

Sistem za pametno generisanje protivnika u Soulslike igrama

Član Tima

- Dušan Lečić SV80/2021

Motivacija

Igre iz žanra [soulslike](#) zasnivaju se na pažljivo dizajniranom svetu. Ručno osmišljavanje svakog protivnika za svaku zonu oduzima vreme i otežava brz razvoj igre. Cilj je da pojedinac-dizajner, uz pomoć sistema baziranog na znanju, klikom definiše kontekst (npr. „noć u močvari, srednja težina“) i u roku od nekoliko sekundi dobije potpuno definisanog neprijatelja koji:

- odgovara lokaciji i narativu,
- nije kopija prethodno generisanih,
- automatski je balansiran prema jačini igrača.

Time se ubrzava ciklus dizajna i podiže raznovrsnost sadržaja.

Pregled problema

Trenutno ne postoje sistemi za automatsko generisanje protivnika u soulslike igrama bazirani na ekspertskom znanju. Proceduralna generacija koristi nasumičnost (roguelike pristup), ali gubi narativnu i balansnu kontrolu. Nedostatak postojećih rešenja:

- ne kombinuju **znanje o svetu igre** (biomi, lokacija, vreme dana) sa **znanjem o igraču** (lvl, statistike).
- ne postoji **razlog zašto** je baš određen protivnik u toj zoni – što ruši imerziju.

Moj sistem se razlikuje jer koristi **rule-based rezonovanje** (forward/backward chaining, zajedno sa template- ovima) da generiše protivnika koji ima smisla u datom kontekstu, koji je balansiran prema igraču i koji omogućava proširivanje baze znanja tokom korišćenja igre.

Metodologija rada

Ulazi u sistem:

- Tip regiona (močvara, planina, pećina, zamak, pustinja itd.)
- Težina regije (laka, srednje-laka, srednja, srednje-teška, teška)
- Informacije o igraču (nivo, klasa, korišćeno oružje)
- Vreme u igri i vremenski uslovi (dan, noć, kiša, vetar, lava, itd.)

Izlazi iz sistema:

Definisani protivnik sa sledecim osobinama:

- Tip protivnika (vojnik, čudovište, duh, goblin, ptica, vuk, boss, itd.)
- Sposobnosti (melee, ranged, magic, stealth)
- Statistike (život (HP), šteta (DMG), odbrana (DEF), otpornost (resistence))
- Ponašanje (agresivan, defanzivan, čeka u zasedi...)
- Dodatne karakteristike (otrov, led, vatra, status efekti)

Baza znanja:

- Pravila grupisana po regionu
 - Močvara – otrovno čudovište, duhovi
 - Zamak – vitezovi, čarobnjaci
 - Planina – ptice, gorske zveri
 - ...
- Pravila balansiranja (igrač niskog nivoa → slabiji neprijatelji; igrač visokog nivoa → jači neprijatelji)
- Šabloni (template) za različite vrste neprijatelja povezane sa tipom regiona.
- Mogućnost **proširivanja baze znanja tokom korišćenja** – dodavanje novih neprijatelja, regiona ili template-ova kroz interakciju sa sistemom

Primer rezonovanja (korak po korak)

Da bi dali primer rezonovanja potrebno je da imamo neki kontekst. Recimo da je kontekst (region= močvara, težina regiona = srednje-teška, vreme = noć i magla, nivo igrača = 28 i katana + dex build.

1. Nivo 1 (region) – filtriranje kandidati iz šablona za „močvaru“: *Swamp Witch*, *Swamp Goblin*...
2. Nivo 2 (balans na osnovu nivoa regije) – zbog *srednje-teška*, prilagođava se HP i DMG za dobar balans igre
3. Nivo 3 (balans na osnovu nivoa igrača) – zbog nivoa 28, prilagođava se HP i DMG za dobar balans igre, ovo uključuje male korekcije na statistike, na primer povećanje HP i DMG protivnika za 5%
4. Nivo 4 (balans na osnovu vremenskih uslova)
5. Nivo 5 (accumulate funkcija) – evaluira kandidate I bira najprikladnijeg

Forward Chaining

Primer ulančavanja forward chaining-a:

- provera regiona => provera težine => nivo i klasa igrača => vremenski uslovi => selekcija protivnika

Backward Chaining

Backward chaining se koristi tako sto cilj (konkretnog protivnika) zadajemo kao cvor stabla, a zatim se sistem reukrizno spusta kroz pravila i uslove koji vode do tog cilja. Razlikuju se dva tipa cvorova(pravila), AND cvor i OR cvor. AND cvorovi zahtevaju da su sva pravila zadovoljena. Medjutim, ako pravila nisu zadovoljena onda da radi selekciju protivnika po OR cvorovima.

- Cilj: Pokusaj da generises protivnika *Father of Flame (boss fight)*, taj protivnik se postavlja kao cvor stabla odlucivanja, ako su svi uslovi za tog protivnika ispunjeni, vrati tog protivnika
- U suprotnom, napravi sledeci nivo stabla sa svim protivnicima iz regiona kao I izabrani protivnik I prover i da li su za njih uslovi ispunjeni, vrati prvog protivnika koji ispunjava uslove
- Opcionalno: *Modifikacija izabranog protivnika (omogucavaju bolji balans)*
- Kada je protivnik izabran, moguće je izvršiti dopunska pravila radi prilagođavanja balansa I dinamike igre, dopunska pravila se primenjuju na protivnika i daju mu dodatne karakteristike

Struktura stabla za backward chaining:

- Nivo 1 – OR čvor: izbor regiona
 - Kandidati: močvara, planina, zamak, pustinja...
 - Sistem bira region kompatibilan sa ciljnim neprijateljem.
- Nivo 2 – AND čvor: težina regiona i nivo igrača
 - Nivo igrača mora biti dovoljan za izazov.
- Nivo 3 – AND čvor: build i oprema igrača
 - Tip build-a igrača (Dex, Strength, Magic...)
 - Tip oružja i opreme

Opcionalno: dopunska pravila

Primeri template-a

Template-ovi se koriste za generisanje gotovih protivnika. Template *Castle Monster*:

- Region: Zamak
- Težina: srednje-teška
- Sposobnosti: melee
- HP: 3580 - 3700
- DMG: 900 – 1200
- DEF: 400
- Otpornost: Vatra
- Ponašanje: defanzivan, čeka u zasedi
- Dodatno: status efekat – poison

Pravila

Template pravila:

when difficulty = Easy then DMG * 0.6, HP * 0.6

when difficulty = Medium then DMG * 1, HP * 1

when difficulty = Hard then DMG * 2, HP * 2.5

when playerLevel <= 10 then HP + HP * 0.1

when playerLevel >= 30 then HP * 1.5

when playerClass = Dex then enemyContains(poison, stealth) * 1.2

when playerClass = Strength then enemyContains(range, magic) * 1.2

when playerClass = Mage and difficulty = Hard then enemyResistance(magic)

Primeri pravila za određeni region (Zamak):

when region = Castle and difficulty = Easy then DMG * 0.6, HP * 0.6

when region = Castle and difficulty = Medium then DMG * 1, HP * 1

when region = Castle and difficulty = MediumHard then DMG * 1.5, HP * 1.5

when region = Castle and difficulty = Hard then DMG * 2.5, HP * 3

when region = Castle and weather = Rain then enemyContains(fire) -> DMG - DMG * 0.8

when region = Castle and weather = Wind then enemyType(rangeRanged) -> DMG - DMG * 0.1

when region = Castle and playerLevel <= 10 then HP + HP * 0.1

when region = Castle and playerLevel >= 100 and playerLevel <= 150 then HP * 2.7

Primeri pravila za klase klase igraca:

when playerClass = Dex then enemyContains(poison, stealth, sneakAttack) * 1.2
when playerClass = Strength then enemyContains(range, magic, armorBreak) * 1.2
when playerClass = Mage and difficulty = Hard then enemyResistance(magic)

Primer pravila za evaluaciju oruzja igraca:

when playerWeapon = Greatsword or playerWeapon = Katana and playerLevel < 50 and difficulty = MediumHard
then enemyContains(magic, range)

when playerWeaponContains(poison) and area = Swamp and difficulty = Hard
then enemyReceivesPoisonResistance()

when playerWeapon = Staff and playerClass = Mage
then enemyResistance(magic) * 1.5

Primer pravilaa za vremenske uslove:

when weather = Rain and enemyContainsStatus(fire)
then DMG - DMG * 0.8

when weather = Wind and enemyType(rangeRanged)
then DMG - DMG * 0.1

when weather = Sandstorm
then enemyContains(stealth) + 0.2

when weather = Fog
then stealth + 0.2

when weather = Snow
then enemyContainsStatus(ice) -> DMG + DMG * 0.2

Primer dopunskih pravila:

when selectedEnemy and playerLevel > enemyLevel then DMG + DMG * 0.2

when selectedEnemy and playerHP > 2000 or playerWeapon = Greatsword
then enemyBehaviour(defensive)

when selectedEnemy and regionDifficulty = Hard
then addStatusEffect(fire, poison, ice) based on enemyType

Primeri pravila za generisanje boss protivnika:

when region = Volcano and playerLevel \geq 60

then spawnBoss(FireDragon), HP * 5, DMG * 4, specialAttack(fireBall, flameBreath)

when region = Swamp and playerLevel \geq 30 and difficulty = Medium

then spawnBoss(PoisonHydra), HP * 2, DMG * 3.2, enemyResistance(poison)

when region = Castle and playerLevel \geq 80

then spawnBoss(IronKnight), HP * 3, DMG * 2.5, enemyResistance(physical)

Stara pravila (pravila koja su odbacena, ali cije ideje mogu biti korisne) :

when playerClass = Mage and difficulty = Hard then enemyResistance(magic)

when (playerWeapon = Greatsword or playerWeapon = katana) and playerLevel < 50 and
difficulty = MediumHard
then enemyContains(magic, range)

when weather = Rain and enemyContainsStatus(fire) then $DMG - DMG * 0.8$

when weather = Wind and enemyType(rangeRanged) then $DMG - DMG * 0.1$

when playerWeaponContains(poison) and area = Swamp and difficulty = Hard then
enemyReceivesPoisonResistance()

when playerHP > 2000 or playerWeapon = Greatsword then enemyBehaviour(defansive)

when region = Volcano and playerLevel >= 60
then bossFight(FireDragogn), $HP * 5$, $DMG * 4$, enemyResistance(fire),
specialAttack(fireBall, flameBreath)

when region = Swamp and playerLevel >= 30 and difficulty = Medium
then spawnBoss(PoisonHydra), $HP * 2$, $DMG * 3.2$, enemyResistance(poison)

when weather = Rain and bossType = FireDragon
then $bossDMG - bossDMG * 0.4$, $bossHP - bossHP * 0.1$