



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی برق

پروژه کارشناسی
گرایش مخابرات

ماشین های خودران -
با استفاده از یادگیری تقویتی

نگارش
محمد رضیئی فیجانی

استادان راهنما
دکتر وحید پوراحمدی و دکتر حمیدرضا امین داور

شهریور ۱۳۹۸

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

صفحه فرم ارزیابی و تصویب پایان نامه - فرم تأیید اعضاء کمیته دفاع

در این صفحه فرم دفاع یا تأیید و تصویب پایان نامه موسوم به فرم کمیته دفاع - موجود در پرونده آموزشی - را قرار دهید.

نکات مهم:

- نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو(دورو)** بلامانع است و انجام آن توصیه می شود.

به نام خدا

تاریخ: شهریور ۱۳۹۸

تعهدنامه اصالت اثر



اینجانب **محمد رضیئی فیجانی** متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

محمد رضیئی فیجانی

امضا

نویسنده پایان نامه، در صورت تمایل میتواند برای پاسخگویی پایان نامه خود را به شخص یا اشخاص و یا ارگان خاصی تقدیم نماید.

پاس کزاري

نویسنده پایان نامه می تواند مراتب امتنان خود را نسبت به استاد راهنما و استاد مشاور و یا دیگر افرادی که طی انجام پایان نامه به نحوی او را یاری و یا با او همکاری نموده اند ابراز دارد.

محمد رضینی فغانی

شهریور ۱۳۹۸

چکیده

در این قسمت چکیده پایان نامه نوشته می‌شود. چکیده باید جامع و بیان‌کننده خلاصه‌ای از اقدامات انجام‌شده باشد. در چکیده باید از ارجاع به مرجع و ذکر روابط ریاضی، بیان تاریخچه و تعریف مسئله خودداری شود.

واژه‌های کلیدی:

کلیدواژه اول، ...، کلیدواژه پنجم (نوشتن سه تا پنج واژه کلیدی ضروری است)

فهرست مطالب

صفحه

عنوان

۱	یادگیری تقویتی	۱
۲	۱-۱ تست	۲
۳	۲ جزئیات فنی پروژه	۳
۴	۱-۲ مقدمه	۴
۵	۲-۲ دورنمای کلی طرح	۵
۶	۱-۲-۲ تقسیم بندی وظایف هر بخش	۶
۷	۳-۲ معرفی نرم افزار پری اسکن	۷
۷	۱-۳-۲ بخش های مختلف نرم افزار پری اسکن	۷
۹	۲-۳-۲ فرمت های فایل های خروجی	۹
۱۰	۴-۲ بررسی دقیق تر فایل سیمولینک	۱۰
۱۱	۱-۴-۲ معرفی بلوک Environment و بررسی جزئیات آن	۱۱
۱۶	۲-۴-۲ بررسی ساختار داده های ارسالی و کد آن در سیمولینک	۱۶
۱۷	۵-۲ بررسی جزئیات بخش پایتون	۱۷
۱۷	۱-۵-۲ معرفی لایه های کد پایتون	۱۷
۱۸	۶-۲ بررسی دقیق تر برخی چالش های فنی پروژه	۱۸
۱۹	منابع و مراجع	۱۹
۲۰	پیوست	۲۰
۲۱	واژه نامه ی فارسی به انگلیسی	۲۱
۲۳	واژه نامه ی انگلیسی به فارسی	۲۳

فهرست اشکال

صفحه

شکل

۵	۱-۲	بلوک دیالگرام لایه های کلی
۷	۲-۲	بخشی از توانایی های نرم افزار پری اسکن در شبیه سازی
۸	۳-۲	آیکون های اضافه شده بر روی محیط دسکتاپ پس از نصب پری اسکن
۸	۴-۲	پنل مدیریت نرم افزار پری اسکن
۹	۵-۲	صفحه گرافیکی محیط پری اسکن
۱۰	۶-۲	فایل سیمولینک ایجاد شده توسط نرم افزار پری اسکن همراه با تغییرات
۱۱	۷-۲	روش صحیح اعمال تغییرات روی فایل سیمولینک
۱۲	۸-۲	فایل سیمولینک - شبیه سازی اتومبیل
۱۳	۹-۲	نگاهی از نزدیک به بلوک Environmnet
۱۴	۱۰-۲	فایل سیمولینک - داخل بلوک Environment
۱۶	۱۱-۲	ساختار جیسون برای بلوک های سیمولینک
۱۸	۱۲-۲	بلوک دیالگرام لایه های پایتون

فهرست جداول

صفحه

جدول

۹	۱-۲ توضیحات فرمت فایل خروجی
۱۲	۲-۲ بررسی ورودی ها و خروجی های مهم در شکل ۸-۲
۱۴	۳-۲ اطلاعات بلوک های فرستندگی گیرندگی در سیمولینک

فهرست نمادها

نماد	مفهوم
\mathbb{R}^n	فضای اقلیدسی با بعد n
\mathbb{S}^n	کره n یکه بعدی
M^m	خمینه m -بعدی M
$\mathfrak{X}(M)$	جبر میدان‌های برداری هموار روی M
$\mathfrak{X}^1(M)$	مجموعه میدان‌های برداری هموار یکه روی (M, g)
$\Omega^p(M)$	مجموعه p -فرمی‌های روی خمینه M
\mathcal{Q}	اپراتور ریچی
\mathcal{R}	تانسور انحنای ریمان
ric	تانسور ریچی
L	مشتق لی
Φ	۲-فرم اساسی خمینه تماسی
∇	التصاق لوی-چویتای
Δ	لاپلاسین ناهموار
∇^*	عملگر خودالحاق صوری القا شده از التصاق لوی-چویتای
g_s	متر ساساکی
∇	التصاق لوی-چویتای وابسته به متر ساساکی
Δ	عملگر لاپلاس-بلترامی روی p -فرم‌ها

فصل اول

یادگیری تقویتی

۱-۱ تست

فصل دوم

جزئیات فنی پروژه

۱-۲ مقدمه

در این پروژه از جهت آنکه نسخه قبلی و پیشینی برای آن نبوده است، به ناچار می‌بایست که کد آن از صفر تا صد آن به صورت دستی نوشته شود. از این‌رو، پیچیدگی‌های بسیار فراوان را به طور خاص در پی داشت. ابزارهای زیادی نیز بنابه شرایط در آن استفاده شد که ارتباط بین آن ابزارها و اجزاء بر این پیچیدگی پیاده‌سازی طرح افزوده بود.

ابزارهای اصلی و کلی که در این پروژه استفاده شده بود، عبارتند از:

- نرم افزار پری اسکن^۱ ، نسخه 8.5.0

- نرم افزار قدرتمند متلب^۲ ، نسخه R2017b

- زبان برنامه نویسی پایتون^۳ ، نسخه 3.6.9

بنابراین برای راه اندازی مجدد کد این پروژه لازم است که موارد بالا روی کامپیوتر شخص به صورت کامل نصب باشد.

همچنین لازم به ذکر است که برخی ابزارات دیگر نیز در این پروژه استفاده شده است که احتمالاً با نصب موارد بالا دیگر نیازی به نصب آن‌ها به صورت جداگانه نیست. هدف این ابزارها ایجاد اتصال بین اجزای اصلی گفته شده است. این گروه شامل موارد زیر هستند:

- سیمولینک^۴ ، جهت اتصال بین متلب و پری اسکن

- شبکه UDP^۵ ، جهت اتصال داده‌های پویا^۵ بین پایتون و سیمولینک

- موتور متلب^۶ ، جهت اتصال داده‌های ساکن^۷ بین پایتون و سیمولینک

در این فصل جزئیات بیشتری در مورد لزوم و دلیل استفاده از این ابزارها بررسی می‌شود.

¹PreScan

²Matlab

³Simulink

^۴برای این منظور از socket در پایتون استفاده شده است.

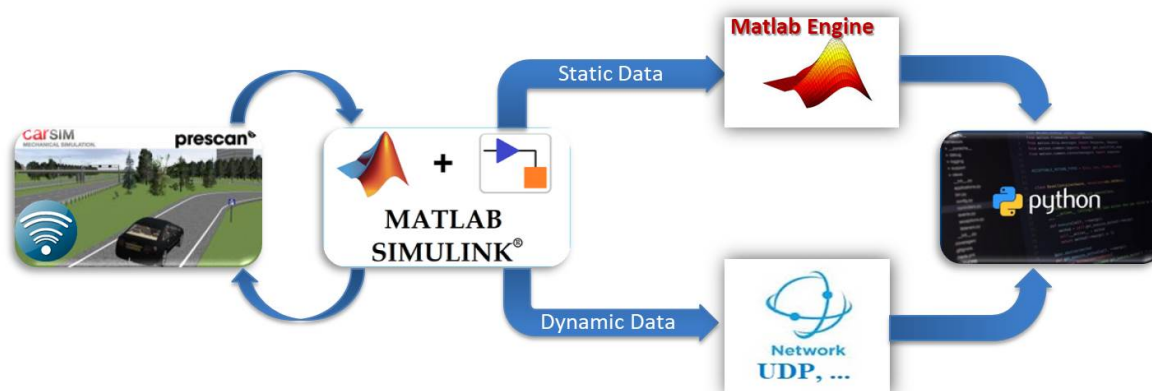
⁵Dynamic Data

⁶Matlab Engine

⁷Static Data

۲-۲ دورنمای کلی طرح

همان‌طور که گفته شد، در این پروژه از ابزار های مختلفی استفاده شده است. برخی ابزارات دیگر نیز جهت ایجاد اتصال بین آن ابزار ها استفاده شده اند. در این بخش، این اجزا به تفصیل بررسی خواهد شد. هر کدام از این اجزا کار مشخصی را بر عهده دارند. شکل ۱-۲ این ارتباط را نشان می‌دهد.



شکل ۱-۲: بلوک دیالگرام لایه های کلی

در شکل ۱-۲ از سمت چپ به راست اجزا یاد شده و نحوه ارتباط آن‌ها بایکدیگر را به‌خوبی نشان می‌دهد. این بلاک ها و ارتباط ها عبارتند از:

- اولین بلاک آن، نرم افزار **پری‌اسکن** می‌باشد. وظیفه اصلی این نرم افزار، شبیه سازی دینامیک یک اتومبیل و یا موتور و ... می‌باشد. همچنین ایجاد یک محیط گرافیکی زیبا و یک پنل کاربری گرافیکی برای ساخت ماشین ها از دیگر حسن های این نرم افزار است. فایل های مهم ایجاد شده توسط این بخش، `pex` و `pb` می‌باشد.
- بلاک بعدی ترکیبی از **متلب و سیمولینک** است. چرا که نرم افزار پری‌اسکن این امکان را دارد که برای کنترل و دسترسی بیشتر به قسمت های کنترلی مختلف، چیزی به نام API ارائه می‌دهد. این API یک فایل سیمولینک را در اختیار کاربران قرار میدهد که در آن بلوک های مشخصی به یکدیگر متصل هستند و با مطالعه و تغییر آن بلوک ها می‌توان کنترل سیستم را به دست گرفت. فایل های مهم این بخش نیز در فرمت `.slx` و `.m` در دسترس هستند.
- همچنین API یاد شده، دستورات دیگری را جهت دریافت داده های استاتیک محیط ساخته شده در این نرم افزار را به کاربران خویش در محیط متلب می‌دهد.

- دو بلوک بعدی، مربوط به اتصال بین متلب و یا سیمولینک با پایتون هستند.
- بلوک بالایی این اتصال را بین داده های استاتیک شامل طول جاده و عرض هر لاین، موقعیت اولیه اتومبیل و جاده، و بسیاری اطلاعات دیگر که بسیاری از آن اطلاعات استفاده نشده اند زیرا در این پروژه مفید نبوده اند. این بلوک، فایل سیمولینک را تغییر نمی دهد.
- بلوک پایینی نیز با استفاده از روش های شبکه کردن، می تواند داده های پویا را از محیط سیمولینک به پایتون منتقل کند. این داده های پویا عبارتند از موقعیت و سرعت و اطلاعات دیگری از اتومبیل در حال حرکت، اطلاعات سنسورها و ... باشد.
- بلوک بعدی پایتون است که خود شامل لایه های دیگری است که در شکل ۲-۱۲ به تفصیل بیان شده است. نکته جالب در آن این است که در آن لایه ها اثری نیز از دو بلوک پیشین آمده است. همچنین بخش اصلی کار، یا به عبارتی مغز و هوش این کار در این قسمت توسعه یافته است.

۱-۲-۲ تقسیم بندی وظایف هر بخش

بخش اصلی کار که وظیفه آن تصمیم گیری و انتخاب مسیر درست توسط یک عامل^۸ (که در این پروژه، عامل همان اتومبیل می باشد) در پایتون انجام می شود. وظیفه اصلی بخش متلب و سیمولینک و پری اسکن، ایجاد یک محیط شبیه سازی است.

یادداشت ۱-۲-۲. این محیط شبیه سازی اهمیت زیادی در الگوریتم های یادگیری تقویتی دارد. زیرا در این الگوریتم ها یک «عامل» با «محیط»^۹ در تعامل است. تعامل در این الگوریتم ها به معنای این است که «عامل» در یک «حالت»^{۱۰} قرار دارد. سپس متناسب با آن یک «حرکت»^{۱۱} انجام می دهد. با این «حرکت»، «محیط» به آن یک مقدار «امتیاز» و یک «حالت» جدید برمی گرداند. بنابراین داشتن یک محیط شبیه سازی کامل و دقیق از اجزای ضروری کار است.

بخش های دیگر مربوط به ارتباط این قسمت ها به یکدیگر بودند که پیچیدگی های زیادی را رقم زده است.

به طور ساده تر و کلی تر می توان گفت که پایتون نقش «عامل» و پری اسکن نقش «محیط» را دارد.

⁸Agent

⁹Environment

¹⁰State

¹¹Action



شکل ۲-۲: بخشی از توانایی های نرم افزار پری اسکن در شبیه سازی

۳-۲ معرفی نرم افزار پری اسکن

می توان در ابتدا گفت که این نرم افزار یک افزونه متلب و سیمولینک است اما توانایی زیادی که آن دارد باعث می شود که بگوییم این محصول از متلب و سیمولینک جعت رسیدن به هدف خود کمک می گیرد. نرم افزار پری اسکن یکی از نرم افزار های بسیار قدرتمند در زمینه شبیه سازی مسایل مربوط به وسایل نقلیه است که می تواند حرکت یک ویله نقلیه را به طور خیلی دقیق و مناسب شبیه سازی کند. همچنین در کنار این وظیفه مهم، یک محیط گرافیکی مناسب را در اختیار کاربران خود قرار می دهد که از دیگر حسن های آن است. شکل ۲-۲ این توانایی ها را به تصویر کشیده است.

همچنین این نرم افزار یک سری فایل خروجی به کاربر می دهد که یکی از فایل های آن فایل سیمولینک است که اجازه تغییر و دسترسی به داخل برخی بلاک ها به ما کمک می کند که اطلاعات خود را از دل آن نرم افزار بیرون بکشیم.^{۱۲}

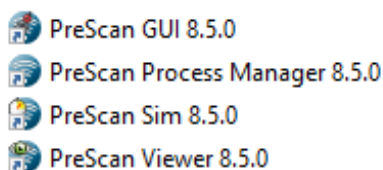
از این رو در مقایسه با محیط های دیگر، محاسن زیادی را داشت که هدف این پروژه را در به کارگیری این ابزار تحت تاثیر قرار داد.

۱-۳-۲ بخش های مختلف نرم افزار پری اسکن

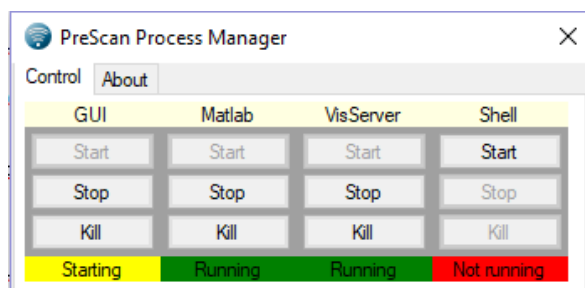
پس از دانلود و نصب نسخه 8.5.0 این نرم افزار چهار آیکون مانند شکل ۲-۳ به محیط دسکتاپ اضافه می کند. اصلی ترین آن ها PreScan Proccess Manager 8.5.0 نام دارد.

^{۱۲} جهت کسب اطلاعات بیشتر و تهیه این نرم افزار به لینک زیر مراجعه کنید:

<https://tass.plm.automation.siemens.com/prescan>



شکل ۲-۳: آیکون های اضافه شده بر روی محیط دسکتاپ پس از نصب پری اسکن
با انتخاب آن صفحه ای مانند زیر باز می شود.



شکل ۲-۴: پنل مدیریت نرم افزار پری اسکن

این پنجره شامل گزینه های زیر است:

- GUI •
- Matlab •
- VisServer •
- Shell •

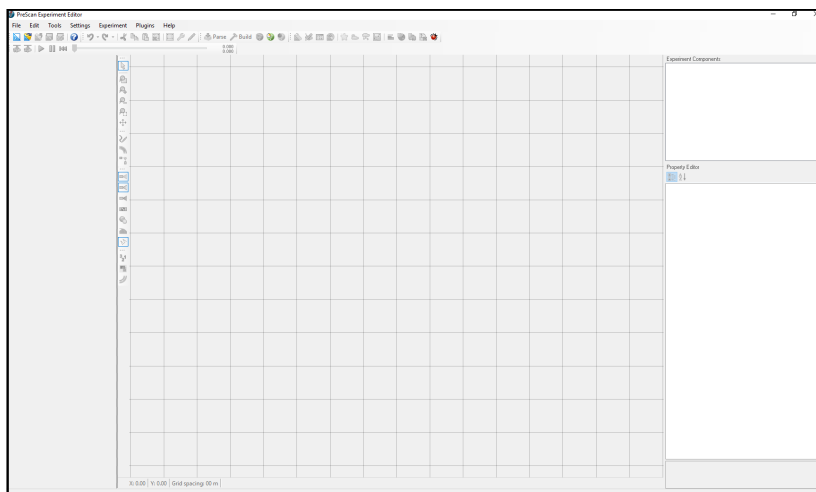
برای ایجاد یک محیط جدید باید GUI را استارت کرد. پس از مدتی صفحه ای مانند شکل ۲-۵ باز می شود.

پس از ایجاد مدل ها و ذخیره آن، فایل های **.pex و **.pb و **_cs.slx ساخته می شود.^{۱۳}
جهت استفاده از فایل سیمولینک باید در شکل ۲-۴ متلب را استارت کنید.

نکته ۲-۳-۱. برای اجرای فایل های سیمولینک خروجی، لازم است که متلب را فقط و فقط با استفاده از نرم افزار پری اسکن و با استفاده از پنل مدیریت نرم افزار معرفی شده در شکل ۲-۴ باز شود. در صورتی که به صورت مستقیم این کار انجام شود، به مشکل منتهی می شود.

دو قسمت دیگر نیز در شکل ۲-۴ وجود دارد که نیازی به استارت کردن آن ها نیست و خودشان در صورت لزوم به صورت خودکار فراخوانی می شوند.

^{۱۳} علامت ** به معنای یک اسم مشترک در این سه فایل استفاده شده است.



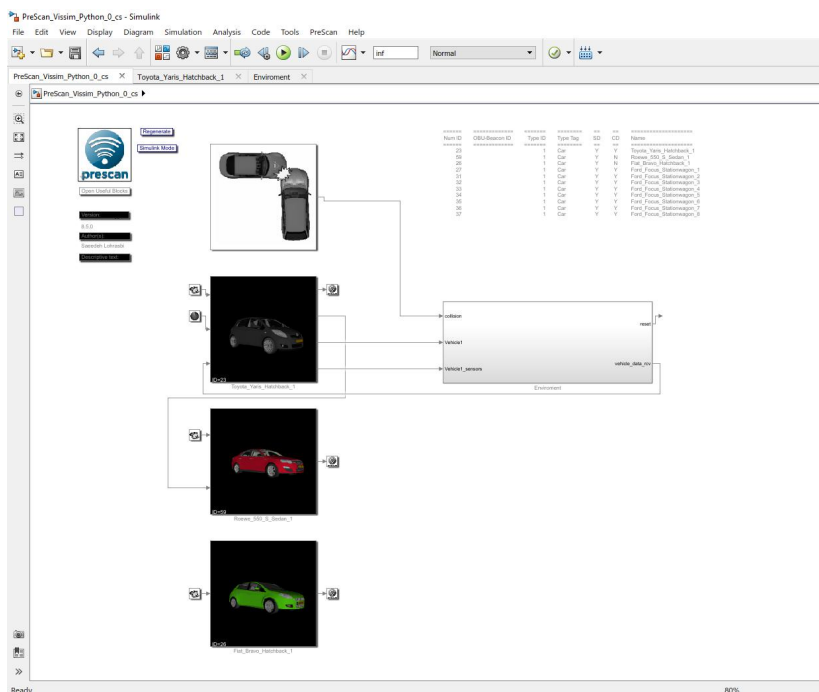
شکل ۲-۵: صفحه گرافیکی محیط پری‌اسکن

۲-۳-۲ فرمت های فایل های خروجی

نرم افزار پری‌اسکن پس از ایجاد یک محیط جدید، فایل ها و پوشه های بسیار زیادی را ایجاد می‌کند. اما در خارج آن پوشه ها ۳ فایل وجود دارد که پسوند آن ها `** .pex` و `** .pb` و `**_cs .slx` می‌باشد. علامت `**` همان اسم پروژه‌ای است که ایجاد کرده ایم. هر یک از این فایل ها به یک بلوک از شکل ۱-۲ مربوط می‌شود.

فرمت فایل	توضیحات
<code>** .pex</code>	این فایل مربوط به اولین بلوک شکل ۱-۲ است و ارتباط مستقیم با GUI دارد. برای تغییر محیط گرافیکی باید این فایل را باز کرد.
<code>** .pb</code>	این فایل برخی از اطلاعات فایل <code>** .pex</code> را در اختیار دارد و با تغییر آن فایل این فایل نیز عوض می‌شود. این فایل حاوی اطلاعات استاتیک محیط ایجاد شده است و مهم ترین کاربرد آن در بلوک موتور متلب که در شکل ۱-۲ نشان داده شده است می‌باشد. پایتون از طریق این فایل این اطلاعات را دریافت می‌کند.
<code>**_cs .slx</code>	این فایل سیمولینک است که برای کار کردن با آن باید از پنل مدیریت شکل ۴-۲ استفاده کرد. این فایل پس از ایجاد از فایل <code>** .pex</code> مستقل می‌شود. این فایل خود قابلیت تغییر دارد و می‌توان بلوک‌های آن را در محیط سیمولینک تغییر داد و بلوک های دیگری به آن افزود. در صورتی که فایل <code>** .pex</code> تغییر کند، این امکان را نیز دارد که از داخل خود سیمولینک با فشردن دکمه ای این تغییرات جدید اعمال شود بدون آن که به تغییرات خود کاربر لطمه ای وارد شود. در این پروژه این فایل، تغییرات بسیاری را تجربه کرد.

جدول ۱-۲: توضیحات فرمت فایل خروجی



شکل ۲-۶: فایل سیمولینک ایجاد شده توسط نرم افزار پری اسکن همراه با تغییرات

جدول ۲-۱ توضیحات لازم را جهت آشنایی با این خروجی ها آورده است.

همچنین در بخش ۲-۴ در مورد فایل *_cs.slx توضیحات دقیق تری در مورد جزئیات آن گفته خواهد شد.

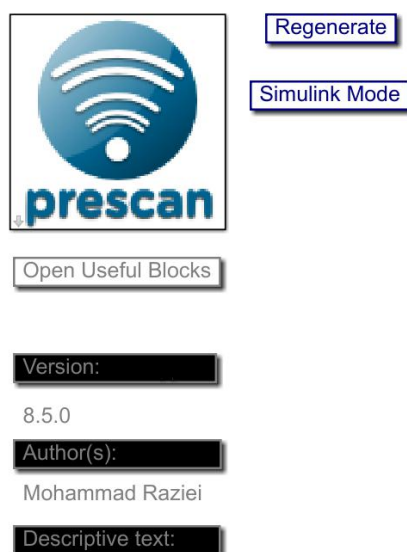
۲-۴ بررسی دقیق تر فایل سیمولینک

فایل سیمولینک ایجاد شده توسط نرم افزار پری اسکن، قابلیت تغییر به دست کاربر را دارد. شکل ۲-۶ فایل تغییر یافته مربوط به این پروژه را نشان می دهد.

بلوک های سمت راست نشان داده شده در سمت راست شکل ۲-۶ توسط نرم افزار پری اسکن ایجاد شده است که البته دست خوش تغییراتی نیز بوده اند.

در صورتی که با استفاده از محیط گرافیکی GUI فایل *.pex تغییر کند، فایل سیمولینک تغییر نمی کند. در برخی موارد این تغییرات ممکن است منجر به پیغام خطا شود.

نکته ۲-۴-۱. در صورتی که فایل *.pex تغییر کند، برای اعمال این تغییرات، باید روی کلمه Regenerate که در شکل ۲-۷ آمده است، کلیک کرد. با این کار، تغییرات جدید اعمال می شود بی آن که تغییرات کاربر تحت تاثیر قرار بگیرد.



شکل ۷-۲: روش صحیح اعمال تغییرات روی فایل سیمولینک

در شکل ۷-۲ همانطور که در نکته ۱-۴-۲ به آن اشاره شد، دکمه ای تحت عنوان Regenerate وجود دارد که استفاده از آن در همان نکته مشخص شده است. همچنین در این تصویر در زیر لوگوی برنامه پری اسکن، اطلاعاتی مانند شماره نسخه نرم افزار (که در این جا 8.5.0 می باشد)، نام نویسنده مشاهده می شود.

در شکل ۶-۲ اولین بلوک سمت راست همان ماشینی است که ما آن را تحت کنترل گرفته ایم. اگه به آن وارد شویم، شکل ۸-۲ را مشاهده می کنیم. در این تصویر ورودی و خروجی ها نقش خیلی مهمی دارند. این اطلاعات در جدول ۲-۲ آمده اند.

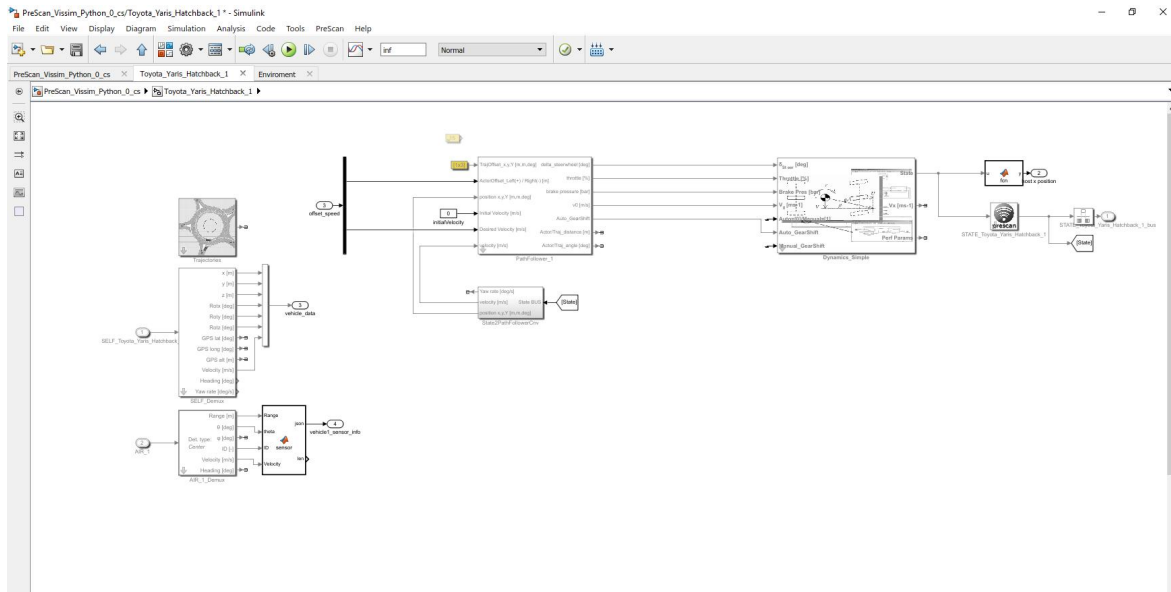
۱-۴-۲ معرفی بلوک Environment و بررسی جزئیات آن

در جدول ۲-۲ صرفا ورودی خروجی های مهم مورد بررسی قرار گرفته است. این ورودی ها و خروجی های مهم به یک بلوک دیگر منتقل می شود. بلوک Environment همان بلوک است که در شکل ۶-۲ نیز مشخص است و در شکل ۹-۲ نیز از زاویه نزدیک تر با جزئیات بیشتر می توان آن را مشاهده نمود. وظیفه اصلی بلوک Environment جمع آوری اطلاعات محیط سیمولینک و همچنین ارسال دستورات کنترلی به آن هاست. اطلاعات جمع آوری شده از محیط سیمولینک شامل موارد زیر است.

- اطلاعات ماشینی که نقش «عامل» را در الگوریتم دارد.

¹⁴json

¹⁵بعدا خواهید دید که مشخص بودن طول بسیار بسیار اهمیت دارد!

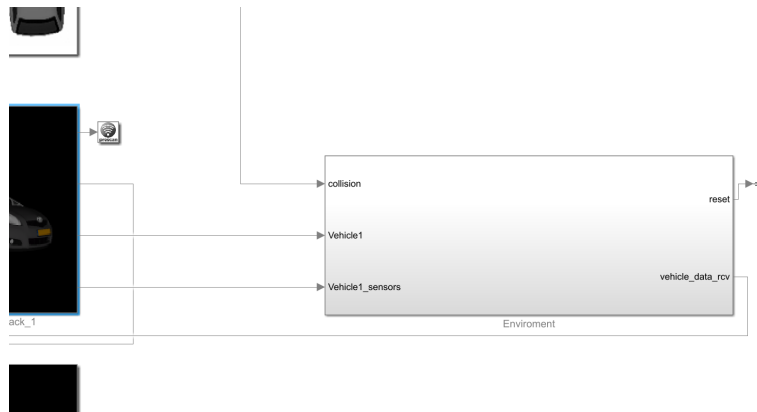


شکل ۸-۲: فایل سیمولینک - شبیه سازی اتومبیل

نوع	عنوان	بلوک مربوط	توضیحات
خروجی	اطلاعات ماشین	SELF_Demux	اطلاعات ماشین، شامل اطلاعات موقعیت (x, y) و (z) و اطلاعات چرخش (حول x, y, z) به همراه سرعت ماشین را خروجی می دهد. این ۷ داده قبل از خروجی توسط یک mux با یکدیگر ادغام می شوند.
خروجی	اطلاعات سنسور ماشین	AIR_1_Demux	اطلاعات سنسور V2C را خروجی می دهد. از آنجا که این سنسور فاصله و زاویه و ... تا ده ماشین نزدیک خود را می دهد. بنابراین هریک از این اطلاعات یک بردار ده تایی است. برای فرستادن آن اطلاعات به خروجی، ابتدا آن ها را به طریقی به فرمت جیسون ^{۱۴} تبدیل می کند و یک رشته کاراکتر با طول مشخص ^{۱۵} را خروجی می دهد.
ورودی	کنترل لاین و سرعت ماشین	PathFollower_1	دستورات کنترلی اتومبیل، شامل کدام خط بودن و مقدار سرعت نهایی، به صورت ورودی وارد یک demux می شود و پس از جدا سازی، به بلوک مربوط متصل می شود.

جدول ۲-۲: بررسی ورودی ها و خروجی های مهم در شکل ۸-۲

- اطلاعات سنسور های ماشین «عامل»
- اطلاعات مربوط به تصادف کردن و کدوم ماشین با کدوم ماشین تصادف کرده است.



شکل ۲-۹: نگاهی از نزدیک به بلوک Environmmnet

این اطلاعات به صورت ورودی به این بلوک وارد می‌شود و دستورات کنترلی شامل کنترل لاین ماشین به همراه مقدار سرعت نهایی ماشینی که نقش «عامل» در الگوریتم دارد، از این بلوک خارج می‌شود.

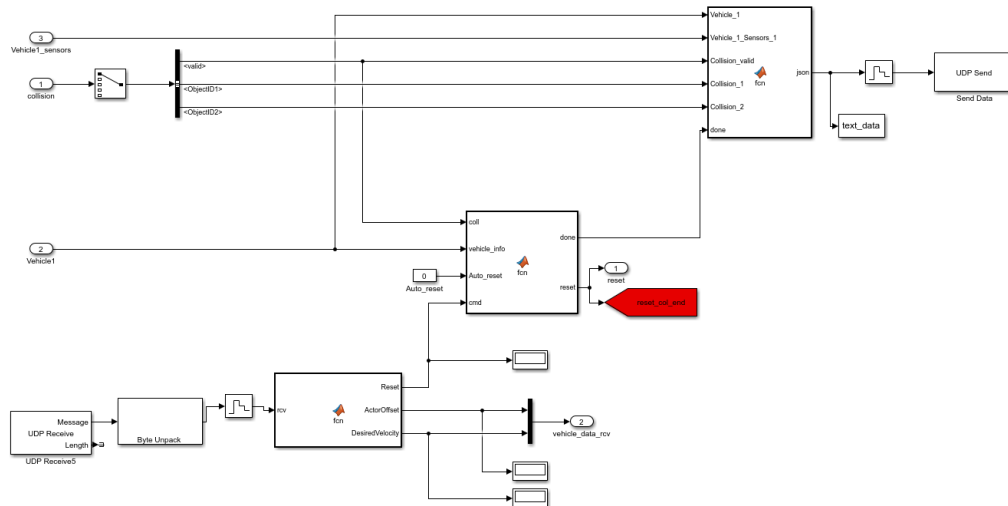
پیشتر صحبت شد که وظیفه تصمیم‌گیری بر عهده کد پایتون است. با این حساب سیمولینک قدرت تشخیص و تصمیم‌گیری ندارد. از این رو وظیفه بلوک Environment نیز تصمیم‌گیری نمی‌باشد بلکه با تکنیک‌هایی سعی بر برقراری ارتباط با کد پایتون دارد. در حقیقت این بلوک نقش واسط بین سیمولینک و پایتون را دارد. این بلوک علاوه بر دستورات کنترلی لاین و سرعت، دستور «شروع کردن دوباره» و یا ریست را نیز دریافت می‌کند. با این دستور تمامی اطلاعات به حالت اول بر خواهد گشت و ماشین «عامل» به جای اول بر خواهد گشت و منتظر دستورات جدید می‌ماند.

شکل ۲-۱ توضیح بسیار کلی در مورد این نحوه ارتباط نشان داده است و در شکل ۲-۱۰ جزئیات بیشتری را در مورد داخل این بلوک را نشان می‌دهد.

اگر به شکل ۲-۱۰ دقت شود دو بلوک UDP دیده می‌شود. (بلوک UDP Send در بالا سمت راست و بلوک UDP Receive در پایین سمت چپ شکل ۲-۱۰) این دو بلوک برای فرستادن داده‌های مورد نیاز به پایتون و گرفتن دستورهای کنترلی از پایتون در سیمولینک تعبیه شده‌اند.

یادداشت ۲-۴-۲. اگر نمودار شکل ۲-۱ در نظر داشته باشیم، خواهیم یافت که این بلوک در شاخه پایینی ارتباط بین سیمولینک و پایتون قرار دارد که از روش‌های شبکه کردن بین این دو انجام می‌شود. این بلوک صرفاً داده‌های پویا را از متلب به سیمولینک انتقال می‌دهد و با داده‌های استاتیک کاری ندارد.

در جدول ۲-۳ اطلاعات دقیق شبکه برای این دو بلوک دیده می‌شود. گفتنی است که این داده‌ها به نحوه مناسب کد شده‌اند تا تنها با استفاده از این دو بلوک بتوان داده‌ها را منتقل کرد.



شکل ۲-۱۰: فایل سیمولینک - داخل بلوک Environment

نام بلوک	نوع بلوک	IP	Port	توضیحات
UDP Receiver	گیرنده	localhost		
Send Data	فرستنده	localhost		

جدول ۲-۳: اطلاعات بلوک های فرستندگی گیرندگی در سیمولینک

روش کد شدن^{۱۶} قبل از ارسال و یا دریافت در این دو بلوک کاملاً با یکدیگر متفاوت هستند. این کار توسط بلوک‌های قبل و بعد دو بلوک مذکور انجام می‌شود. در شکل ۲-۱۰ سه بلوک تابع متلب^{۱۷} وجود دارد که از مهم‌ترین نقش را دارند. این نقش‌ها در ادامه توضیح توضیح مفصل داده شده‌اند. خلاصه این وظایف در زیر آمده است:

- **بلوک فرستنده:** این بلوک در سمت بالا سمت راست تمام اطلاعاتی که به بلوک کلی Environment وارد می‌شود را به نحوی مناسب به همراه متغیر done که از یک تابع متلب دیگر خارج می‌شود را برای پایتون طی ساختار مشخصی (جیسون) ارسال می‌کند. موضوع به صورت بسیط در بخش ۲-۴-۲ مورد بررسی قرار گرفته است.

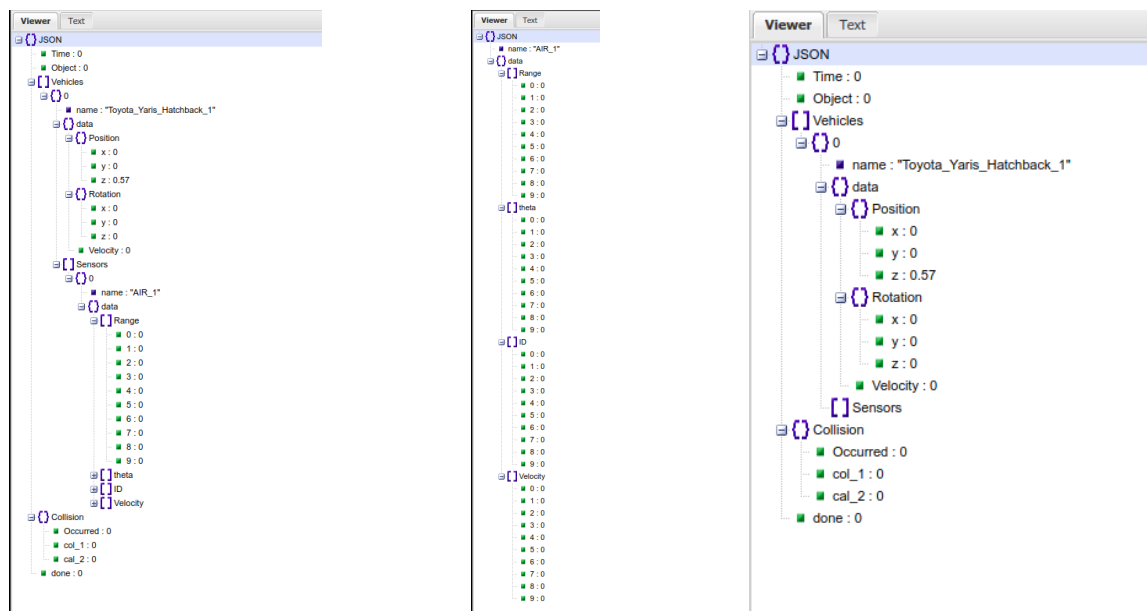
- **بلوک گیرنده:** این بلوک تنها وظیفه آن این است که داده‌هایی که پس از بلوک Byte Unpacking آمده است، را انتخاب می‌کند که هر یک از این دیتاها چه مفهومی هستند. از آن‌جا که ۳ داده کنترلی (لاین و سرعت و ریست) از طرف پایتون ارسال می‌شود، بلوک Byte Unpacking وظیفه این ۳ داده را به فرمت double در می‌آورد (این فرمت در سمت پایتون نیز به همین شکل قرار داده شده است). بنابراین این بلوک Byte Unpacking است که کار اصلی را انجام می‌دهد و تنها وظیفه این بلوک تفکیک و اسم گذاری برروی خروجی بلوک Byte Unpacking است.

- **بلوک محاسبه تمام شدن و ریست کردن:** این بلوک با دو خروجی مهم می‌دهد. یکی از آن خروجی‌ها done است که توسط بلوک فرستنده نیز برای پایتون ارسال می‌شود. و مقدار دیگری را که خروجی می‌دهد دستور ریست کردن است. این دستور در لبه مثبت (از صفر به یک برسد) ریست می‌کند. منطق این بلوک یک منطق باینری است که در شکل؟؟ روش محاسبه آن نیز آمده است. همچنین این بلوک یک مقدار به اسم Auto_reset_flag نیز به عنوان ورودی دریافت می‌کند. این مقدار که می‌تواند صفر و یا یک باشد، تصمیم می‌گیرد که در صورتی که $done = 1$ شد آیا ریست کند و یا ریست نکند. اگر یک باشد ریست می‌کند و گرنه منتظر رسیدن دستور از پایتون می‌ماند.

¹⁶Encoding¹⁷Matlab-function block

۲-۴-۲ بررسی ساختار داده های ارسالی و کد آن در سیمولینک

بلوک فرستنده در شکل ۲-۱۰ اطلاعات دریافت شده را طی ساختار مشخصی به ساختار جیسون در می آورد. نمونه ای از این ساختار در شکل ۲-۱ آمده است.



شکل ادغام شده (ج)

(ب) جیسون
سنسور

(آ) اطلاعات خام جیسون کلی

شکل ۲-۱۱: تصویر (آ) نحوه ساختاردهی کلی در بلوک تابع متلب واقع در زیرسیستم Environment را نشان می دهد. در این قسمت اطلاعات ماشین شامل دو قسمت data و Sensors است. اطلاعات سنسور های این ماشین از تصویر (ب) تامین می شود. این ساختار در همان زیرسیستم ماشین تبدیل به جیسون شده و برای زیرسیستم Environmnet ارسال می شود. قسمت سنسور یک آرایه است تا بتوان چندین سنسور مختلف را برای آن ارسال کرد. بنابراین ساختاری مانند تصویر (ج) باید را در نهایت برای پایتون ارسال می کند.^{۱۹}

شکل ۲-۱۱(ب) همان ساختاری است که پیشتر در مورد آن صحبت شد. با توجه به این که این ساختار (در ادامه خواهید دید که) در اطلاعات سنسور ماشین به کار گرفته می شود و هر سنسور ممکن است اطلاعات خاص خود را داشته باشد. برای یکسان سازی این اطلاعات، در اولین لایه دو گزینه data و name را به طور قرار دادی بین همه سنسور ها یکسان است تا در کد پایتون بتوان با توجه به اسم آن ها ادامه ساختار که در data می باشد، قابل تشخیص باشد. اما از آن جا که صرفاً از اطلاعات یک سنسور استفاده شده است این شکل ساختاردهی تنها یک گزینه برای توسعه های آتی آن است.

در این ساختار ۴ اطلاعات نشان داده شده که هر یک از آن اطلاعات خود یک بردار ۱۰ تایی است، برای پایتون ارسال می شود.

^{۱۹} این عکس با استفاده از سایت <http://jsonviewer.stack.hu/> تهیه شده است.

شکل ۲-۱۱(آ) ساختار کلی است که محیط Environmnet از آن استفاده می‌کند. در این بلوک علاوه بر داده‌هایی مانند اطلاعات ماشین و سنسورش، اطلاعات تصادف و اطلاعات تمام شدن و یا نشدن شبیه‌سازی که پیش‌تر در مورد آن اطلاعات صحبت شد، دو اطلاعات اضافه دیگر نیز می‌فرستد.

- Time: زمان شبیه‌سازی را همراه با دیتا دیگر ارسال می‌کند و به اصلاح یک ساختار زمانی ایجاد می‌شود.^{۲۰}

- Object: این عبارت کمک به کد پایتون می‌کند که ماشین هدف و یا عامل را از بین ماشین‌های موجود در لیست Vehicles بیابد.^{۲۱}

در نهایت با ادغام شدن اطلاعات سنسور نیز، ساختار کلی به شکل ۲-۱۱(ج) در خواهد آمد. در بخش ۲-۶ از برخی از چالش‌های ساختاردهی کردن به این شکل، صحبت خواهد شد. همچنین خروجی‌هایی آن نیز را می‌توانید در همان بخش بیابید.

۲-۵ بررسی جزئیات بخش پایتون

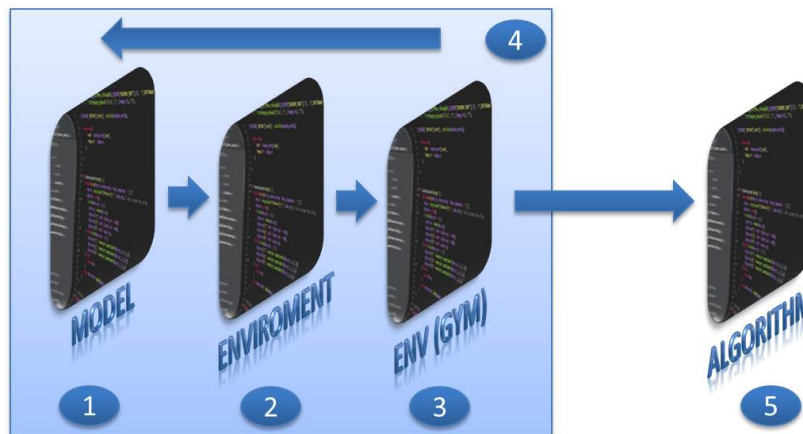
بارها اشاره شد که پایتون بخش اصل تصمیم‌گیری را برعهده دارد. در بخش ۲-۴ سعی شد تا به نحو مناسبی دیتای محیط شبیه‌سازی را به پایتون منتقل کند و دستورات کنترلی را نیز از پایتون به محیط شبیه‌سازی ارسال کند. در این بخش بر روی مفاهیم پایتونی آن مانور می‌دهیم. از آن‌جا که کد پایتون باید به سیمولینک متصل شود و دیتا را به الگوریتم کنترلی خود (که از الگوریتم‌های یادگیری تقویتی استفاده شده است)، برساند و از آن‌جا دستورات را دریافت کند و به سیمولینک برساند، نیازمند لایه‌هایی است که هر لایه بخشی از این کارها انجام دهد.

۲-۵-۱ معرفی لایه‌های کد پایتون

کد پایتون از لایه‌های مختلف تشکیل شده است. از هر لایه به لایه بعدی سطح زبان بالاتر می‌رود. این لایه‌ها در شکل ۲-۵-۱ مشخص شده‌اند. در لایه‌های ابتدایی، سطح استفاده از دستورات بسیار ابتدایی است و در هر لایه با تعریف توابع و کلاس‌هایی این امکان را ایجاد کرده‌اند که بدون در نظر گرفتن

²⁰Time-struct

^{۲۱}از آنجایی که در این پروژه لیست Vehicles یک آبجکت بیشتر ندارد، پس این بخش نیز صرفاً ظرفیت توسعه‌پذیری این کد را بالا می‌برد.



شکل ۲-۱۲: بلوک دیالگرام لایه های پایتون

این که در سطوح پایین تر چه اتفاقاتی می افتد، در سطوح بالاتر از آن امکانات استفاده کرد. در ادامه این بحث مفصل توضیح داده خواهد شد.

در شکل ۲-۱۲ ۴ لایه برنامه نویسی شده با ۵ عدد نشان داده شده است. توضیحات زیر متناسب با هریک از این شماره ها در نظر گرفته شده اند:

۱. **Model** (مدل) : این بخش

۲. **Environment** (محیط) :

۳. **Env** :

۴. **gym** :

۵. **Algorithm** (الگوریتم) :

۲-۶ بررسی دقیق تر برخی چالش های فنی پروژه

```

'{"Time":0,"Object":0,"Vehicles":[{"name":"Toyota_Yaris_Hatchback_1","data":{"Position":{"x":0.00000e+00,"y":0.00000e+00,"z":5.70000e-01},"Rotation":{"x":0.00000e+00,"y":0.00000e+00,"z":0.00000e+00},"Velocity":0.00000e+00},"Sensors":[{"name":"AIR_1","data":{"Range":[0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00],"theta":[0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00],"ID":[0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00],"Velocity":[0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00,0.00000e+00]}]}],"Collision":{"Occurred":0,"col_1": 0,"col_2": 0},"done":0}'
  
```

منابع و مراجع

پیوست

موضوعات مرتبط با متن گزارش پایان نامه که در یکی از گروه‌های زیر قرار می‌گیرد، در بخش پیوست‌ها آورده شوند:

۱. اثبات‌های ریاضی یا عملیات ریاضی طولانی.
۲. داده و اطلاعات نمونه (های) مورد مطالعه (Case Study) چنانچه طولانی باشد.
۳. نتایج کارهای دیگران چنانچه نیاز به تفصیل باشد.
۴. مجموعه تعاریف متغیرها و پارامترها، چنانچه طولانی بوده و در متن به انجام نرسیده باشد.

کد میپل

```
with(DifferentialGeometry):  
with(Tensor):  
DGsetup([x, y, z], M)  
frame name: M  
a := evalDG(D_x)  
D_x  
b := evalDG(-2 y z D_x+2 x D_y/z^3-D_z/z^2)
```

واژه‌نامه‌ی فارسی به انگلیسی

آ	Cartesian product حاصل ضرب دکارتی
اسکالر Scalar	خ
ب	Automorphism خودریختی
بالابر Lift	د
پ	Degree درجه
پایا Invariant	ر
ت	microprocessor ریزپردازنده
تناظر Correspondence	ز
ث	Submodule زیرمدول
ثابت‌ساز Stabilizer	س
ج	Character سرشت
جایگشت Permutation	ص
چ	Faithful صادقانه
چند جمله‌ای Polynomial	ض
ح	

Connected	همبند	Inner product	ضرب داخلی
	ی		ط
Edge	یال	Loop	طوقه
			ظ
		Valency	ظرفیت
			ع
		Nonadjacency	عدم مجاورت
			ف
		Vector space	فضای برداری
			ک
		Complete reducibility	کاملاً تحویل پذیر
			گ
		Graph	گراف
			م
		Permutation matrix	ماتریس جایگشتی
			ن
		Disconnected	ناهمبند
			و
		Invertible	وارون پذیر
			ه

واژه‌نامه‌ی انگلیسی به فارسی

A	Homomorphism همریختی
Automorphism خودریختی	I
B	Invariant پایا
Bijection دوسویی	L
C	Lift بالابر
Cycle group گروه دوری	M
D	Module مدول
Degree درجه	N
E	Natural map نگاشت طبیعی
Edge یال	O
F	One to One یک به یک
Function تابع	P
G	Permutation group گروه جایگشتی
Group گروه	Q
H	

Quotient graph	گراف خارج‌قسمتی	Trivial character	سرشت بدیهی
R		U	
Reducible	تحویل پذیر	Unique	منحصربفرد
S		V	
Sequence	دنباله	Vector space	فضای برداری
T			

Abstract

This page is accurate translation from Persian abstract into English.

Key Words:

Write a 3 to 5 KeyWords is essential. Example: AUT, M.Sc., Ph. D, ..