m-1)]$$

Gure adibidean:

$$\zeta(E,t+1) \geq 3.1.3968.0.8611 = 3.6080$$

Beraz, E_0 eskemaren egokitasun-ratioak ona izaten segitzen du.

Mutazioarekin prozesu berari jarraituz, $\zeta(E,t+1)$ formulak honelako azken itxura hartuko luke:

$$(2) \zeta(E,t+1) \geq \zeta(E,t) \cdot (\text{ebal}(E,t)/F'(t)) [1-p_g \cdot n(E)/(m-1) - o(E) \cdot p_m]$$

Laburtuz, (2) ekuazioak honakoa adierazten digu: eskema txikia eta ordena txikikoa bada, gurutzaketaren eta mutazioaren eragina txikia dela eta gainera bere egokitasun-maila handia bada, denboran zehar espontzialki hazten dela.

BIBLIOGRAFIA

1. Diaz, A., Glover, F., Ghazin, H.M., Gonzalez, J.L., Moscato, P. eta Tsengs, F.T. Optimizacion Heuristica y Redes Neuronales. Editorial Paraninfo, 1996.
2. Fogel, D.B. An Introduction to Simulated Evolutionary Optimization. 3-14 or. IEEE Transactions on Neural Networks. Vol 5, N-1, January, 1994.
3. Goldberg, D.E. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, 1989.
4. Holland, J. H. Algoritmos geneticos. Investigacion y Ciencia. 38-45 or. Septiembre, 1992.
5. Kelly, J.P. eta Osman, I.H. Meta-heuristics: Theory and Applications. Kluwer Academic Publishers, 1996.
6. Michalewicz, Z. Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, 1992.
7. Reeves, C.R. Modern Heuristic Techniques for Combinatorial Problems. Blackwell Scientific Publications, 1993.
8. Ribeiro, J.L., Treleaven, P.C. eta Alippi, C. Genetic-Algorithms Programming Environments. 28-43 or. COMPUTER, June, 1994.
9. Srinivas, M. eta Patnaik, L.M. Genetic Algorithms: A Survey. 17-26 or. COMPUTER, June, 1994.