

Virtuální herní světy

Cykly činností a produkce

Radek Richtř

October 26, 2020

Eat(); Sleep(); Code(); Repeat();

Chování

Jako chování se dá označit celý soubor činností i jevů:

- dialogy postavy
- změna lokace / Pohyb postav
- jejich aktuální činnost (animace)
- oblečení
- reakce na prostředí
- atd.

Chování - Dialog

Dialog je po vzhledu ta nejpřímočarejší charakterizace (chování) postavy.

- **dialogy postavy** (P8)
- změna lokace / pohyb postav
- jejich aktuální činnost (animace)
- oblečení
- reakce na prostředí
- atd.

Chování - Lokace

Byť se to může zdát triviální, může umístění postavy hodně říct.

- Pozice postavy je neočekávaná (farmář v lese)
- Pozice postavy něco naznačuje (lovec v zámeckém lese)
- Očekávaná změna (kostel v neděli)
- Neočekávaná změna (goblin za městem)
- atd.

V dobrém návrhu lze za každou neočekávanou změnou “schovat” příběh. K tomu, aby něco bylo neočekávané, je však potřeba nejdříve vybudovat “rutinu”.

Ve špatném návrhu jsou neočekávané změny projevy chyb.

Chování - Lokace

Byť se může zdát triviální, může umístění postavy hodně říct:

- Farmář v lese - Co tam schovává? Kam šel?
- Lovec v zámeckém lese - Pytláci? Kolik dá za mlčení?
- Kostel v neděli - Jsou pak domy prázdné a nehlídané?
- Goblin za městem - Chystá tlupa něco? Vyhnal ho kmen?
- atd.

V dobrém návrhu lze za každou neočekávanou změnou “schovat” příběh. K tomu, aby něco bylo neočekávané, je však potřeba nejdříve vybudovat “rutinu”.

Ve špatném návrhu jsou neočekávané změny projevy chyb.

Chování - Lokace

Byť se může zdát triviální, může umístění postavy hodně říct.

- Každá postava má nějaké “typické” místo (místa) výskytu
- Práce, domov (dům, jeskyně, atp.)
- Základní a minimalistická je změna lokace v noci a ve dne
- Chytré určení míst výskytu a doby cesty mezi nimi vytváří dojem života
- I minimální randomizace při velkém počtu postav stačí

V dobrém návrhu lze za každou neočekávanou změnou “schovat” příběh. K tomu, aby něco bylo neočekávané, je však potřeba nejdříve vybudovat “rutinu”.

Ve špatném návrhu jsou neočekávané změny projevy chyb.

Chování - činnosti

Činnost, respektive to, co postava dělá (a že něco dělá) by mělo být nějak indikováno.

- dialogy postavy
- změna lokace / pohyb postav
- **jejich aktuální činnost**
- oblečení
- reakce na prostředí
- atd.

Chování - činnosti

Činnost, respektive to, co postava dělá (a že něco dělá) by mělo být nějak indikováno

Vizuálně nejjasnějším indikátorem činnosti je animace dané postavy.

- Animace začínají a končí ve stejném neutrálním stavu, nebo jsou vytvořeny přechody
- Animace může být v zásadě velmi obdobná a lišící se jen lokací, či nástrojem
- Základní činností je pohyb!
- Další častou činností je dialog (jak ho vizualizovat?)
- atd.

Pro účely semestrální práce je jasné, nemusí být namapovány všechny animace. I v takovém případě, by ale nějak mělo být patrné že se něco děje.

Chování - činnosti

Činnost, respektive to, co postava dělá (a že něco dělá) by mělo být nějak indikováno

Vizuálně nejjasnějším indikátorem činnosti je animace dané postavy.

- Animace začínají a končí ve stejném neutrálním stavu, nebo jsou vytvořeny přechody
- Animace může být v zásadě velmi obdobná a lišící se jen lokací, či nástrojem
- Základní činností je ~~pohyb~~ idle animace!
- Další častou činností je dialog (jak ho vizualizovat?)
- atd.

Pro účely semestrální práce je jasné, nemusí být namapovány všechny animace. I v takovém případě, by ale nějak mělo být patrné že se něco děje.

Chování - činnosti

Vizualizovat činnost bez vizualizace činnosti lze například textem postavy, nějakým indikátorem, ikonou, popiskem, atp. Metodu je přitom vhodné volit i podle tématu - nepř. v surrealisticky, či komicky založené hře projde stojící dělník stěžující si na to jak strašně moc má práce, i když bude jen stát a ani se nehne.

Chování - Oblečení

Triviální ukázkou toho, že postava reaguje na svoje okolí je případně změna oblečení (denní doba, počasí).

- dialogy postavy
- změna lokace / Pohyb postav
- jejich aktuální činnost (animace)
- **oblečení**
- reakce na prostředí
- atd.

Chování - Oblečení

Triviální ukázkou toho, že postava reaguje na svoje okolí je případně změna oblečení.

Je samozřejmě kontraproduktivní měnit kompletně a chaoticky oblečení postav, zvláště, pokud nejdou jinak rozeznat podle nějakého dominantního rysu (například konkrétní postava drží v ruce asparágus).

Změna, může být i v zásadě drobná a přesto mít velký vliv - například sluneční brýle v poledne, klobouk v dešti, zrezlá zbroj, nebo pracovní rukavice.

Chování - Reakce

Klíčovou součástí chování postavy je to jak reaguje na okolní situace (a jestli vůbec).

- dialogy postavy
- změna lokace / pohyb postav
- jejich aktuální činnost (animace)
- oblečení
- **reakce na prostředí**
- atd.

Chování - Reakce

Klíčovou součástí chování postavy je to jak reaguje na okolní situace (a jestli vůbec).

Ať už postava dělá co dělá, měla by (pokud je to vnímavá NPC) reagovat na to, co se děje v jejím okolí - komentářem, změnou činnosti, útekem před požárem, nějak.

Tím, že postavy reagují na okolí přestává být ze světa mnoho miniaturních mikroversmírů, ale stává se jedním velkým světem.

Formalizace

Předchozí text je sice pěkný (je?), obsahuje spoustu zajímavých (snad?) informací, pro přenesení do virtuálního světa je však zapotřebí jej nějak konkrétněji formalizovat.

Řekněme, že to, co aktuálně postava dělá, co provádí, i to jak zrovna reaguje je **stav**.

Formalizace

Řekněme, že to, co aktuálně postava dělá, co provádí, i to jak zrovna reaguje je **stav**:

- spánek
- pohyb někam
- rychlý pohyb někam
- skok
- práce
- panika
- dialog
- je zraněn
- mrtev, atd.

Formalizace

Výčet vztahů by měl být pro daný svět/hru a typ postav kompletní, byť je jasné, že nemusí každá postava využít všechny, teoreticky tak tomu být může (KC:D).

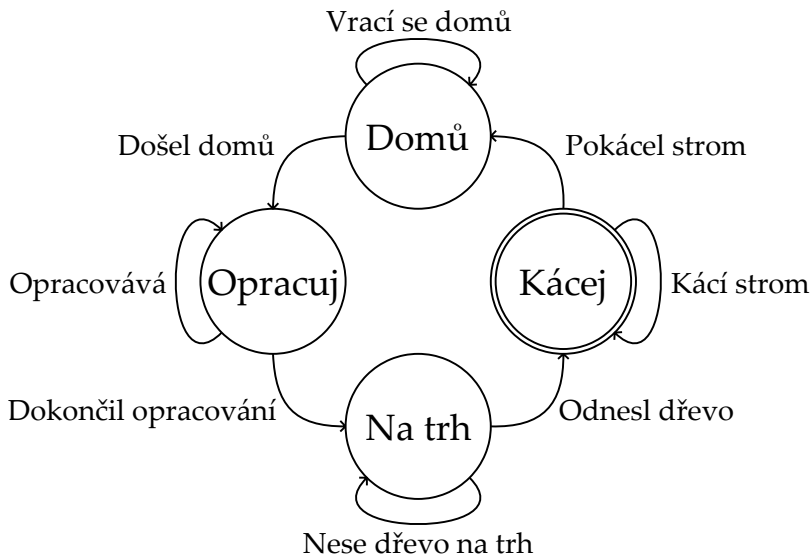
- spánek
- pohyb někam
- rychlý pohyb někam
- skok
- práce
- panika
- dialog
- je zraněn
- mrtev, atd.

Formalizace

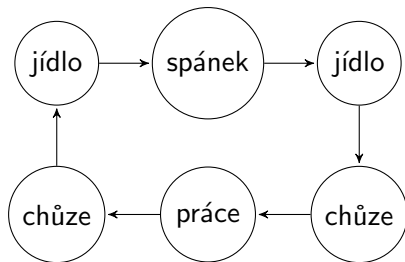
Můžeme bezpečně stavy prohlásit za stavy (uzly) stavového automatu. Výsledný automat pak bude charakterizovat chování určité postavy.

- Spící postava
- pohyb někam
- rychlý pohyb někam
- skok
- práce
- panika
- dialog
- je zraněn
- mrtev, atd.

Graf 1

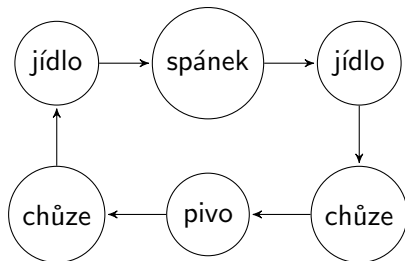


Graf 2



Kdy se postava vzbudí? Bude to vždy ve stejnou dobu? Je třeba animovat její spánek? Co když po cestě někoho potká? Jak rychle půjde z práce a jak rychle do ní?

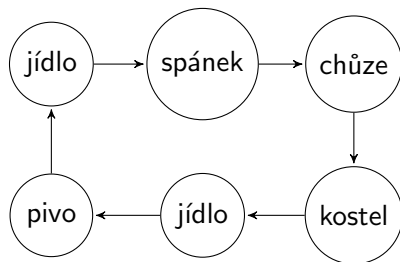
Graf 2



Kdy se postava vzbudí? Bude to vždy ve stejnou dobu? Je třeba animovat její spánek? Co když po cestě někoho potká? Jak rychle půjde z práce a jak rychle do ní?

V sobotu se nepracuje!

Graf 2



Kdy se postava vzbudí? Bude to vždy ve stejnou dobu? Je třeba animovat její spánek? Co když po cestě někoho potká? Jak rychle půjde z práce a jak rychle do ní?

V neděli taky ne!

Přechody

Jednotlivé stavy mezi sebou mohou přecházet jak naprosto přirozeně (postava jde ke stromu, dojde ke stromu a začne pracovat, vpřechod vždy v 8:00, atp.), nebo naopak důsledkem vnějších vlivů (začne pršet, někdo postavu probodne mečem, atp.).

Vytvořením přirozeného cyklu získá postava chování, něco, co bude v naprosté většině do nekonečna opakovat a spoluvytvářet prostředí virtuálního světa.

Přechody

Jednotlivé stavy mezi sebou mohou přecházet jak naprosto přirozeně (postava jde ke stromu, dojde ke stromu a začne pracovat, vpřechod vždy v 8:00, atp.), nebo naopak důsledkem vnějších vlivů (začne pršet, někdo postavu probodne mečem, atp.).

Vytvořením přirozeného cyklu získá postava chování, něco, co bude v naprosté většině do nekonečna opakovat a spoluvytvářet prostředí virtuálního světa.

Nestandardní situace jsou to, co dělá postavy a děj ve světě zajímavým.

Formalizace

Nestandardní situace a tedy plán “reakce na cokoli” sice zajistí aby se postava v průběhu požáru nezačala v klidu vybavovat s hráčem jako by se nechumelilo, ale poskytuje také obrovský prostor k tomu, aby nějaká kombinace akce a stavu unikla, nebo nebyla zpracována správně.

Aparát grafů (počty uzlů, počty hran) dává jisté možnosti, jak kontrolovat validnost výsledných grafů chování.

Pokud všechny stavy prohlásíme za součást množiny S a všechny možné akce za součást množiny A , pak hrany indukující reakci postavy na akci vytváří úplný bipartitní graf.

Formalizace

Aparát grafů (počty uzlů, počty hran) dává jisté možnosti, jak kontrolovat validnost výsledných grafů chování.

Pokud všechny stavy prohlásíme za součást množiny S a všechny možné akce za součást množiny A , pak hrany indukující reakci postavy na akci vytváří úplný bipartitní graf.

V množině uzlů S pak požadujeme existenci nějaké uzavřené, ne nutně eulerovské kružnice délky alespoň dva.

Přechody

Postavu pak charakterizuje to, jaké stavy jí vybereme jako přirozené. Mezi těmi se bude postava pohybovat, pokud nenastene nějaká vnější specifická okolnost (např. někdo ji zapálí).

Přechody tedy mohou být jak důsledky jiných akcí, mohou být nastavené na konkrétní čas, nemusí přitom jít jen o akce směřované přímo na ní, ale i na objekty v okolí.

Přechody

Přechody tedy mohou být jak důsledky jiných akcí, mohou být nastavené na konkrétní čas, nemusí přitom jít jen o akce směřované přímo na ní, ale i na objekty v okolí.



Přechody

Řekneme, že to, co aktuálně postava dělá, co provádí, i to jak zrovna reaguje je **stav**.

To, co nutí postavu změnit svůj stav, až už vnitřní, či vnější, jsou **přechody**.

Konkrétními hodnotami - kdy postava zpanikaří, jestli bude bojovat, nebo uteče atp. se zabývá mechanika (P9).

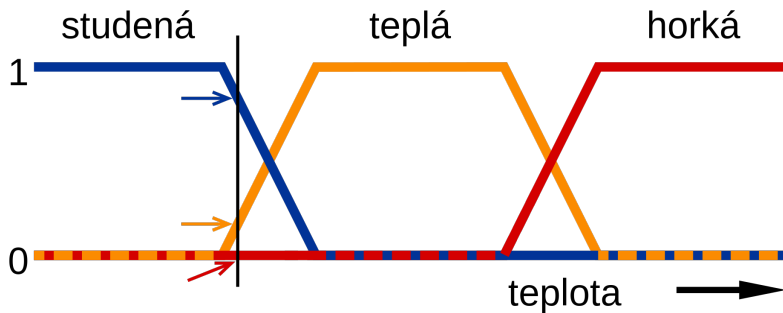
Fuzzy logika - short intro

Pro potřeby her lze s výhodou upotřebit vlastnosti fuzzy logiky [GP96, GP94].

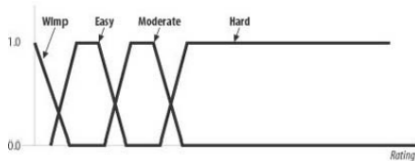
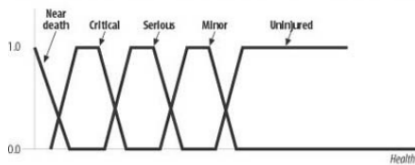
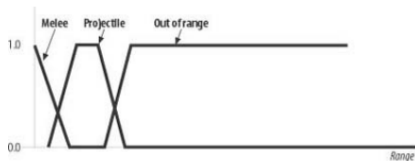
Fuzzy logika je vícehodnotová logika charakterizována tím, že operuje nikoli s jen s konkrétními (binárními) stavy, ale i s přechody mezi nimi.

Přirozeně tak reprezentuje stavy, které nemusí být přesně rozpoznatelné, resp. ohraničené na nějaké ose významnosti. Fuzzy logika tak umožňuje reprezentovat i vágnost přirozené řeči [PI12] a neurčitost, stejně jako pro reálný svět ne vždy přirozené chování při “ostrém” přechodu logických stavů.

Fuzzy logika - short intro



Fuzzy logika - short intro



trapezoidy jsou nejčastější z důvodu jednoduchosti, používají se ale i sigmoidy

Fuzzy logika - short intro

Od roku 1996, kdy se fuzzy logika začala v herním průmyslu používat její použití stoupá, udává se jako jedna z nejobecnějších a nejužitečnějších technik s možným použitím v zásadě v každé hře.

Základní výhoda v v případě rozhodovacího procesu NPC (výběr zbraně, použití zbraně, změna stavu, ...). její zapojení do AI je nejen přirozené, ale zlepšuje dojem z hry [Acampora2010] a chování NPC.

Využití funkcí fuzzy logiky jako responzních křivek pro rozhodovací logiku je přirozené a jednoduše definované.

Fuzzy logika - short intro

AI bez Fuzzy logiky může (díky FSM) jednoduše určit co se má dělat (spatří nepřítele - změní stav na utíkat).

Dodatečné parametry (s jakou pravděpodobností, jak rychle, atp.) je možné spočítat ale jsou mimo základní logiku FSM, tedy jen jako nadstavba.

Fuzzy logika toto řeší přirozeně bez nutnosti systém rozšiřovat o nějakou mechanickou nadstavbu.

Jednou mechanikou se tedy řeší nejen co se má dít (další stav) ale i jestli se to má dít.

Intuitivně

Vstup: množství střeliva

- malo munice: uteč!
- hodne munice: útoč!
- jinak: bojuj defenzívně

Vstup: množství zdraví

- malo zdraví: uteč!
- hodne zdraví: útoč!
- jinak: bojuj defenzívně

Intuitivně

```
if(health < 10)
    if(ammo < 5) return ESCAPE
    if(ammo >= 5 && ammo < 20) return CAUTIOUS
    else return NORMAL
else if(health >= 10 && health < 20)
    if(ammo < 5) return CAUTIOUS
    if(ammo >= 5 && ammo < 20) return NORMAL
    else return ATTACK
else
    if(ammo < 5) return NORMAL
    if(ammo >= 5 && ammo < 20) return ATTACK
    else return FULLATTACK
```

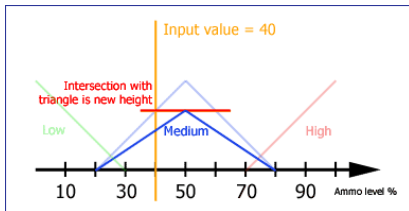
Intiutivně

Co když chceme přidat další proměnou? Co když jich přidáme víc?

- počet nepřatel
- únava
- agresivita
- vhodné / nevhodné prostředí
- atd.

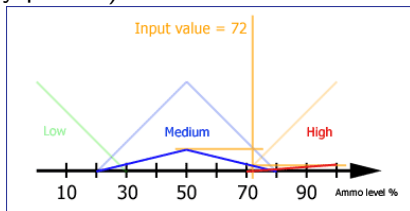
Intuitivne+: input system

zjištění váhy



Intuitivne+: input system

zjištění váhy (jiný příklad)



Intuitivne+

Vstup: množství nepřátel

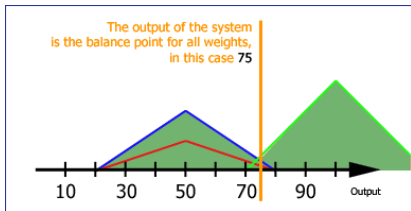
- málo nepřátel: útoč!
- středně nepřátel: braň se!
- hodně nepřátel: uteč!

škála je opačná!

- hodně nepřátel: uteč!
- středně nepřátel: braň se!
- málo nepřátel: útoč!

Intuitivne+: input system

Výstup



Fuzzy FSM - short intro

Hybridem a pokusem využití jak přirozeně definovaného Fuzzy přechodu, tak jednoduše organizovatelného FSM jsou (nečekaně) Fuzzy FSM (FuSM).

Objekt se pak nachází v nějakém stavu FuSM, definice jeho přechodu do jiného stavu pak závisí na responzní křivce.

FSM to FuSM

Převod je možný na dvojí úrovni:

- 1 Fuzzy přechod: Namapování přechodů mezi stavy pomocí responzní křivky
- 2 Fuzzy stav: FuSM se nachází ve více stavech současně s určitou mírou věrohodnosti

První případ je triviální pro samotnou rozhodovací mechaniku a triviálně převoditelný. Druhý případ je méně častější, přesto dobře použitelný pro abstraktnější stavy

link: [fuzzylib](#)

Fuzzy logika - short intro

Nevýhody

- postavený na znalostní bázi
- časté kontroly stavu a přechodu
- náročnější pro větší počet agentů než obyčejné FSM

Konec

Dotazy?



Janos L Grantner and Marek J Patyra.

Fuzzy logic finite state machine models for real time systems. In [NAFIPS/IFIS/NASA'94. Proceedings of the First International Joint Conference of The North American Fuzzy Information Processing Society Biannual Conference. The Industrial Fuzzy Control and Intellige](#), pages 296–300. IEEE, 1994.



Janos L Grantner and MJ Patyra.

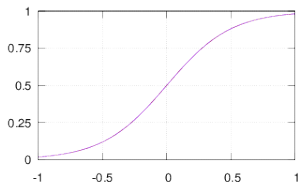
Reprogrammable fuzzy logic finite state machine model. In [Proceedings of North American Fuzzy Information Processing](#), pages 492–496. IEEE, 1996.

Bibliography

Sigmoida

Sigmoida je funkce s typickým S-tvarem. Asi nejčastějším předpisem je:

$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$



Může ale jít i o hyperbolický tangens, Gudermannovi funkci, arkustangens atp.