



Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Simulated Car Racing Competition 2011

Dokumentácia k projektu

READY
2win




Torcs Championship

Tím číslo:	20	Bc. Maroš Bednár
Vedúci projektu:	Ing. Peter Vilhan	Bc. Adam Brček
Predmet:	Tvorba softvérového systému v tíme	Bc. Marek Briš
Študijný program:	Informačné systémy/Softvérové inžinierstvo	Bc. Marián Florek
Akademický rok:	2010/2011	Bc. Vojtech Juhász
		Bc. Ivan Valenčík
		Bc. Juraj Kosmeľ

Obsah

0. Úvod	2
0.1 Účel a rozsah dokumentu	2
0.2 Prehľad dokumentu	2
0.3 Použité skratky a pojmy	2
0.4 Použitá notácia	3
1. Analýza	4
1.1 TORCS – The Open Racing Car Simulator	4
The Simulated Car Racing Championship	5
Demolition Derby	8
1.2 Existujúce riešenia.....	8
LUIGI CARDAMONE	8
AUTOPIA	9
COBOSTAR.....	10
2. Špecifikácia	12
2.1 Ciele projektu	12
2.2 Špecifikácia požiadaviek	13
2.3 Identifikácia prípadov použitia a hráčov	14
3. Návrh	16
3.1 Základná koncepcia architektúry	16
3.2 Návrh modulov	17
4. Použitá literatúra	19

0. Úvod

0.1 Účel a rozsah dokumentu

Predkladaný dokument obsahuje dokumentáciu k tímovému projektu *Simulation racing car*, ktorý riešime v rámci predmetu Tímový projekt. Forma dokumentu zodpovedá pravidlám a požiadavkám spomenutého predmetu. Obsahuje analýzu, špecifikáciu a návrh riešenia, neskôr pribudne aj opis implementácie, testovania a používateľská príručka. Finálna verzia dokumentu bude mať predpokladaný rozsah niekoľko desiatok strán.

0.2 Prehľad dokumentu

Dokumentácia je rozdelená do viacerých, už vyššie spomenutých, kapitol. Prvá kapitola obsahuje analýzu problémovej oblasti, opis súťaže, podmienok účasti ako aj jazdných vlastností a možností auta.

V časti špecifikácia definujeme ciele, ktoré sa budeme snažiť splniť a tiež sa už venujeme konkrétnym požiadavkám kladeným na vytváraného autopilota.

Návrh obsahuje konkrétne navrhované riešenie a identifikáciu jednotlivých častí a modulov autopilota spolu s popisom.

Spomenuté časti sú prvým výsledkom našej práce na projekte a budú postupne doplnené, najmä čo sa návrhu týka o detailné popisy a riešenia. Taktiež pribudne popis samotnej implementácie, testovania, ako aj používateľská príručka k vytvorenému prototypu.

0.3 Použité skratky a pojmy

TORCS – The open racing car simulator

autopilot (bot) – inteligentný agent, ktorý v prostredí TORCS riadi vozidlo

tick – doba čakania servera na odozvu

0.4 Použitá notácia



Autopilot

hráč



UC2 Predbiehanie oponenta

prípád použitia



asociácia



Rýchla obnova auta mimo trate

cieľ

1. Analýza

Kapitola obsahuje analýzu problémovej oblasti, v našom prípade svet TORCS a analyzuje existujúce riešenia autopilotov fungujúcich v tomto prostredí.

1.1 TORCS – The Open Racing Car Simulator

TORCS [4] je open source 3D simulátor automobilových pretekov dostupný na platformách Linux, FreeBSD, Mac OS X a Windows. Pôvodne ho vytvorila dvojica autorov Eric Espié a Christophe Guionneau, v súčasnosti projekt vyvíja Bernhard Wymann. TORCS je napísaný v programovacom jazyku C++ a umožňuje zavodiť predprogramovaným agentom, *botom*, ako aj hráčom za pomoci klávesnice či myši.

TORCS je považovaný za jednu z najlepších open source alternatív ku komerčným závodným hrám z viacerých dôvodov [3]:

- Je založený na prepracovanom fyzikálnom modeli, ktorý berie do úvahy množstvo aspektov, ako napríklad kolízie, trakciu, aerodynamiku, spotrebu paliva a ďalšie.
- Vizualizácia je založená na sofistikovanom 3D modeli.
- Hráči majú k dispozícii vyše 40 tratí rozdelených do troch kategórií – klasické cestné preteky (road), ovály (oval) a hrbolaté nespevnené trate na štýl plochej dráhy (dirt).

Hlavným využitím je však vývoj vlastných botov, prípadne úprava existujúcich vyše 50 protivníkov kvôli dosiahnutiu čo najlepších výsledkov. Títo agenti sa zostavujú ako samostatné C++ moduly, ktoré je možné ľahko skompilovať a pridať do hry. V každom kontrolnom kroku (*tick*) má bot prístup k aktuálnemu stavu hry, ktorý obsahuje informácie o aute a trati, ako aj o protivníkoch. Bot kontroluje auto pomocou plynového pedálu a brzdy, radiacej páky a volantu.

Platforma TORCS sa využíva na usporadúvanie viacerých medzinárodných súťaží, z ktorých najvýznamnejšie sú Majstrovstvá jazdcov a podujatia organizované platformou The TORCS Racing Board. Novšími súťažami sú The Simulated Car Racing Championship a Demolition Derby – tieto využívajú odlišný prístup k informáciám o trati než je tomu v pôvodnom simulátore.

The Simulated Car Racing Championship

Súťaž je založená na platforme TORCS, avšak je potrebné doinštalovať softvér, ktorý pôvodnú architektúru rozširuje v troch bodoch [3]:

- Architektúra sa mení na klient-server aplikáciu: boti sú spúšťaní ako externé procesy prepojené s pretekovým serverom pomocou UDP spojení.
- Do hry je zapracovaný reálny čas: v každom hernom ticku (približne 20 ms simulovaného času) server odošle sensorové vstupy každému botovi a čaká 10 ms reálneho času na prijatie akcie od agentov. Ak nie je prijatá žiadna akcia, simulácia pokračuje ďalej a použije sa naposledy prijatá akcia.
- Softvér vytvára fyzické oddelenie medzi kódom jazdca a pretekovým serverom pomocou vytvorenia abstraktnej vrstvy sensorov a efektorov – vďaka nej sa obmedzuje prístup k informáciám o trati dostupných v pôvodnej hre. Dostupné senzory a efekторы (použité v ročníku 2010) sú uvedené v tabuľkách 01 a 02.

Tabuľka číslo 01: Popis dostupných efektorov 2010.

Názov	Rozsah	Popis
accel	[0; 1]	Virtuálny plynový pedál
brake	[0; 1]	Virtuálny brzdový pedál
clutch	[0; 1]	Virtuálny spojkový pedál
gear	-1,0,1,...,7	Rýchlostný stupeň
steering	[-1; 1]	Hodnota zatočenia: -1 a 1 znamenajú naplno doprava resp. doľava, čo zodpovedá uhlu 0,785398 rad.
focus	[-90; 90]	Smer <i>focus</i> diaľkometerov v stupňoch.
meta	0; 1	Príkaz meta-riadenia: 0 predstavuje žiadnu zmenu; 1 žiadosť pre súťažný server o reštart preteku.

Tabuľka číslo 02: Popis dostupných senzorov.

Názov	Rozsah (jednotky)	Popis
angle	$[-\pi; \pi]$ (rad)	Uhol medzi orientáciou auta a orientáciou osi trate.
curLapTime	$[0; \infty]$ (s)	Čas odjazdený počas aktuálneho kola.
damage	$[0; \infty]$ (body)	Aktuálne poškodenie auta (čím vyššia hodnota, tým väčšie poškodenie).
distFromStart	$[0; \infty]$ (m)	Vzdialenosť auta od štartovej čiary.
distRaced	$[0; \infty]$ (m)	Vzdialenosť prejdená od začiatku preteku.
focus	$[0; 200]$ (m)	Vektor 5 diaľkometerov; každý vracia vzdialenosť medzi okrajom trate a autom v rozsahu 200 metrov. Sensory vzorkujú 5stupňový priestor pozdĺž smeru poskytnutého od klienta (akciou <i>focus</i>). Sensory môžu byť použité len raz za sekundu simulovaného času, v opačnom prípade poskytnú hodnotu mimo rozsahu (platí aj v prípade, že auto je mimo trate).
fuel	$[0; \infty]$ (l)	Aktuálne množstvo paliva.
gear	$\{-1, 0, 1, \dots, 7\}$	Aktuálny prevod (-1 je spiatka, 0 neutrál).
lastLapTime	$[0; \infty]$ (s)	Čas prejdenia posledného kola.
opponents	$[0; 200]$ (m)	Vektor 36 senzorov zameraných na súperov: každý pokrýva úsek 10 stupňov v rozsahu 200 metrov a vracia vzdialenosť najbližšieho oponenta v pokrytej oblasti. Sensory pokrývajú celý priestor okolo auta, v smere hodinových ručičiek od π po $-\pi$ vzhľadom k osi auta.
racePos	$\{1, 2, \dots, N\}$	Pozícia v preteku vzhľadom na protivníkov.
rpm	$[2000; 7000]$ (ot/min)	Počet otáčok za minútu motora.
speedX	$[-\infty; \infty]$ (km/h)	Rýchlosť auta pozdĺž pozdĺžnej osi auta.
speedY	$[-\infty; \infty]$ (km/h)	Rýchlosť auta pozdĺž priečnej osi auta.
speedZ	$[-\infty; \infty]$ (km/h)	Rýchlosť auta pozdĺž osi z auta.
track	$[0; 200]$ (m)	Vektor 19 diaľkometerov; každý vracia vzdialenosť medzi okrajom trate a autom v rozsahu 200

		metrov. Sensory pokrývajú priestor pred autom, v smere hodinových ručičiek od $\pi/2$ po $-\pi/2$ vzhľadom k osi auta. Ak sa auto nachádza mimo trate, návratová hodnota je mimo rozsahu.
trackPos	$(-\infty; \infty)$	Vzdialenosť medzi autom a osou trate. Hodnota je normalizovaná vzhľadom na šírku trate: 0 predstavuje auto na osi trate, -1 v prípade, že auto je na pravom okraji a 1 v prípade, že auto je na ľavom okraji trate.
wheelSpinVel	$[0; \infty]$ (rad/h)	Vektor 4 senzorov reprezentujúci rýchlosť otáčok kolies.
z	$[-\infty; \infty]$ (m)	Vzdialenosť ťažiska auta od povrchu trate pozdĺž osi z.

Cieľom majstrovstiev je zostrojiť bota pre závodné auto, ktorý bude súťažiť na niekoľkých neznámych tratiach najprv sám (na čas), a potom proti ostatným súperom. Boti vnímajú závodné prostredie radom senzorov a uskutočňujú typické jazdné akcie pomocou efektorov.

Majstrovstvá sa skladajú z deviatich pretekov uskutočňovaných počas troch súťaží (na každej sa konajú tri preteky) a v tomto formáte fungujú od roku 2009 (prvý pretek založený na rovnakom softvéri bol o rok skôr). Každý rok dochádza k jemným úpravám týkajúcich sa efektorov (pridanie spojky) či senzorov, napríklad zmena ich rozsahu. Významnou črtou je tiež zavedenie šumových senzorov – jedná sa o senzory focus, opponents a track z tabuľky 02. Podstatou je úprava návratovej hodnoty senzorov o takzvaný normálny šum, čiže o štandardnú odchýlku rovnajúcu sa istému percentu z nameraných hodnôt, pričom sa toto percento medzi jednotlivými ročníkmi majstrovstiev mení.

Každý pretek sa skladá z troch častí [2]:

- *Tréning (Warm-up)* – každý pilot jazdí sám; táto fáza je určená na zber dôležitých informácií o trati a na vylepšenie správania pilota.
- *Kvalifikácia (Qualifying)* – každý pilot jazdí sám na všetkých troch pretekoch súťaže; osem pilotov, ktorí prejdú najväčšiu vzdialenosť, sa kvalifikuje do závodu.
- *Samotný závod (Race)* – osem najlepších pilotov jazdí spolu. Preteky sa skladajú z ôsmich behov na každej z troch tratí – v každom behu štartuje pilot z iného miesta. Na konci preteku sú pilotom pridelené body systémom 10-8-6-5-4-3-2-1. Dodatočné

dva body získa pilot, ktorý dosiahol najrýchlejšie kolo ako aj pilot, ktorý absolvoval preteky s najmenším poškodením auta.

Demolition Derby

Súťaž [1] je svojou architektúrou podobná The Simulated Car Racing Championship, rovnako je nutné doinštalovať softvér ovplyvňujúci pôvodný TORCS. Prvý ročník sa mal uskutočniť v roku 2010, z dôvodu nedostatku účastníkov však bol zrušený.

Cieľom je zostrojiť bota pre závodné auto, ktorý dokáže efektívne narážať do protivníkov, pričom sa bude snažiť minimalizovať vlastné poškodenie, podstatou je teda zničenie všetkých súperov, víťazom je posledné stojace auto. Demolition Derby sa odohráva na veľkej kruhovej trati (dĺžka 640m, šírka 90m, povrchom je asphalt). Sensory a efekty sú podobné ako pri súťaži The Simulated Car Racing Championship (tabuľky 01 a 02), odstránené sú efekty *speedZ* a z.

Model poškodenia je navrhnutý tak, že cieľom je naraziť do ostatných áut z boku alebo zozadu. Čelné nárazy, rovnako ako nárazy do steny nevedú k poškodeniu.

Všetci prihlásení piloti sa musia kvalifikovať do záverečného boja súťažiením s každým protivníkom v kvalifikačných 1vs1 zápasoch. Po každom zápase, ktorý trvá maximálne 5 minút simulovaného času, auto s menším poškodením získava bod. Osem najlepších pilotov potom súťaží proti sebe v rovnakom čase v desiatich finálových zápasoch. V týchto súbojoch sa všetkým autám vynuluje poškodenie zakaždým, keď je nejaký účastník vyradený z boja. Najčastejší víťaz sa stane šampiónom Demolition Derby.

1.2 Existujúce riešenia

V nasledujúcej časti sú opísané 3 konkrétne riešenia iných tímov, ktoré dosiahli prvé miesta v rokoch 2008, 2009 a 2010. V každom je opísaný algoritmus, klady a zápory.

LUIGI CARDAMONE

Víťaz z roku 2008. Hlavnou ideou tohto riešenia bolo vyvinúť konkurencieschopného pilota, ktorý by využíval čo najmenej vstupov. Architektúra pozostáva z vyvíjajúcich sa neurónových sietí implementujúcich základné jazdné správanie, spojené s hotovým správaním pre štart, obnovu po kolízii, preradzovaním rýchlostí a predbiehaním protivníkov.

Vstupom do neurónovej siete boli informácie priamo súvisiace s riadením, čiže rýchlosť a senzory zachytávajúce okraje trate. Ako výstup bola zvolená dvojica natočenie kolies a pridávanie/brzdzenie. Ak sa pilot nachádzal na rovnej dráhe, výstupy pre brzdzenie

a pridávanie boli kompletne ignorované a auto akcelerovalo na plný plyn. Tento návrh nútil pilota jazdiť čo najrýchlejšia a zabraňoval plytvaniu času na bezpečné ale pomalé jazdenie.

Rozhodujúcou schopnosťou pilota je obiehajúce protivníkov. V prípade tohto riešenia bolo explicitne implementované tak, že upravovalo výstupy neurónovej siete dvoma spôsobmi. Ak sa pilot nachádzal na rovnej dráhe, vždy sa pokúsil o predbiehač manéver. Ak sa však blížila zákruta, brzdil tak, aby nedošlo ku kolízii.

Pre radenie rýchlostí a obnovovanie sa z kolízií využili predprogramované techniky z príkladu pilota voľne dostupného spolu s TORCS. Štart preteku označili ako základnú a veľmi podstatnú časť preteku z dvoch dôvodov:

1. dochádza tam často ku zrážkam a auto sa môže značne poškodiť
2. je možné obehnúť väčšie množstvo áut naraz čo prispeje k lepšiemu konečnému výsledku

Vo svojom riešení predstavili veľmi jednoduchú štartovaciu techniku, založenú na rýchlom presunutí sa na okraj trate a predbehnutí čo najviac protivníkov hneď na začiatku.

Na vývoj neurónovej siete využili techniku NEAT, založenú na veľmi podobnom princípe ako evolučné algoritmy. Každú neurónovú sieť testovali jedno kolo na jednej konkrétnej trati z menom Wheel 1, ktorá obsahuje väčšinu zaujímavých vlastností z tratí, ktoré TORCS obsahuje. Na výpočet vhodnosti pre jednotlivé neurónové siete využívali tri hodnoty a to prejdená dráha, priemerná rýchlosť a počet tikov servera kedy sa pilot nachádzal mimo trate.

Medzi hlavné nedostatky riešenia patrí neschopnosť pilota ísť aspoň trochu podľa ideálnej stopy, vzhľadom k tomu, že väčšinu trate jazdí stredom. Nie je implementovaný žiadny druh učenia sa počas pretekov vzhľadom k tomu, že sa využíva vyvinutá neurónová sieť. Medzi kladné stránky patrí množina vedomostí o trati, ktorá je minimálna a vysoký stupeň zovšeobecnenia, čo znamená vytvorenie jednej neurónovej siete pre všetky trate a to iba na základe jedného kola na jednej trati.

AUTOPIA

Tím Autopia bol víťazom posledného ročníka súťaže typu championship v roku 2010. Líder tohto tímu, Enrique Onieva, sa zúčastnil viacerých predchádzajúcich ročníkov a zaoberá sa modulárnou parametrizovateľnou architektúrou. V tomto riešení rozdelili pilota na viac modulov, ktoré riešia jednoduchšie oblasti riadenia/problémy. Presnejšie jedná sa o šesť nasledovných modulov:

- kontrola radenia

- maximálna rýchlosť - modul určujúci maximálnu rýchlosť na segmente
- kontrola rýchlosti - modul kontrolujúci brzdu a plyn; slúži na upravenie rýchlosti, aby vyhovovala modulu maximálnej rýchlosti; okrem toho implementuje ABS a TCS
- kontrola riadenia - ovláda natočenie kolies
- modul učenia – deteguje segmenty trate, kde by sa mala rýchlosť upravovať, vo väčšine prípadov ide o rovinky alebo o miesta pred ostrými zákrutami
- modul protivníkov - aplikuje zmeny v riadení, aby vyhovovali situáciám, keď sú oponenti blízko

Na určenie rýchlosti využívajú fuzzy logiku, kde implementovali sedem jednoduchých pravidiel. Takýto prístup im umožňuje rýchlu odozvu a možnosť venovať sa ostatným zložitejším úlohám riadenia. Pravidlá vo fuzzy logike vylepšovali genetickými algoritmi na viacerých tratiach v rámci TORCS.

Modul protivníkov označili ako kritický, pretože stratégia protivníkov je neznáma a je ťažké sa rozhodnúť či predbiehať alebo brzdiť. Ako výsledok hĺbkovej analýzy implementovali jednoduché pravidlá ako je náhle zatočenie alebo znižovanie rýchlosti tesne pred zrážkou.

Výhody sú v jednoduchosti a parametrizovateľnosti riešenia, čo umožňuje ľahké rozšírenie. Modul protivníkov zabezpečuje obiehajúce viacerých protivníkov s utržením čo najnižšieho poškodenia auta a umožňuje rozšíriteľnosť, čo sa týka zabraňovaniu v obíhaní inými protivníkmi.

COBOSTAR

Ako prvý skočil v roku 2009. Pilot je kombinácia techniky, ktorá vstup v podobe senzorov premieňa na akciu, a princípu úpravy správania sa, ktorý vylepšuje aktuálnu akciu za pomoci nejakého odhadu alebo vedomosti o budúcnosti alebo minulosti. Základné riadenie zahrňuje iba vzdialenosť a uhol najdlhšieho senzora k okraju trate. Pre riadenie mimo trate sa uvažuje uhol k osi trate a vzdialenosť k trati. Táto technika bola optimalizovaná a rozšírená o ABS, ASR, recovery modul, kontrolu skoku, detektor nárazov a vyhýbanie sa protivníkom.

Optimalizácia sa vykonávala nad zoskupením 16 parametrov, kde jednotlivé skupiny testovali na všetkých tratiach v rámci TORCS. Vhodnosť skupiny parametrov určovali na základe vzdialenosti, ktorú prešiel počas 10000 tikov servera. Nakoniec bola zvolená skupina ktorá prešla najväčšiu vzdialenosť vzhľadom ku každej trati, týmto spôsobom vylúčili lokálne najlepšiu skupinu a zvolili si všeobecne najvýhodnejšiu skupinu parametrov.

Učenie počas jazdenia bolo založené na závažnosti nehody a krokoch pred nehodou, čo mohlo viesť k zníženiu rýchlosti alebo upraveniu správania v ďalšom kole. Implementovali monitorovania protivníkov takým spôsobom, že do senzorov merajúcich vzdialenosť ku krajom trate premietli miesto a čas za aký sa autá zrazia a následne sa riadili podľa základnej techniky. Takto vytvorili efektívne a robustné správanie zohľadňujúce správanie protivníkov.

Pozitíva tohto riešenia sú v jednoduchom riadení a prepojení riadenia s vyhýbaním sa kolíziám s protivníkmi. Avšak toto jednoduché riadenie, ktoré berie do úvahy iba senzory merajúce vzdialenosť od okraja trate, nie je optimálne a nepočíta so žiadnymi inými údajmi o trati.

2. Špecifikácia

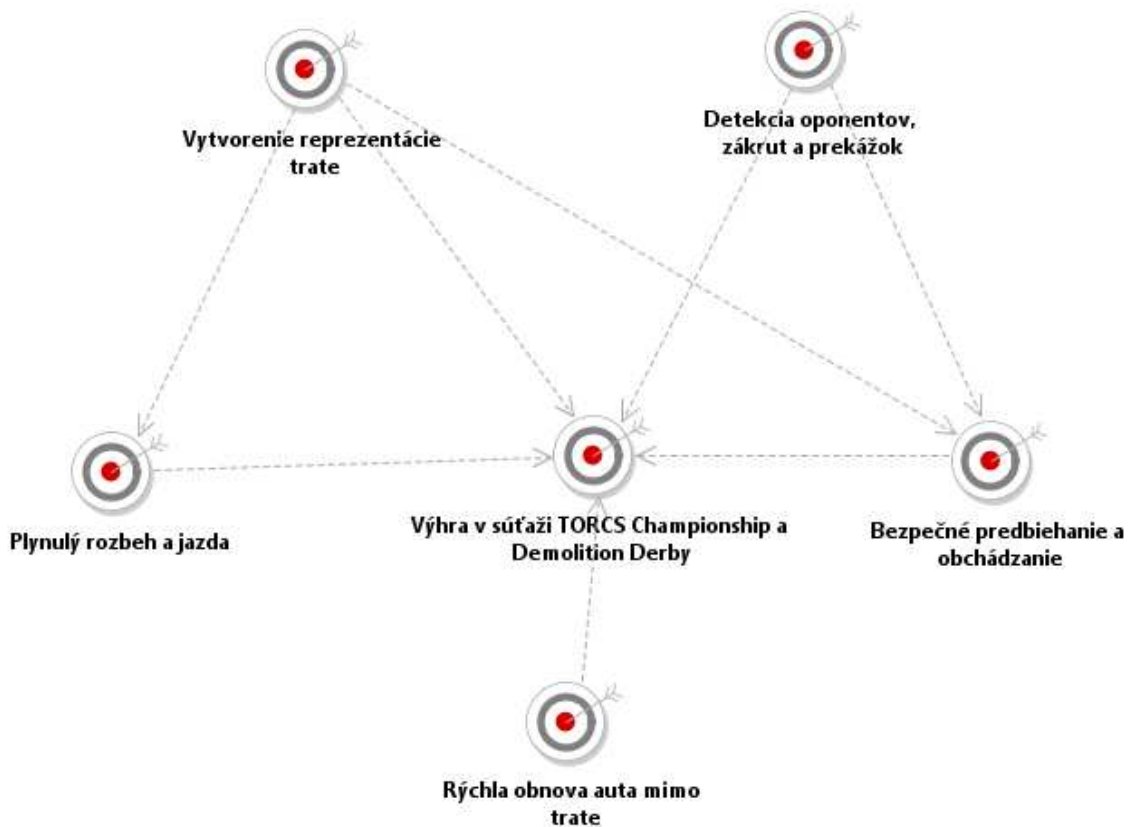
V tejto kapitole sa bližšie pozrieme na požiadavky kladené na nami vytváraného autopilota ako aj ciele, ktoré náš projekt sleduje, a ktoré sa snažíme naplniť. Taktiež priblížime správanie autopilota, ktoré bude ilustrované diagramom prípadov použitia.

2.1 Ciele projektu

V nasledujúcich riadkoch uvádzame sumarizáciu cieľov, ktoré sme si stanovili. Identifikované ciele sú ilustrované diagramom cieľov na Obr. 01.

Identifikované ciele:

- Výhra v súťaži TORCS Championship a Demolition Derby
 - Vytvorenie reprezentácie trate - je potrebné vytvoriť matematickú reprezentáciu trate, aby sme boli schopní určiť čo najlepšiu stopu, ktorou náš autopilot pôjde. Zároveň táto reprezentácia posluží aj na konfiguráciu rôznych nastavení jazdy, ako rýchlosť, miesta brzdzenia, či častých a možných kolízií so súpermi.
 - Detekcia oponentov, zákrut a prekážok - je potrebné detegovať počas jazdy priestor, v ktorom sa auto nachádza, z dôvodu predvídania a prispôsobenia jazdy resp. prípravy na obiehanie.
 - Bezpečné predbiehanie a obchádzanie - splnenie tohto cieľa významne pomôže naplneniu celkového cieľa, ktorým je cieľ vyhrať. Prípadná kolízia pri predbiehaní alebo obchádzaní spôsobí stratu času, ktorá môže mať za následok celkový neúspech v súťaži.
 - Rýchla obnova auta mimo trate - taktiež dôležitý cieľ, keďže v súťaži ide o čas a čím rýchlejšie sa dokážeme dostať späť na trať po vybočení z nej, tým sú naše šance na úspech väčšie.
 - Plynulý rozbeh a jazda - plynulosť jazdy je určite dôležitým krokom k víťazstvu a k naplneniu hlavného cieľa.



Obr. 01 Identifikované ciele

2.2 Špecifikácia požiadaviek

Systém bude založený a testovaný na platforme TORCS, ktorá je bližšie špecifikovaná v časti *Analýza*. Našou úlohou je implementovať autopilota, ktorý bude schopný jazdiť na tratiach tejto platformy, a to v dvoch súťažiach:

- Championship
- Demolition Derby,

ktoré sú taktiež detailne popísané v analýze. Našou hlavnou úlohou bude navrhnuť autopilota, ktorý prejde určenú trať čo najrýchlejšie, respektíve s čo najmenším poškodením. Musíme sa zamerať na viaceré oblasti, čo sa jazdy a jazdných vlastností vozidla týka. Dôležité je zvoliť správnu metodiku jazdy, pamätať si trať alebo reagovať na aktuálny stav auta na trati. Trate sú vytvorené na rôznych povrchoch, pričom na každom sú jazdné vlastnosti auta iné a preto je potrebné prispôbiť spôsob jazdy vzhľadom na aktuálny povrch. Na naučenie trate a zistenie všetkých jej vlastností máme k dispozícii testovacie kolá, tzv. warm up, počas ktorých

prechádzame traťou a môžeme zbierať informácie potrebné pre vytvorenie ideálnej stopy a konfiguráciu ostatných parametrov tak, aby sme dosiahli čo najlepší čas na danej trati.

Dôležitou oblasťou je obnova auta po zidení z trate. Tu je potrebné definovať všetky možnosti, ktoré môžu nastať a taktiež spôsoby riešenia daných stavov. Ďalšou oblasťou, na ktorú sa musíme zamerať, je predbiehanie oponentov, kde je potrebné monitorovať aktuálnu polohu oponentov, našu rýchlosť ako aj možnosť predbiehania vzhľadom na trať.

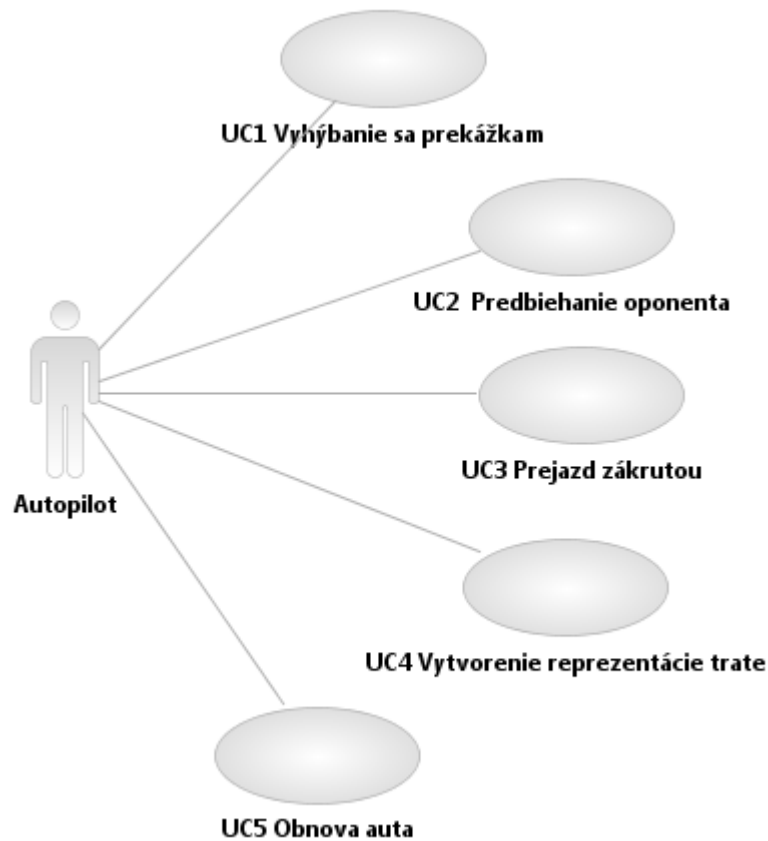
V súťaži Demolition Derby je potrebné zamerať sa najmä na spôsob ako čo najdlhšie udržať auto nepoškodené a zároveň zničiť súperov. Musíme riešiť možné spôsoby vyhýbania a narážania. V súťaži Demolition Derby je potrebné zamerať sa najmä na spôsob ako čo najdlhšie udržať auto nepoškodené a zároveň zničiť súperov čelnými nárazmi. Za týmto účelom treba vymyslieť techniky jazdy, ktoré autu umožnia vyhýbať sa útokom súperov a dajú mu schopnosť rozoznať vhodné príležitosti na vykonanie narazenia do súpera. Auto pri tom musí byť schopné zvládať rôzne počty súperov, keďže kvalifikácia sa odohráva systémom jeden proti jednému a vo finále proti sebe bojuje 8 áut naraz, pričom sú postupne vyradované.

2.3 Identifikácia prípadov použitia a hráčov

V súvislosti s naším systémom nevystupujú žiadni hráči, ak pravdaže nerátame používateľa, ktorý spustí zdrojové kódy autopilota na serveri TORCS. Jediným zmysluplným hráčom je samotný autopilot, ktorý vykonáva rôzne aktivity v súvislosti s prechádzaním trate. Pre vyššie spomenutého hráča sme identifikovali tieto prípady použitia (Obr. 02):

- *UC1 Vyhýbanie sa prekážkam* - autopilot po identifikácii prekážky na trati vyhodnotí situáciu a zvolí vhodný spôsob obídenia danej prekážky v prípade potreby aj zastavenie vozidla
- *UC2 Predbiehanie oponenta* - autopilot deteguje oponenta, vyhodnotí možnosť predbiehania vzhľadom na možnosti trate a rýchlosť, určí ideálnu stopu predbiehania a predbehne daného oponenta
- *UC3 Prejazd zákrutou* - autopilot identifikuje zákrutu, vzhľadom na zapamätanú trať zvolí vhodný spôsob prejdenia zákruty - miesto vjazdu a výjazdu, rýchlosť, natočenie kolies a pod.

- UC4 *Vytvorenie reprezentácie trate* - autopilot prechádza trať viackrát za sebou a zisťuje všetky potrebné parametre - vytvára reprezentáciu trate pre potreby následnej súťaže a identifikáciu čo najideálnejšej stopy
- UC5 *Obnova auta* - autopilot po vyjdení z trate určí vzhľadom na svoju polohu a smer vhodný spôsob obnovy a vracia sa späť na trať



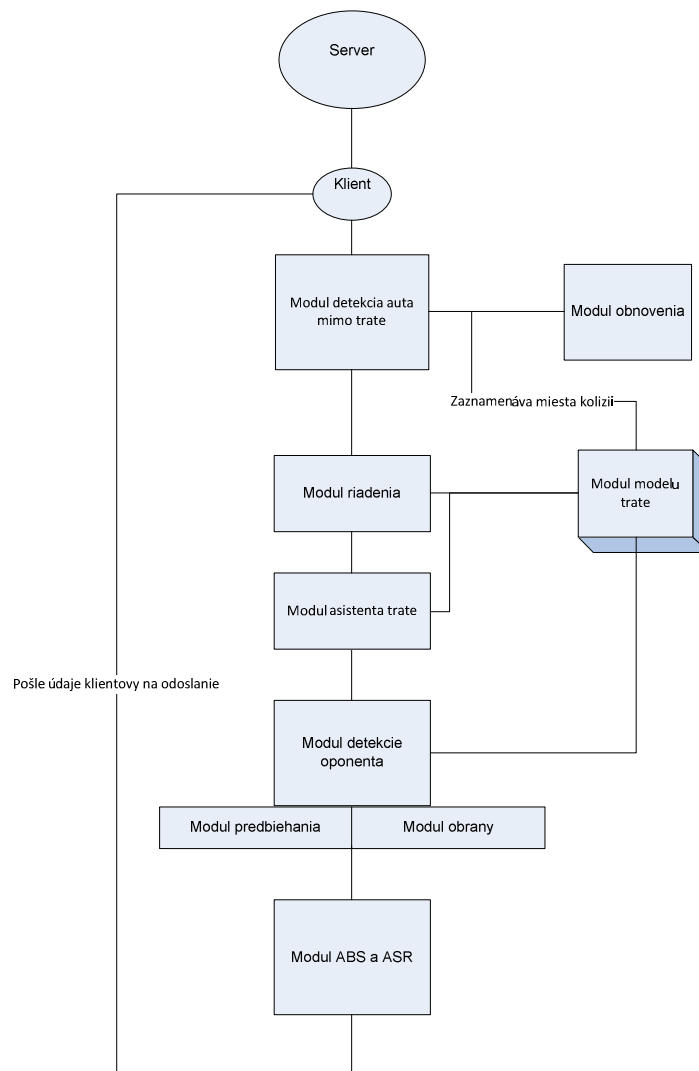
Obr. 02 Prípady použitia

3. Návrh

Naším cieľom je vytvoriť autopilota, ktorý bude schopný dosahovať najlepšie možné výsledky pri nasadení aj do vopred neznámej trate.

3.1 Základná koncepcia architektúry

Keďže našou úlohou je zúčastniť sa v oboch súťažiach, je potrebné navrhnuť architektúru autopilota v čo najväčšej miere prenosnosti. Takto navrhnutý autopilot bude ľahko zostrojiteľný pre akúkoľvek súťaž, pretože každá jedna potrebuje základné moduly (napríklad modul pre ovládanie plynu a brzdy, modul riadenia, modul radenia a pod.). Po viacerých návrhoch architektúr sme sa rozhodli práve pre architektúru zobrazenú na Obr. 03.



Obr.03 Základný návrh architektúry autopilota

3.2 Návrh modulov

Podkapitola obsahuje opis jednotlivých modulov zo základnej koncepcie architektúry autopilota. Nevyhnutné moduly pre chod autopilota:

- modul klienta
- modul detekcie auta mimo trate
- modul obnovenie
- modul riadenie

Modul klienta

Klient slúži na komunikáciu so serverom. Prijíma údaje zo senzorov, s ktorými jednotlivé moduly pracujú a menia ich. Na základe týchto údajov klient spätne posiela na server akcie, ktoré je potrebné vykonať. Komunikácia prebieha v tzv. „Tick-och“ čo je približne 20 milisekúnd.

Modul detekcie auta mimo trate

Úlohou modulu je detegovanie automobilu, či sa nachádza mimo jazdnej dráhy trate.

Modul obnovenia (Recovery)

Tento modul slúži na návrat autopilota z mimo jazdnej dráhy v čo najkratšom čase. V prvom kroku modul zistí, v ktorej časti mimo trate sa nachádza. Následne zistí, v akom smere sa auto nachádza vzhľadom k trati. Na základe týchto informácií prispôsobí nastavenia hodnôt efektorov.

Modul riadenia (Driving)

Modul pre riadenie má za úlohu prispôbovať rýchlosť auta a smer jazdy. Rýchlosť jazdy je ovplyvnená tromi efektormi a to plyn, brzda a prevodový stupeň. Pri zmene prevodového stupňa je za určitých okolností vhodné aktivovať aj efektor pre spojku. Smer jazdy je možné meniť podľa nastavenia efektora pre natočenie volantu.

Modul modelu trate (Lap model)

Na dosiahnutie čo možno najlepších výsledkov je potrebné vedieť ako trať vyzerá. Na to slúži práve tento modul, ktorý zaznamenáva prejdenú trasu cesty spoločne s časom. Každým kolom sa porovnáva čas, za ktorý bola trať prejdená. Ak bol čas kratší ako čas nachádzajúci sa v modeli, tak sa model aktualizuje. Model bude obsahovať aj miesta, v ktorých prišlo ku kolíziám.

Modul asistenta trate (Line Assistant)

Na dosiahnutie maximálnej možnej rýchlosti na trati dohliada modul asistenta trate. Úlohou tohto modulu je určenie ideálnej stopy na trati a následná korekcia na ideálnu stopu ak sa v nej nenachádza.

Modul detekcie oponenta

Úlohou moduluje je detekcia dobiehajúceho alebo približujúceho sa oponenta. Na základe informácií v modeli trate rozhodne, či má autopilot zvoliť modul predbiehania alebo obrany.

Modul predbiehania (Overtake)

Tento modul zaisťuje predbiehanie oponenta v prípade, že pri predbiehaní alebo počas predbiehania autopilot nevyletí mimo trať.

Modul obrany (Defense)

Podobnú úlohu ako modul predbiehania má modul obrany. Rozdiel je v tom, že autopilot sa snaží ubrániť si pozíciu od oponenta ale len za cenu nevyletenia z dráhy alebo osobného kontaktu.

Modul ABS a ASR

Modul slúži na reguláciu stlačenia brzdy v prípade ABS, aby nevznikali súvislé brzdne dráhy, na ktorých je auto neovládateľné. Pri ASR sa reguluje stlačenie plynu na úroveň, aby nedochádzalo k preklzavaniu kolies, čo taktiež vedie k neovládateľnému autu.

4. Použitá literatúra

- [1] Butz, M.V., Linhardt, M.J.: Demolition Derby
<http://www.coboslab.psychologie.uniwuerzburg.de/competitions/>
- [2] Loiacono, D., Cardamone, L.: Simulated Car Racing Championship
<http://cig.ws.dei.polimi.it/>
- [3] Loiacono, D. et al.: The 2009 Simulated Car Racing Championship. In: *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 2, 2010, 131-147.
- [4] Wymann, B., Espié, E.: TORCS
<http://torcs.sourceforge.net>