

تخمین پارامترهای دینامیک طولی خودرو به روش شبیه‌سازی سخت‌افزار در حلقه

کنترل مناسب دینامیک طولی خودرو نیاز به داشتن اطلاع دقیق از مقدار پارامترهای دینامیک خودرو دارد. در این مقاله الگوریتم تخمین پارامتر با نام فرم درجه دوم خطی به صورت زمان کوتاه ($STLQF$) برای تخمین هم‌زمان پارامترهای جرم خودرو، زاویه شیب جاده و ضریب درگ کلی آیرودینامیک خودرو توسعه داده می‌شود. سپس، کارکرد مؤثر تخمین‌گر $STLQF$ با مقایسه عملکرد این تخمین‌گر با روش حداقل مربعات بازگشتی (RLS) نشان داده می‌شود. در پایان، الگوریتم $STLQF$ به صورت سخت‌افزار در حلقه شبیه‌سازی می‌شود تا از این طریق تخمین‌گر $STLQF$ با حداقل هزینه و در کوتاه‌ترین زمان برای نمونه-سازی سریع توسعه پیدا کند.

ایمان تاهباززاده مقدم^۱

کارشناسی ارشد

جواد مرزبان راد^۲

دانشیار

واژه‌های راهنما: دینامیک طولی خودرو، تخمین پارامتر، فرم درجه دوم خطی به صورت زمان کوتاه، شبیه-سازی سخت‌افزار در حلقه

۱- مقدمه

با توجه به اهمیت پیش‌گیری از سوانح رانندگی، توسعه‌ی سیستم‌های هوشمند حمل و نقل و تجهیزات کمک به راننده^۲ مورد توجه سازمان‌های کنترل ترافیک، شرکت‌های خودروسازی و پژوهشگران موسسه‌های تحقیقاتی قرار گرفته است. سیستم‌های کنترل سرعت تطبیقی^۴ خودرو، کنترل پایداری الکترونیکی^۵ خودرو، کنترل کشش^۶ خودرو و سایر سامانه‌های درون خودرویی نمونه‌های تجاری از این دست می‌باشند که کارکرد زیادی در خودروهای سواری پیدا کرده‌اند. هدف این سیستم‌ها، فراهم نمودن آسایش بیشتر برای راننده از طریق گازدهی، ترمزگیری و فرمان‌دهی وسیله نقلیه به صورت خودکار و بدون نیاز به کمک راننده می‌باشد. علی‌رغم تحقیقات وسیعی که در سال‌های اخیر جهت توسعه سیستم‌های کنترل دینامیک خودرو انجام شده است؛ لیکن عمده‌ی این تحقیقات مربوط به طراحی الگوریتم‌های متنوع برای لایه‌های مختلف سامانه‌های کنترلی خودرو می‌باشد.

^۱ نویسنده مسئول، کارشناسی ارشد، دانشکده مهندسی خودرو، دانشگاه علم و صنعت ایران i.tahbaz@gmail.com

^۲ دانشیار، دانشکده مهندسی خودرو، دانشگاه علم و صنعت ایران

^۳ Driving Assistant Systems (DAS)

^۴ Adaptive Cruise Control (ACC)

^۵ Electronic Stability Control (ESC)

^۶ Traction Control System (TCS)

این درحالی است که فارغ از نوع الگوریتم حاکم بر سامانه کنترلی، عملکرد این سیستم‌ها متأثر از یک چالش عمده می‌باشد: وجود عدم قطعیت‌های پارامتری در دینامیک خودرو و تغییرات در زاویه شیب جاده که تأثیر زیادی بر پاسخ دینامیک طولی و عرضی خودرو می‌گذارد [۱].

با توجه به تأثیر بسیار زیاد نامعینی‌های پارامتری بر کنترل دینامیک خودرو و اهمیت مدل‌سازی آن، در طی سال‌های اخیر پژوهش‌های قابل توجهی در زمینه تخمین پارامترهای مؤثر بر دینامیک خودرو صورت گرفته است. درعین حال، علاقه شرکت‌های خودروسازی و مؤسسات تحقیقاتی به توسعه الگوریتم‌های تخمین‌گر مدل - مبنا جهت تخمین پارامترهای جرم و زاویه شیب جاده که قابلیت کارکرد در کنترل دینامیک طولی و عرضی خودرو دارند، رو به گسترش است [۱]. نامعینی‌های پارامتری در دینامیک طولی خودرو، عملکرد سیستم‌های کنترل دینامیک خودرو را به شدت تحت تأثیر قرار می‌دهند. جرم خودرو که وابسته به نحوه بارگذاری خودرو می‌باشد در طول سفرهای مختلف و با توجه به تعداد سرنشینان خودرو، جرم بار صندوق و جرم سوختی که خودرو حمل می‌کند امکان تغییر دارد [۲]. تغییر در نوع جاده و تغییرات پیوسته در زاویه شیب جاده عامل مهم دیگری می‌باشد که در صورت همراه شدن با تغییرات زیاد در جرم خودرو تأثیر نامساعدی را بر رفتار دینامیکی خودرو می‌گذارد [۳]، [۴].

عامل مهم دیگر، تغییر در مقدار ضریب درگ کلی آیرودینامیک خودرو وابسته به تغییر شکل در بدنه خودرو بدلیل نصب باربند، باز بودن شیشه‌های خودرو و غیره می‌باشد که بخصوص در سرعت‌های بالاتر از ۶۰ کیلومتر بر ساعت، تبدیل به عاملی مقاوم در برابر حرکت خودرو می‌شود. این نامعینی‌ها کارکرد مطلوب سیستم‌های کنترلی خودرو را دچار مشکل می‌کنند و در صورتی که الگوریتم کنترلی مورد نظر امکان تطبیق با تغییراتی که در دینامیک خودرو و جاده ایجاد می‌شود نداشته باشد، نظیر تغییر زاویه شیب جاده و تغییر در نوع جاده، امکان ناپایداری سیستم کنترلی وجود دارد [۵].

این در حالی است که در بسیاری از مطالعات صورت گرفته در حوزه طراحی و توسعه سامانه‌های کنترل دینامیک خودرو نه تنها مقادیر ثابتی برای پارامترهای مؤثر بر دینامیک طولی خودرو نظیر جرم خودرو، ضریب درگ کلی آیرودینامیک خودرو، زاویه شیب جاده و اصطکاک بین تیر خودرو و جاده در نظر گرفته شده است، بلکه استراتژی کنترلی ثابتی نیز که امکان تطبیق رفتار با تغییرات دینامیک خودرو ندارد مورد استفاده قرار داده شده است. تحقیقات تجربی که بر روی سیستم‌های تعقیب سرعت خودرو انجام شده است نشان می‌دهند که کنترلرهای مرسوم نظیر PID در صورتی که با ضرایب ثابت طراحی شوند توانایی محدودی در کنترل سرعت خودرو در حضور عدم قطعیت‌های پارامتری دارند [۶].

راهکار مناسب‌تر برای بهبود عملکرد یک کنترل‌کننده سرعت با ساختار مرسوم، توسعه الگوریتم تخمین پارامتری است که پارامترهای دینامیک طولی خودرو را به صورت سریع و بلادرنگ شناسایی کند و عملکرد الگوریتم کنترلی خودتنظیم را در یک سیستم حلقه بسته بهبود دهد. در مطالعات [۳] و [۷]، الگوریتم کنترل تطبیقی مدل مرجع^۱ برای کنترل سرعت خودرو به نحوی توسعه داده شده است که امکان تعقیب سرعت مبنا همراه با تخمین پارامترهای جرم خودرو و زاویه شیب جاده فراهم می‌شود. پژوهش [۸] یک جبران‌کننده تطبیقی را جهت تخمین بلادرنگ پارامترهای جرم و اینرسی چرخشی مدل سه درجه آزادی

^۱ Model Reference Adaptive Control (MRAC)

خودرو شامل دینامیک صفحه‌ای طراحی کرده است. در مطالعات [۹]، [۱۰] و [۱۱] روش‌های مبتنی بر داده‌های سنسوری جهت تخمین مقدار متغیر با زمان شیب جاده ارائه شدند که به ترتیب از داده‌های سنسور شتاب‌سنج طولی، سیستم موقعیت‌یاب جهانی^۱ و سیستم تشخیص نور و مسافت‌یابی^۲ استفاده می‌کردند. مطالعه [۱۲] با استفاده از مجموع اطلاعات اندازه‌گیری شده توسط GPS و سنسور واحد اندازه‌گیری اینرسی^۳، موفق به استخراج ویژگی‌های سطح جاده شده است؛ به نحوی که بر مبنای این ویژگی‌ها مقادیر شیب جاده و شعاع انحنای جاده بدست آمدند.

علاوه بر این، مروری بر پژوهش‌های پیشین نشان می‌دهد که محققان الگوریتم‌های مدل-مبنای متنوعی را جهت تخمین جرم و اینرسی خودرو توسعه داده‌اند. خانواده‌ی تخمین‌گرهای مبتنی بر روش حداقل مربعات^۴ نظیر حداقل مربعات بازگشتی^۵، مجموع حداقل مربعات^۶ و روش حداقل مربعات اصلاح شده بوسیله فاکتور فراموشی دسته‌ای از این روش‌ها هستند که بر روی مدل‌های دینامیک تعلیق، دینامیک عرضی، دینامیک طولی و دینامیک سیستم انتقال قدرت خودرو پیاده‌سازی شده‌اند [۲]، [۱۳]، [۱۴]. مطالعات [۱۵] و [۱۶] جهت تخمین هم‌زمان جرم و مقدار زاویه متغیر با زمان شیب جاده برای خودروهای سنگین و سواری به ترتیب از روش RLS و تئوری لیاپانوف با رویکرد برگشتی^۷ استفاده کرده‌اند. مطالعه [۱۷] تخمین‌گر دو مرحله‌ای نوینی را مبتنی بر روش حداقل مربعات آشیانه‌ای^۸ جهت تخمین پارامترهای جرم خودرو، ضریب مقاومت غلتشی جاده و سرعت باد که همگی با مقدار ثابت فرض شده‌اند ارائه کرده است. مطالعه [۱۸] از مدل دینامیکی پنج درجه آزادی خودرو شامل دینامیک تعلیق، جهت تخمین پارامترهای جرم و اینرسی خودرو استفاده کرده است. در این مطالعه، چهار روش شناسایی به نام‌های فیلتر کالمن بازگشتی^۹، فیلتر کالمن توسعه یافته^{۱۰}، تخمین‌گر مبتنی بر گرادیان نزولی^{۱۱} و RLS مورد استفاده قرار گرفته‌اند و نتایج تخمین آن‌ها با یکدیگر مقایسه شده است. در مطالعه [۱۹] روش فیلتر کالمن دوگانه برای تخمین پارامترهای جرم معلق و موقعیت آنی مرکز جرم ارائه شده است. سپس مقدار اینرسی خودرو با استفاده از دو پارامتر تخمینی به دست آمده است. مطالعه [۲۰] نیز روش فیلتر کالمن دوگانه خنثی^{۱۲} را برای تخمین هم-زمان نیروهای عرضی تایلر، زاویه لغزش جانبی و جرم خودرو به کار برده است.

هرچند در پژوهش‌های صورت گرفته تاکنون، روش‌های متنوعی جهت تخمین پارامترهای مؤثر بر دینامیک طولی خودرو توسعه داده شده‌اند، لیکن نقاط ضعف عمده‌ای در عملکرد الگوریتم‌های تخمین‌گر ارائه شده در این پژوهش‌ها مشاهده می‌شود که از جمله‌ی آن‌ها می‌توان به این موارد اشاره کرد:

¹ Global Positioning System (GPS)

² Light Detection and Ranging (LIDAR)

³ Inertial Measurement Unit (IMU)

⁴ Least Squares (LS)

⁵ Recursive Least Squares (RLS)

⁶ Total Least Squares (TLS)

⁷ Lyapanov Theory with Back Stepping Approach

⁸ Nested Least Squares

⁹ Recursive Kalman Filter (RKF)

¹⁰ Extended Kalman Filter (EKF)

¹¹ Gradient-Based

¹² Unscented Dual Kalman Filter

- دقت ناکافی در تخمین پارامترها [۱]، [۱۶]
- بار محاسباتی بالا [۷]، [۸]، [۲۱]
- سرعت همگرایی اندک [۳]، [۱۰]
- تعریف فرضیات ساده‌کننده نظیر تعیین محدوده برای تغییر پارامترها [۱۳]، [۱۵]
- ناتوانی در تخمین پارامترهای متغیر با زمان [۱۱]، [۱۴]

در این مطالعه، معادلات حاکم بر مدل دینامیک طولی یک خودروی سواری با موتور بنزینی و سیستم جعبه دنده خودکار استخراج می‌شود. سپس، الگوریتم تخمین پارامتر با نام فرم درجه دوم خطی به صورت زمان کوتاه^۱ [۲۲] که یک روش تخمین گر جدید می‌باشد برای تخمین هم‌زمان و بلادرنگ پارامترهای جرم خودرو، زاویه شیب جاده و ضریب درگ کلی آیرودینامیک توسعه داده می‌شود. از ویژگی‌های بارز این تخمین گر، دقت بالا، تأخیر اندک در تخمین پارامترها و بار محاسباتی مناسب می‌باشد. در پایان، تخمین گر STLQF به صورت سخت‌افزار در حلقه^۲ شبیه‌سازی می‌شود تا از این طریق الگوریتم STLQF با حداقل هزینه و در کوتاه‌ترین زمان برای یک نمونه‌سازی سریع توسعه پیدا کند.

۲- مدل سازی دینامیک طولی خودرو

در این بخش، نحوه استخراج معادلات حاکم بر مدل دینامیک طولی یک خودروی سواری با سیستم جعبه دنده خودکار تشریح می‌شود. همان‌طور که در شکل (۱) نشان داده شده است، مدل دینامیک طولی خودرو شامل دو زیر قسمت اصلی می‌باشد [۲۳]: (۱) دینامیک سیستم انتقال قدرت که اجزای اصلی آن موتور، مبدل گشتاور^۳، جعبه دنده خودکار، محورها و چرخ‌ها است؛ (۲) دینامیک حرکتی خودرو که نیروهای خارجی و مقاوم وارد بر خودرو را مدل می‌کند. مدل‌های شماتیک شبیه‌سازی موتور و مبدل گشتاور به ترتیب در شکل‌های (۲) و (۳) نشان داده شده‌اند.

عمده فرض ساده‌سازی که برای این مدل دینامیکی مفروض شده است، صرف نظر از دینامیک تایلر خودرو می‌باشد؛ بنابراین، فرض عدم لغزش در تماس تایلر خودرو و زمین نیز مستقیماً اعمال می‌شود. این فرض، در اغلب مطالعاتی که در حوزه کنترل دینامیک طولی خودرو انجام شده است، مطرح شده است [۲۴]. لازم به ذکر است که صرف نظر کردن از مدل دینامیک تایلر خودرو باعث تخمین بیش از حد نیروی کشنده خودرو می‌شود که برای جاده‌های خشک با دقت بالایی قابل قبول است. معادله حاکم بر دینامیک موتور با در نظر گرفتن بار گشتاور پمپ که بر موتور وارد می‌شود به شکل رابطه (۱) است [۲۳]:

$$I_{ei} \dot{\omega}_e = T_e(\alpha_e, \omega_e) - T_i \quad (1)$$

¹ Short Time Linear Quadratic Form (STLQF)

² Hardware in the Loop Simulation (HILS)

³ Torque Converter

در رابطه (۱)، T_e نشان‌دهنده گشتاور موتور، α_e درصد بازشدگی دریچه گاز، ω_e سرعت دورانی موتور و I_{ei} مجموع لختی اینرسی موتور و پمپ می‌باشد. در این رابطه گشتاور پمپ از رابطه $T_i = \left(\frac{\omega_e}{K_f}\right)^2$ بدست می‌آید و گشتاور توربین از طریق یک رابطه تناسبی با گشتاور پمپ طبق رابطه (۲) بدست می‌آید.

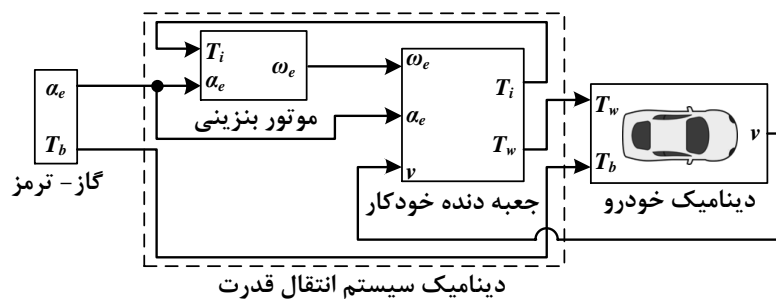
$$T_t = C_{tr} T_i \quad (2)$$

در روابط (۱) و (۲)، K_f و C_{tr} به ترتیب فاکتور ظرفیت و نسبت گشتاور مبدل می‌باشند که تابعی غیرخطی از نسبت سرعت مبدل گشتاور $C_{sr} = \omega_{in} / \omega_e$ هستند. سرعت ورودی جعبه دنده که با ω_{in} نشان داده شده است، طبق رابطه (۳) با سرعت خودرو v رابطه پیدا می‌کند:

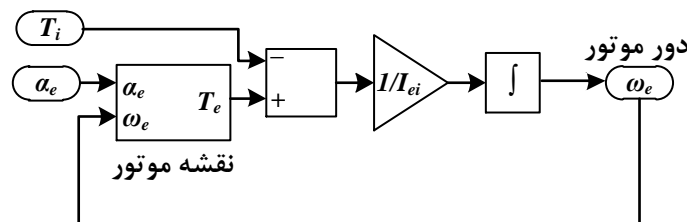
$$\omega_{in} = \frac{R_g R_{fd} v}{r_w} \quad (3)$$

در رابطه (۳)، R_g ضریب دنده، R_{fd} ضریب دیفرانسیل و r_w شعاع مؤثر چرخ می‌باشد. دیاگرام سیستم جعبه دنده خودکار شامل قانون تعویض دنده و مدل‌های مبدل گشتاور و تولید گشتاور در چرخ‌ها در شکل (۴) نشان داده شده است. قانون تعویض دنده تابعی از سرعت خروجی سیستم جعبه دنده ω_{out} و درصد بازشدگی دریچه گاز است که نحوه تعویض دنده را معین می‌کند. گشتاور کلی که در چرخ‌ها تولید می‌شود طبق رابطه (۴) با گشتاور توربین که خود تابعی از گشتاور موتور می‌باشد رابطه دارد:

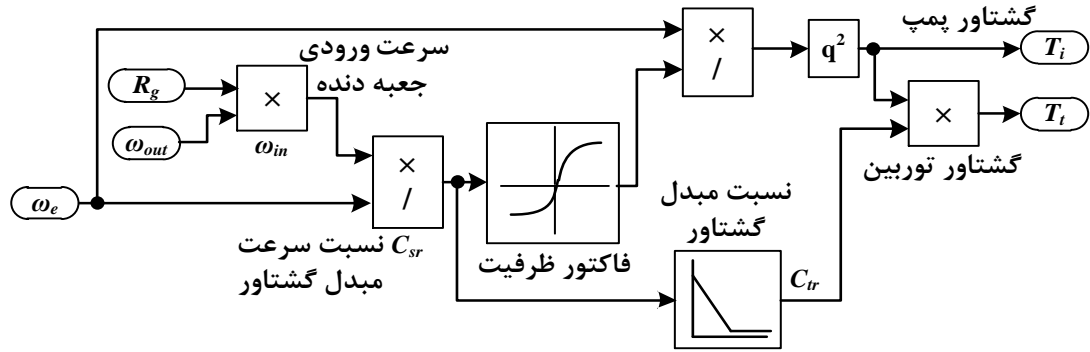
$$T_w = R_g R_{fd} T_t \quad (4)$$



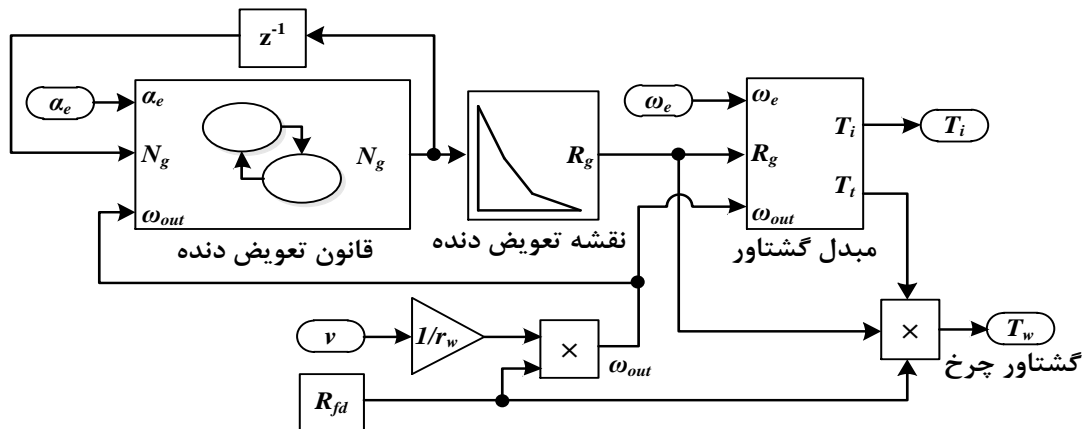
شکل ۱- مدل دینامیک طولی خودرو شامل دینامیک سیستم انتقال قدرت و دینامیک حرکتی خودرو



شکل ۲- مدل دینامیک موتور



شکل ۳- مدل مبدل گشتاور



شکل ۴- مدل سیستم جعبه دنده خودکار

معادلات حاکم بر نیروهایی که به خودرو وارد می‌شوند، شامل نیروی طولی F_x ، نیروی مقاومت غلتشی F_{roll} ، نیروی گرانشی F_{grav} و نیروی درگ آیرودینامیکی F_{aero} در روابط (۵) تا (۸) ارائه شده‌اند.

$$F_x = \frac{(T_w + T_b)}{r_w} \quad (۵)$$

$$F_{aero} = \frac{1}{2} C_d A_f \rho v^2 \quad (۶)$$

$$F_{roll} = fmg \cos(\beta_r) \quad (۷)$$

$$F_{grav} = mg \sin(\beta_r) \quad (۸)$$

در رابطه (۵)، T_b و T_w به ترتیب نشان‌دهنده گشتاورهای تولید شده در سیستم انتقال قدرت و سیستم ترمزی می‌باشند. گشتاور ترمزی متناسب با مقدار فشاری P_b است که در پشت دیسک ترمزی ایجاد می‌شود و با استفاده از رابطه (۹) بدست می‌آید.

$$T_b = K_b P_b \quad (۹)$$

در رابطه فوق، K_b ضریب تبدیل فشار/گشتاور سیستم ترمزی است. در روابط (۶) تا (۸)، C_d ضریب درگ آیرودینامیک خودرو، A_f سطح مقطع مؤثر خودرو، $\rho = 1.206 \text{ kg/m}^3$ چگالی هوا، f ضریب مقاومت غلتشی، β_r زاویه شیب جاده، $g = 9.81 \text{ m/s}^2$ شتاب

گرانشی زمین و m جرم کلی خودرو می‌باشد. با صرف نظر از دینامیک چرخ‌ها، رابطه حاکم بر دینامیک طولی خودرو به شکل رابطه (۱۰) بدست می‌آید.

$$m\dot{v} = F_x - F_{aero} - F_{roll} - F_{grav} \Rightarrow m\dot{v} = F_x - \frac{1}{2}C_d A_f \rho v^2 - fmg \cos(\beta_r) - mg \sin(\beta_r) \quad (10)$$

مطابق رابطه (۱۰) سرعت خودرو در هر لحظه از حل عددی یک معادله دیفرانسیل بدست می‌آید.

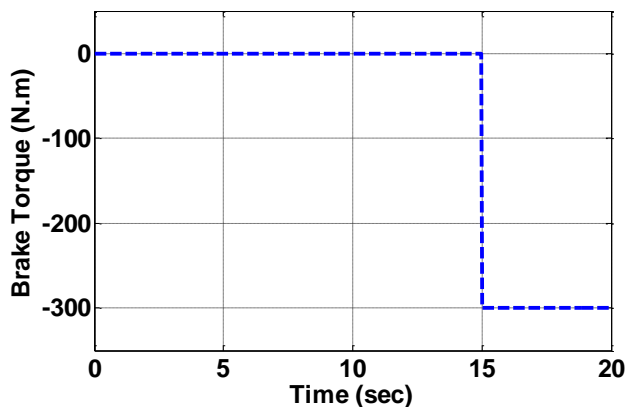
۲-۱- اعتبارسنجی مدل دینامیک طولی خودرو

مدل خودروی نمونه‌ای که در این مطالعه مورد استفاده قرار گرفته است، خودروی سدان با سایز متوسط و جرم اسمی ۱۲۵۰ کیلوگرم می‌باشد. قدرت موتور این خودروی ۵ سرعته، ۷۵ کیلووات و با حداکثر گشتاور ۱۵۵ نیوتن-متر می‌باشد. پارامترهای اسمی این خودروی نمونه از نرم‌افزار CarSim استخراج شده و در جدول (۱) لیست شده‌اند. برای اعتبارسنجی روابط حاکم بر مدل دینامیک طولی خودرو، پاسخ‌های مدل مطالعه فعلی در یک مانور حرکت طولی با پاسخ‌های مدلی که در نمونه برنامه‌ای با عنوان sldemo_autotrans در جعبه‌ابزار Simulink که توسط شرکت MathWorks عرضه شده است مقایسه شده است. در نمونه برنامه sldemo_autotrans مدل دینامیک طولی خودروی سواری با سیستم جعبه دنده خودکار شامل سیستم انتقال قدرت و اجزای اصلی آن شامل موتور، مبدل گشتاور، جعبه دنده خودکار، محورها و چرخ‌ها شبیه‌سازی شده است. در مانور حرکت طولی مورد نظر برای اعتبارسنجی مدل، ورودی‌های متغیر با زمان درصد باز شدگی دریچه گاز و گشتاور ترمزی به مدل دینامیک طولی خودرو اعمال شده‌اند که به ترتیب در شکل‌های (۵) و (۶) نشان داده می‌شوند.

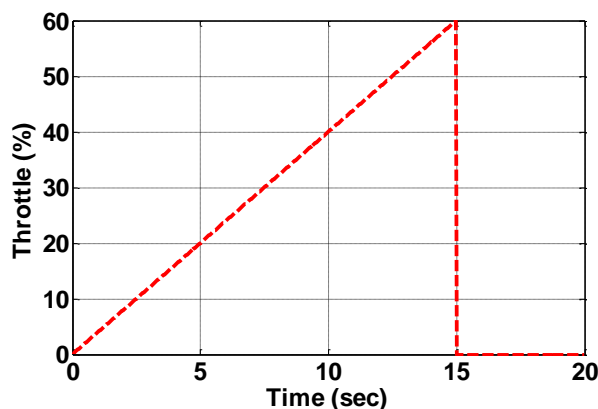
شکل دو منحنی از خروجی‌های خودرو شامل سرعت طولی خودرو و گشتاور موتور در طی مانور آزمون برای مدل شبیه‌سازی شده در این مقاله (Present Model) و مدل مرجع در نرم‌افزار Simulink (Valid Model) به ترتیب در شکل‌های (۷) و (۸) نشان داده شده‌است. جهت مقایسه بهتر نتایج مدل استفاده شده در این مقاله با مدل مرجع sldemo_autotrans، میانگین درصد خطای مربوط به سرعت خودرو و گشتاور موتور به دست می‌آید که در جدول (۲) آورده شده است.

جدول ۱- مقادیر اسمی پارامترهای خودروی نمونه

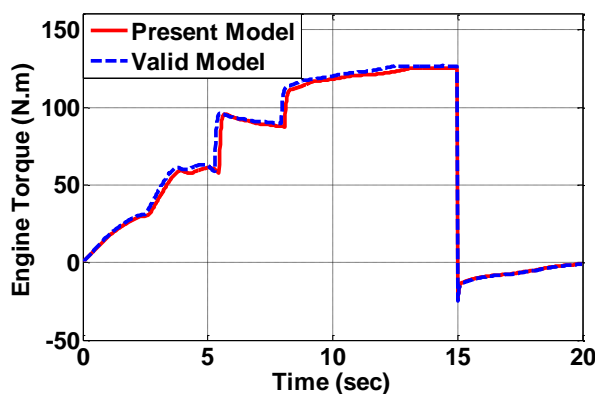
پارامتر	توصیف پارامتر	مقدار
m (kg)	جرم خودرو	۱۲۵۰
r_w (m)	شعاع مؤثر چرخ	۰/۲۹۸
R_g	ضریب دنده	[۵: ۰/۸۴، ۴: ۱/۰۳، ۳: ۱/۳۶، ۲: ۲/۱۲، ۱: ۳/۷۸]
R_{fd}	ضریب دیفرانسیل	۴/۱
K_b (N.m/Mpa)	ثابت تبدیل فشار- گشتاور	۵۴۰
I_{ei} (kg.m ²)	مجموع لختی اینرسی موتور و پمپ	۰/۱۶
$C_{d0}A_{f0}$ (m ²)	ضریب درگ کلی آیرودینامیک	۰/۴۸
f	ضریب مقاومت غلتشی	۰/۰۱۵



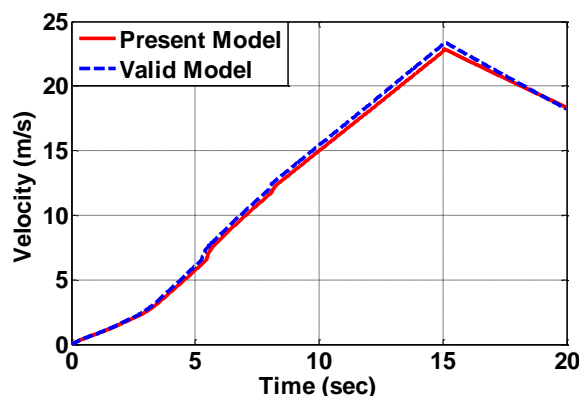
شکل ۶- منحنی گشتاور ترمزی اعمال شده



شکل ۵- منحنی درصد بازشدگی دریچه گاز



شکل ۸- منحنی گشتاور موتور هر دو مدل



شکل ۷- منحنی سرعت طولی خودرو هر دو مدل

جدول ۲- میانگین درصد خطای خروجی‌های مدل خودروی این مقاله در مقایسه با مدل معتبر sldemo_autotrans

گشتاور موتور	سرعت خودرو	خروجی مدل
۳/۵۹ %	۲/۸۳ %	میانگین درصد خطای نتایج مدل‌ها

مطابق نتایج به دست آمده از جدول (۲)، مدل دینامیک طولی خودروی سواری که در این بخش شبیه‌سازی شده است، دارای اعتبار کافی می‌باشد؛ لذا امکان استفاده از این مدل در شبیه‌سازی‌های بعدی وجود دارد.

۳- طراحی تخمین‌گر فرم درجه دوم خطی به صورت زمان کوتاه

در این بخش، نحوه طراحی تخمین‌گر فرم درجه دوم خطی به صورت زمان کوتاه (STLQF) که الگوریتمی جدید برای تخمین پارامترهای متغیر با زمان سیستم دینامیکی می‌باشد، بر اساس روابط مرجع [۲۲] تشریح می‌گردد. سپس از این تخمین‌گر جهت تخمین پارامترهای جرم خودرو، زاویه شیب جاده و ضریب درگ کلی آیرودینامیک خودرو در آزمونی مبتنی بر داده‌های شبیه‌سازی شده‌ی رایانه‌ای و تجربی استفاده می‌شود. برخی از ویژگی‌های برجسته‌ی الگوریتم تخمین‌گر STLQF به شرح زیر می‌باشند:

- بار محاسباتی مناسب
- سرعت همگرایی بالا، به دلیل استفاده از تحلیل‌های به فرم زمان کوتاه
- نآریب بودن تخمین، به دلیل استفاده از مزایای تخمین‌گر با فرم درجه دوم خطی

- قابلیت تخمین پارامترها به صورت بلادرنگ و با تأخیر زمانی اندک
 - امکان تخمین چندین پارامتر به صورت هم‌زمان
 - عدم نیاز به تعریف فرضیات ساده‌کننده بر روی محدود تغییرات پارامترهای سیستم
- لذا استفاده از تخمین گر مدل - مبنای STLQF باعث می‌شود که امکان تخمین پارامترهایی نظیر زاویه شیب جاده بدون استفاده از سنسورهای گران‌قیمت GPS و LIDAR فراهم شود.
- تخمین گر STLQF روشی است که بر پایه کمینه‌سازی یک تابع هزینه با فرم درجه دوم خطی است و شامل ترکیبی خطی از خطاهای تخمین می‌باشد. تابع هزینه STLQF در فضای زمان گسسته به شکل رابطه (۱۱) تعریف می‌شود.

$$J = \frac{1}{2} [w[n] \quad w[n-1] \quad w[n-2]] \mathbf{A} \begin{bmatrix} w[n] \\ w[n-1] \\ w[n-2] \end{bmatrix} \quad (11)$$

در رابطه (۱۱) n نشان‌دهنده حالت سیستم در نمونه‌ی n ام و \mathbf{A} یک ماتریس حقیقی، متقارن و مثبت-معین است:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{bmatrix} \quad (12)$$

با جای‌گذاری رابطه (۱۲) در رابطه (۱۱)، تابع هزینه به شکل رابطه (۱۳) بدست می‌آید:

$$J = \frac{1}{2} \{a_{11}w^2[n] + a_{22}w^2[n-1] + a_{33}w^2[n-2] + 2a_{12}w[n]w[n-1] + 2a_{13}w[n]w[n-2] + 2a_{23}w[n-1]w[n-2]\} \quad (13)$$

در رابطه (۱۳)، w خطای تعمیم‌یافته مدل‌سازی می‌باشد که شامل اطلاعات بیشتری از خطای تخمین $e[n]$ است و مطابق رابطه‌ی (۱۴) تعریف می‌شود:

$$w[n] = e[n] + \chi_1 \left[\frac{e[n] - e[n-1]}{T} \right] + \chi_2 \left[\frac{e[n] - 2e[n-1] + e[n-2]}{T^2} \right] \quad (14)$$

در رابطه (۱۴)، χ_1 و χ_2 ضرایب دلخواه هستند و T گام زمانی^۱ است. با ساده‌سازی، رابطه (۱۴) به شکل رابطه (۱۵) بازنویسی می‌شود:

$$w[n] = \alpha e[n] + \beta e[n-1] + \gamma e[n-2] \quad (15)$$

پارامترهای α ، β و γ به شکل رابطه (۱۶) تعریف می‌شوند:

$$\alpha = 1 + \frac{1}{T} + \frac{1}{T^2}, \beta = -\frac{1}{T} + \frac{2}{T^2}, \gamma = \frac{1}{T^2} \quad (16)$$

خطای تخمین $e[n]$ در رابطه (۱۵) به شکل رابطه $e[n] = z[n] - \Phi[n]^T \Theta[n]$ تعریف می‌شود که در این رابطه $z[n]$ خروجی سیستم است. $\Phi[n]$ و $\Theta[n]$ نیز به ترتیب بردار پارامترهای تبدیل‌یافته و بردار رگرسورهای تبدیل‌یافته سیستم می‌باشند. بردار رگرسورهای اصلی سیستم $\Psi_{L \times 1}[n]$ که در رابطه (۱۷) تعریف شده است، با استفاده از رابطه (۱۸) به بردار $\Phi_{m \times 1}[n]$ تصویر می‌شود. همچنین، بردار پارامترهای اصلی سیستم $\Theta_{L \times 1}[n]$ که در حالت کلی به شکل متغیر با زمان می‌باشد، طبق رابطه (۱۹) تعریف می‌شود.

¹ Sample Time

$$\Psi_{L \times 1}[n] = \begin{bmatrix} \psi_1[n] \\ \psi_2[n] \\ \vdots \\ \psi_L[n] \end{bmatrix} \quad (17)$$

$$\Phi_{m \times 1}[n] = [\psi_1[n] \mathbf{V}_{fcn}^r, \dots, \psi_L[n] \mathbf{V}_{fcn}^r] = [\varphi_1[n], \varphi_2[n], \dots, \varphi_{m-1}[n], \varphi_m[n]] \quad (18)$$

$$m = L \times r$$

$$\Theta_{L \times 1}[n] = \begin{bmatrix} \theta_1[n] \\ \theta_2[n] \\ \vdots \\ \theta_L[n] \end{bmatrix} \quad (19)$$

در رابطه (۱۷)، L تعداد پارامترهای سیستم دینامیکی را نشان می‌دهد و تابع \mathbf{V}_{fcn}^r در رابطه (۱۸) پارامترهای سیستم را از فضای متغیر با زمان به پارامترهایی با مقدار ثابت نگاشت می‌کند:

$$\mathbf{V}_{fcn}^r = [fcn^r, fcn^{r-1}, \dots, fcn^1, 1.0]^T \quad (20)$$

با اعمال تابع برداری \mathbf{V}_{fcn}^r بر روی رگرسیون‌های سیستم، پارامتر متغیر با زمان $\theta_k[n]$ با یک تابع چندجمله‌ای از درجه r ام و به فرم $\theta_k[n] \approx p_{kr} fcn^r + p_{k(r-1)} fcn^{r-1} + \dots + p_{k1} fcn^1 + p_{k0}$ تخمین زده می‌شود که $p_{kj}, 0 \leq j \leq r$ ضرایب چندجمله‌ای و fcn تابع پیوسته‌ای از زمان است. به این ترتیب بردار پارامترهای جدید مطابق روابط (۲۱) و (۲۲) تعریف می‌شوند:

$$\boldsymbol{\theta} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_L]_{(L \times (r+1)) \times 1} \quad (21)$$

$$\mathbf{p}_k = [p_{kr}, p_{k(r-1)}, \dots, p_{k0}]_{(r+1) \times 1} \quad (22)$$

$$k = 1, 2, \dots, L$$

با تعریف فضایی جدید از رگرسیون‌های سیستم، مساله تخمین پارامترهای اصلی سیستم با کمینه‌سازی تابع هزینه J مطابق رابطه (۱۳) قابل حل است. به این منظور پنجره لغزان زمان کوتاه با طول $2 \times WL + 1$ نمونه روی ماتریس رگرسیون‌های Φ لغزانده می‌شود و پاسخ مساله $\frac{\partial J}{\partial \boldsymbol{\theta}} = 0$ به ازای تمامی پارامترهای سیستم بدست می‌آید. با حل معادله جبری $\frac{\partial J}{\partial \boldsymbol{\theta}} = 0$ ، معادلات خطی-جبری به شکل رابطه (۲۳) بدست می‌آیند که از حل این دستگاه معادلاتی، بردار پارامترهای سیستم تخمین زده می‌شوند. یعنی با اعمال روش فوق، با لغزاندن پنجره از نمونه $2 \times WL - n$ تا نمونه n ام، مقدار پارامتر در نمونه $n - WL$ محاسبه می‌شود. با توجه به ریاضیات حاکم بر الگوریتم STLQF، این تخمین‌گر پارامترهایی را که دارای تغییرات پیوسته و به صورت متغیر با زمان می‌باشند با دقت مناسبی شناسایی می‌کند.

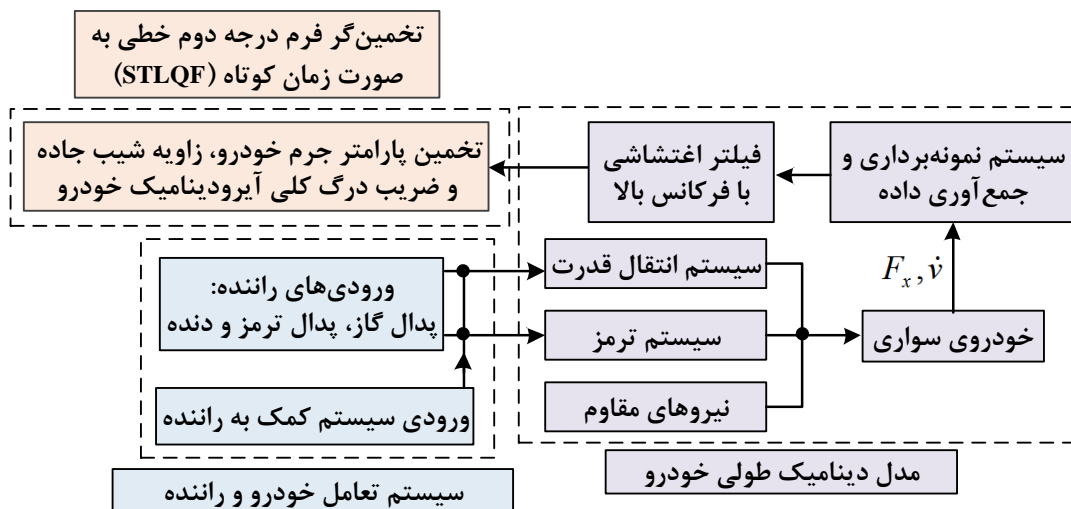
(۲۳)

$$\sum_{n=4}^N \{ \{ a_{11}(\alpha\varphi[n] + \beta\varphi[n-1] + \gamma\varphi[n-2])(\alpha\varphi^T[n] + \beta\varphi^T[n-1] + \gamma\varphi^T[n-2]) + a_{22}(\alpha\varphi[n-1] + \beta\varphi[n-2] + \gamma\varphi[n-3])(\alpha\varphi^T[n-1] + \beta\varphi^T[n-2] + \gamma\varphi^T[n-3]) + a_{33}(\alpha\varphi[n-2] + \beta\varphi[n-3] + \gamma\varphi[n-4])(\alpha\varphi^T[n-2] + \beta\varphi^T[n-3] + \gamma\varphi^T[n-4]) + a_{12}[(\alpha\varphi[n] + \beta\varphi[n-1] + \gamma\varphi[n-2])(\alpha\varphi^T[n-1] + \beta\varphi^T[n-2] + \gamma\varphi^T[n-3]) + (\alpha\varphi[n-1] + \beta\varphi[n-2] + \gamma\varphi[n-3])(\alpha\varphi^T[n] + \beta\varphi^T[n-1] + \gamma\varphi^T[n-2])] + a_{13}[(\alpha\varphi[n] + \beta\varphi[n-1] + \gamma\varphi[n-2])(\alpha\varphi^T[n-2] + \beta\varphi^T[n-3] + \gamma\varphi^T[n-4]) + (\alpha\varphi[n-2] + \beta\varphi[n-3] + \gamma\varphi[n-4])(\alpha\varphi^T[n] + \beta\varphi^T[n-1] + \gamma\varphi^T[n-2])] + a_{23}[(\alpha\varphi[n-1] + \beta\varphi[n-2] + \gamma\varphi[n-3])(\alpha\varphi^T[n-2] + \beta\varphi^T[n-3] + \gamma\varphi^T[n-4]) + (\alpha\varphi[n-2] + \beta\varphi[n-3] + \gamma\varphi[n-4])(\alpha\varphi^T[n-1] + \beta\varphi^T[n-2] + \gamma\varphi^T[n-3])] \} \} = \sum_{n=4}^N \{ \{ a_{11}(\alpha z[n] + \beta z[n-1] + \gamma z[n-2])(\alpha\varphi[n] + \beta\varphi[n-1] + \gamma\varphi[n-2]) + a_{22}(\alpha z[n-1] + \beta z[n-2] + \gamma z[n-3])(\alpha\varphi[n-1] + \beta\varphi[n-2] + \gamma\varphi[n-3]) + a_{33}(\alpha z[n-2] + \beta z[n-3] + \gamma z[n-4])(\alpha\varphi[n-2] + \beta\varphi[n-3] + \gamma\varphi[n-4]) + a_{12}[(\alpha z[n] + \beta z[n-1] + \gamma z[n-2])(\alpha\varphi[n-1] + \beta\varphi[n-2] + \gamma\varphi[n-3]) + (\alpha z[n-1] + \beta z[n-2] + \gamma z[n-3])(\alpha\varphi[n] + \beta\varphi[n-1] + \gamma\varphi[n-2])] + a_{13}[(\alpha z[n] + \beta z[n-1] + \gamma z[n-2])(\alpha\varphi[n-2] + \beta\varphi[n-3] + \gamma\varphi[n-4]) + (\alpha z[n-2] + \beta z[n-3] + \gamma z[n-4])(\alpha\varphi[n] + \beta\varphi[n-1] + \gamma\varphi[n-2])] + a_{23}[(\alpha z[n-1] + \beta z[n-2] + \gamma z[n-3])(\alpha\varphi[n-2] + \beta\varphi[n-3] + \gamma\varphi[n-4]) + (\alpha z[n-2] + \beta z[n-3] + \gamma z[n-4])(\alpha\varphi[n-1] + \beta\varphi[n-2] + \gamma\varphi[n-3])] \} \}$$

۳-۱- توسعه تخمین گر STLQF جهت تخمین پارامترهای دینامیک طولی خودرو

دیگرام بلوکی سیستم تعامل خودرو و راننده همراه با تخمین گر STLQF و مدل دینامیک طولی خودرو در شکل (۹) نمایش داده شده است. مطابق ساختار ارائه شده، زاویه دریچه گاز، فشار ترمزی و شماره دنده مطلوب توسط راننده یا سیستم کمک به راننده به خودرو اعمال می‌شود. این ورودی‌ها در مدل دینامیک طولی خودرو که از سه زیرسیستم اصلی انتقال قدرت، ترمز و مدل نیروهای مقاوم تشکیل شده است، نیروی طولی F_x مؤثر جهت حرکت خودرو با شتابی به بزرگی \dot{v} را منجر می‌شوند. دو متغیر F_x و \dot{v} توسط سیستم نمونه‌برداری مناسب ثبت و ضبط می‌شوند و در حالت عملیاتی پس از فیلتر شدن در اختیار الگوریتم STLQF قرار می‌گیرند تا پارامترهای جرم خودرو، زاویه شیب جاده و ضریب درگ کلی آیرودینامیک خودرو تخمین زده شوند.

به‌منظور پیاده‌سازی الگوریتم STLQF بر روی مدل دینامیک طولی خودرو، یک مدل رگر سور خطی مورد نیاز است. جهت تخمین پارامترهای جرم خودرو، زاویه شیب جاده و ضریب درگ کلی آیرودینامیک خودرو، و اعمال تغییر متغیر $f = \tan(\varepsilon)$ ، رابطه (۱۰) به فرم رابطه (۲۴) بازنویسی می‌شود:



شکل ۹- دیگرام بلوکی سیستم تعامل خودرو و راننده همراه با تخمین گر STLQF و مدل دینامیک طولی خودرو

$$m(\dot{v}) + C_d A_f \left(\frac{1}{2} \rho v^2 \right) + m \sin(\varepsilon + \beta_r) \left(\frac{g}{\cos(\varepsilon)} \right) = F_x \quad (24)$$

با استفاده از رابطه (۲۴)، بردار پارامترهای سیستم به شکل رابطه (۲۵) تعریف می‌شود:

$$\Theta[n] = \begin{bmatrix} m[n] \\ C_d A_f [n] \\ m[n] \sin(\varepsilon + \beta_r)[n] \end{bmatrix} \quad (25)$$

متغیرهای رگرسور خطی و خروجی سیستم در نمونه‌ی n ام در روابط (۲۶) و (۲۷) تعریف می‌شوند:

$$\begin{cases} \Phi_1[n] = \dot{v}[n] \\ \Phi_2[n] = \frac{1}{2} \rho v^2 [n] \\ \Phi_3[n] = \frac{g}{\cos(\varepsilon)} [n] \end{cases} \quad (26)$$

$$z[n] = F_x [n] + \sigma_N \cdot S_N(t) \quad (27)$$

در رابطه (۲۷)، $S_N(t)$ متغیری برای شبیه‌سازی نویز با توزیع نرمال و میانگین صفر یا همان نویز سفید گوسی^۱ می‌باشد که دارای انحراف معیار σ_N است و در فرایند اندازه‌گیری رگرسور خروجی سیستم ایجاد می‌شود. به این ترتیب پارامترهای مجهول سیستم شامل $m[n]$ ، $C_d A_f [n]$ و $\beta_r [n]$ که در شبیه‌سازی‌های این مطالعه به ترتیب با نمادهای $\theta_1 [n]$ ، $\theta_2 [n]$ و $\theta_3 [n]$ نمایش داده می‌شوند، با اعمال الگوریتم STLQF و با استفاده از رگرسورهای اندازه‌گیری شده‌ی سیستم قابل تخمین زدن هستند.

۴- شبیه‌سازی الگوریتم تخمین گر STLQF

در این بخش، عملکرد الگوریتم STLQF جهت تخمین پارامترهای جرم خودرو، ضریب درگ کلی آیرودینامیک خودرو و زاویه شیب جاده مورد بررسی قرار می‌گیرد. نتایج تخمین بدست آمده از الگوریتم STLQF با نتایج حاصل از پیاده‌سازی تخمین گر مرسوم حداقل مربعات بازگشتی (RLS) که از تئوری بیشترین احتمال ممکن^۲ استفاده می‌کند، مقایسه می‌شود. لازم به ذکر است که سرعت همگرایی بالا از جمله ویژگی‌های الگوریتم RLS می‌باشد که کارکرد آن را در کاربردهای بلادرنگ توسعه داده است [۲۵].

در طول شبیه‌سازی، مدل خودروی نمونه با پارامترهای اسمی لیست شده در جدول (۱) فرض می‌شود. داده‌های زاویه شیب جاده نیز با استفاده از سیستم موقعیت‌یاب جهانی (GPS) از یکی از خیابان‌های مرتفع شهر تهران جمع‌آوری می‌شوند که در ادامه روند این کار به تفصیل تشریح می‌شود.

¹ White Gaussian Noise (WGN)

² Maximum Likelihood Theory

۴-۱- جمع‌آوری داده‌های زاویه شیب جاده به صورت تجربی

امروزه با پیشرفت فناوری و گسترش زیرساخت‌های ارتباطی و مخابراتی، بسیاری از تلفن‌های همراه مجهز به سامانه GPS شده‌اند. سخت افزار این سامانه منظومه‌ای از ایستگاه‌های زمینی و ۲۴ ماهواره را شامل می‌شود. این ماهواره‌ها زمین را دور می‌زنند و نرم‌افزار این سامانه با انجام محاسبات ریاضی امکان سنجش موقعیت مکانی، موقعیت عرضی جغرافیایی، سرعت حرکت و جهت حرکت جسم مجهز به این سامانه را فراهم می‌کند. نرم‌افزار Android-speedometer، یکی از نرم‌افزارهای محبوب تلفن همراه با سیستم عامل Android می‌باشد که امکان قرائت داده‌های GPS را در گوشی تلفن همراه فراهم می‌کند. این داده‌ها شامل مدت زمان سفر، مسافت طی شده، میانگین سرعت در طی سفر، حداکثر سرعت، نقشه راه سفر، منحنی سرعت بر حسب زمان و منحنی ارتفاع بر حسب زمان با گام زمانی ۱، ۵، ۱۰ و ۱۵ ثانیه جهت نمونه‌برداری می‌باشد. در این مطالعه، از نرم‌افزار Android-speedometer، جهت جمع‌آوری داده‌های زمینی بخشی از خیابان سعادت آباد- خیابان بهرود استفاده شده است. روند انجام این کار به این شرح می‌باشد:

یک گوشی تلفن همراه مجهز به سامانه GPS و نرم‌افزار Android-speedometer به داشبورد خودروی سواری ۲۰۶ متصل می‌شود؛ به نحوی که امکان لرزش و حرکت نداشته باشد. خودروی سواری از حالت سکون شروع به حرکت می‌کند و مسیری با طول تقریبی ۱/۳ کیلومتر را که در بخشی از خیابان سعادت آباد- خیابان بهرود واقع بوده و دارای شیب‌هایی با زوایای تند می‌باشد را می‌پیماید. مطابق شکل (۱۰) که از Google-Map استخراج شده است، مبدا شروع حرکت شهرک بهاران، ابتدای پارک ژوراسیک (نماد نارنجی رنگ در نقشه) و مقصد حرکت بلوار پیام واقع در شرق پارک پرواز می‌باشد (نماد قرمز رنگ در نقشه). منحنی‌های سرعت خودرو بر حسب زمان، ارتفاع خودرو از سطح آزاد دریا بر حسب زمان، مسافت طی شده بر حسب زمان و ارتفاع خودرو از سطح آزاد دریا بر حسب مسافت طی شده که از اطلاعات سامانه GPS استخراج شده‌اند، به ترتیب در شکل‌های (۱۱) تا (۱۴) نمایش داده می‌شوند. گام زمانی اطلاعات ۱ ثانیه می‌باشد و داده‌های مربوط به نقاط مجهول با درون‌یابی خطی تخمین شده است.

لازم به ذکر است که مطابق شکل (۱۱) داده خام سرعت خودرو بر حسب زمان که از GPS استخراج می‌شود دارای تعداد زیادی ناپیوستگی است. داده سرعت خودرو با استفاده از فیلتر میانگین متحرک^۱ که طبق رابطه (۲۸) طراحی شده است، فیلتر می‌شود و در شبیه‌سازی‌های بعدی از اطلاعات فیلتر شده استفاده می‌گردد. در رابطه (۲۸)، u ، y و k به ترتیب داده‌ی دارای ناپیوستگی، داده‌ی فیلتر شده و تعداد داده‌های استفاده شده در پنجره میانگین متحرک می‌باشند [۲۶]. در این مقاله، از طول پنجره ۴ برای فیلتر کردن داده‌ها استفاده شده است.

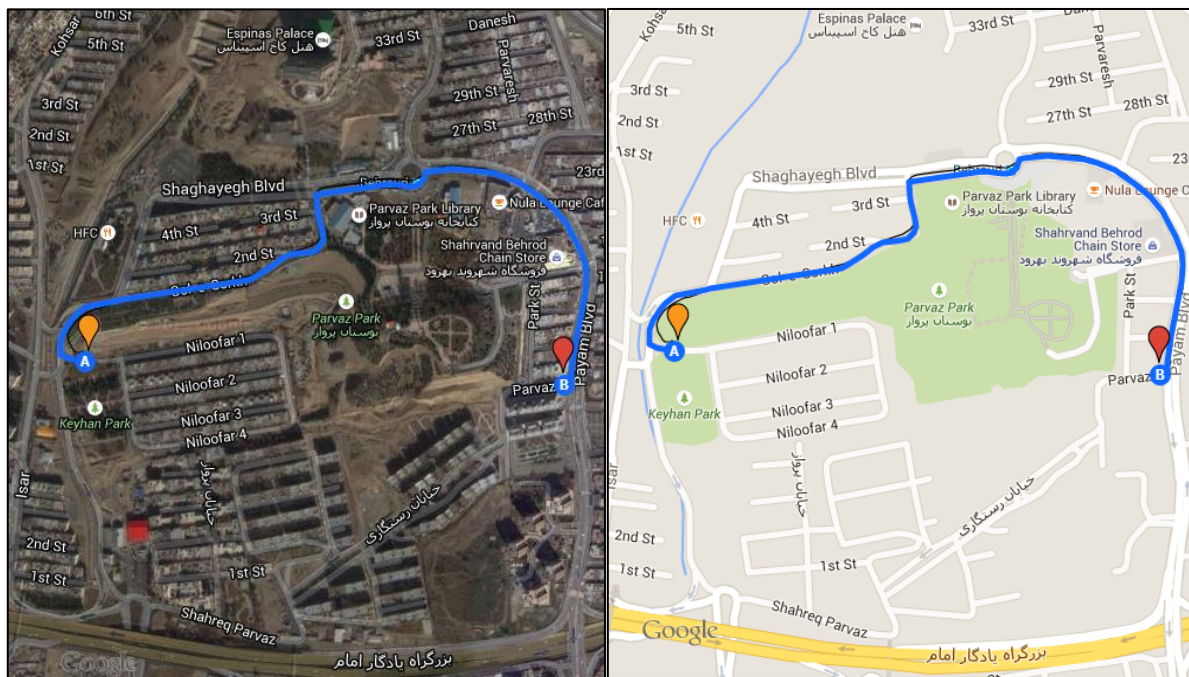
$$y[n] = \frac{1}{k} \sum_{i=0}^{k-1} u[n-i] \quad (28)$$

با استفاده از اطلاعات شکل (۱۴) و استفاده از رابطه (۲۹)، مقدار زاویه شیب جاده در هر نقطه بدست می‌آید که در شکل (۱۵) نمایش داده شده است. در رابطه (۲۹)، $H[i]$ و $X[i]$ به ترتیب ارتفاع و موقعیت محلی نقطه i ام نسبت به مرجع می‌باشند.

¹ Moving Average Filter

$$\beta[i] = \tan^{-1} \left(\frac{H[i+1] - H[i]}{X[i+1] - X[i]} \right) \times \frac{180}{\pi} \quad (29)$$

همان طور که در شکل (۱۵) مشاهده می شود، مسیر مورد نظر دارای پستی- بلندی های زیاد، با شیب تند و موضعی به خصوص در ابتدای شروع حرکت می باشد. با جمع آوری این داده های تجربی، امکان ارزیابی الگوریتم STLQF میسر می شود که این روند در بخش بعد تشریح می گردد.



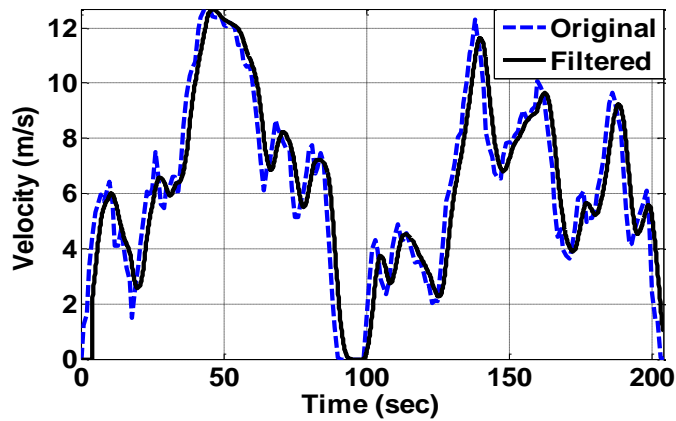
(الف)



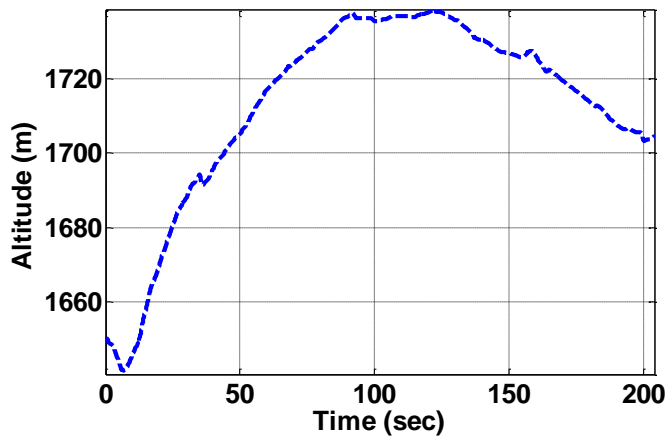
(ب)

شکل ۱۰- الف) مسیر حرکت خودرو که از نرم افزار Google-Map استخراج شده است،

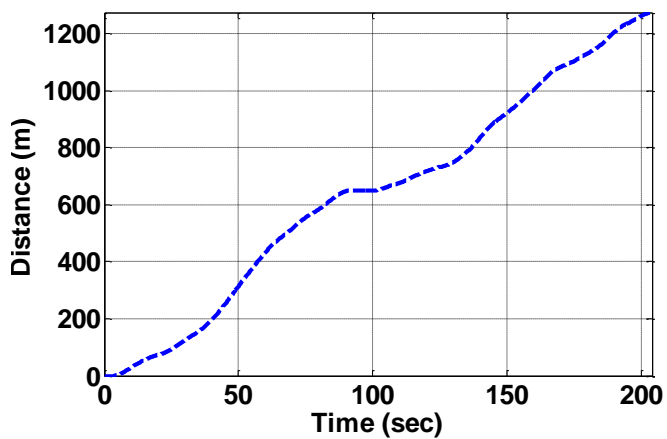
ب) نمایی از خیابان بهرود- پارک ژوراسیک



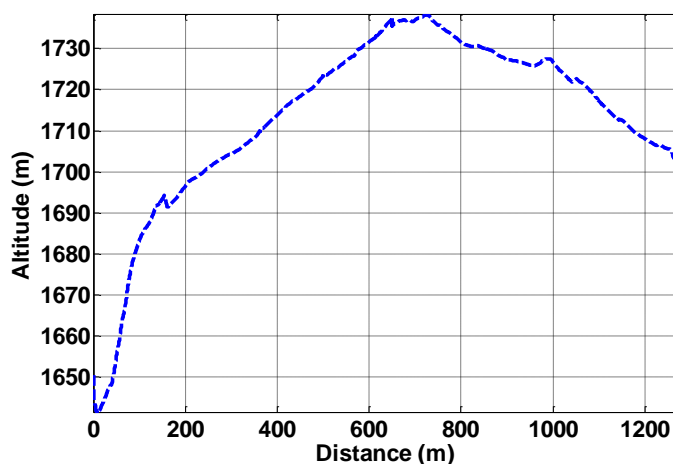
شکل ۱۱- سرعت حرکت خودرو بر حسب زمان که از سامانه GPS استخراج شده است



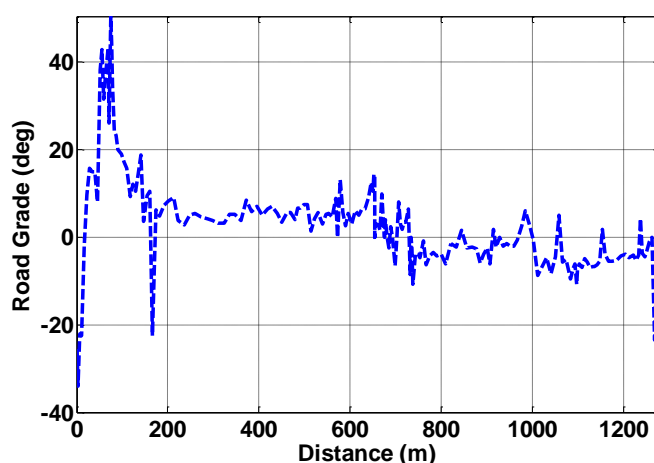
شکل ۱۲- تغییر ارتفاع خودرو از سطح آزاد دریا بر حسب زمان که از اطلاعات سامانه GPS استخراج شده است



شکل ۱۳- مسافت طی شده بر حسب زمان که از سامانه GPS استخراج شده است



شکل ۱۴- ارتفاع خودرو از سطح آزاد دریا بر حسب مسافت طی شده که از سامانه GPS استخراج شده است



شکل ۱۵- مقدار زاویه شیب جاده بر حسب زمان که از پردازش اطلاعات سامانه GPS بدست آمده است

۴-۲- نتایج تخمین الگوریتم STLQF بر مبنای داده‌های شبیه‌سازی شده رایانه‌ای و تجربی

در بخش ۴-۱، روند بدست آوردن زاویه شیب جاده به صورت نقطه به نقطه در یکی از خیابان‌های مرتفع شهر تهران تشریح شد. در این بخش، نحوه تخمین مقادیر زاویه شیب این جاده به همراه پارامترهای مؤثر بر دینامیک طولی خودرو با استفاده از الگوریتم STLQF ارزیابی می‌شود. به این منظور فرض می‌شود که خودروی سواری با مشخصاتی مطابق جدول (۱) در طی یک فرایند شبیه‌سازی شده رایانه‌ای، بخشی از مسیر خیابان سعادت آباد- خیابان بهرود را به مدت ۷۵ ثانیه طی می‌کند.

در طی این فرایند، ورودی گاز مطابق شکل (۱۶) به مدل دینامیک طولی خودرو، که نحوه مدل‌سازی آن در روابط (۱) تا (۱۰) این مقاله ارائه شده بود، داده می‌شود و خودرو مسیری را با زاویه شیب جاده مطابق آنچه در شکل (۱۷) نشان داده شده است می‌پیماید. به این ترتیب، با اعمال ورودی گاز در هر لحظه، مقدار نیروی طولی خودرو F_x با استفاده از روابط (۱) تا (۵) به دست می‌آید.

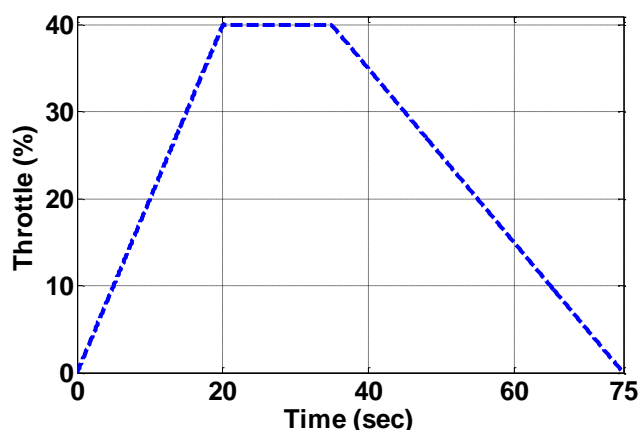
نیروی مقاومت خارجی نیز که به‌طور اخص ناشی از حرکت خودرو بر روی جاده با زاویه شیب متغیر با زمان همراه با نیروی مقاومت غلتشی و نیروی درگ آیرودینامیکی می‌باشد با استفاده از روابط (۶) تا (۸) بدست

می‌آید. سرعت خودروی نمونه نیز در هر لحظه با حل عددی رابطه (۱۰) بر حسب متغیر \dot{v} و با استفاده از روش آدامز-بشفورث^۱ که در رابطه (۳۰) آمده است نتیجه می‌شود. در شکل (۱۸) منحنی سرعت خودرو در طول سفر مشاهده می‌شود. در این مطالعه، مقدار گام زمانی در طول شبیه‌سازی $T = 0.01$ ثانیه انتخاب شده است.

$$v_n = v_{n-1} + \frac{3}{2} \times T \times \dot{v}_{n-1} - \frac{1}{2} \times T \times \dot{v}_{n-2} \quad (30)$$

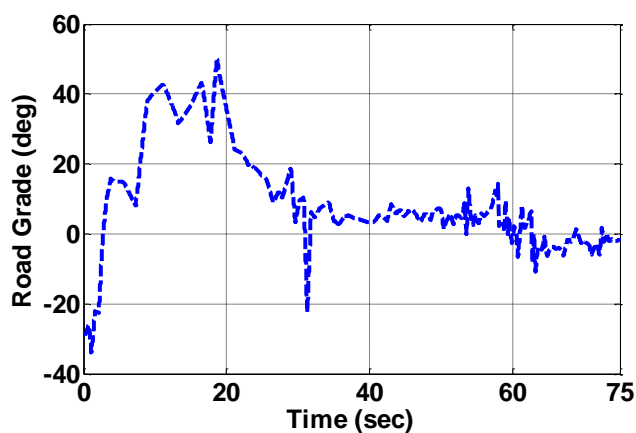
در این مطالعه به منظور تخمین پارامترهای مؤثر بر حرکت طولی خودرو، تابع پیوسته متغیر با زمان f_{cn} در الگوریتم STLQF به شکل سینوسی $\sin(t)$ انتخاب می‌شود و سایر پارامترهای این الگوریتم در جدول (۳) لیست شده‌اند. مطابق جدول (۳)، درجه تابع برداری \mathbf{V}_{fcn}^r برابر یک ($r = 1$) لحاظ شده است تا بار محاسباتی تخمین‌گر حداقل شود. لذا پارامتر متغیر زمان $\theta_k[n]$ به فرم درجه یک و متغیر با زمان به صورت چندجمله‌ای سینوسی تعریف می‌شود: $\theta_k[n] \approx p_{k1} \sin(t) + p_{k0}$, $k = 1, 2, 3$. مقدار انحراف معیار نویز سفید σ_N نیز در رابطه (۲۷) برابر 0.0001 در نظر گرفته شده است. برای طراحی الگوریتم RLS از مرجع [۱۴] استفاده شده است و ضریب فراموشی این الگوریتم بنا بر پیشنهاد همین مرجع برابر 0.95 لحاظ شده است.

با استفاده از رگرسیون‌های ورودی و خروجی مدل دینامیک خودرو، مطابق روابط (۲۴) و (۲۷)، و با پیاده‌سازی الگوریتم‌های تخمین‌گر STLQF و RLS، پارامترهای جرم خودرو، ضریب درگ کلی آیرودینامیک خودرو و زاویه شیب جاده که مقادیر واقعی آن‌ها به ترتیب در جدول (۱) و شکل (۱۷) نشان داده شده بود تخمین زده می‌شوند که نتایج تخمین‌ها در شکل‌های (۱۹) تا (۲۱) نشان داده شده‌اند. الگوریتم STLQF 0.05 ثانیه در تخمین پارامترها تأخیر دارد، چرا که مقدار WL برابر ۵ و گام زمانی $T = 0.01$ ثانیه فرض شده است.

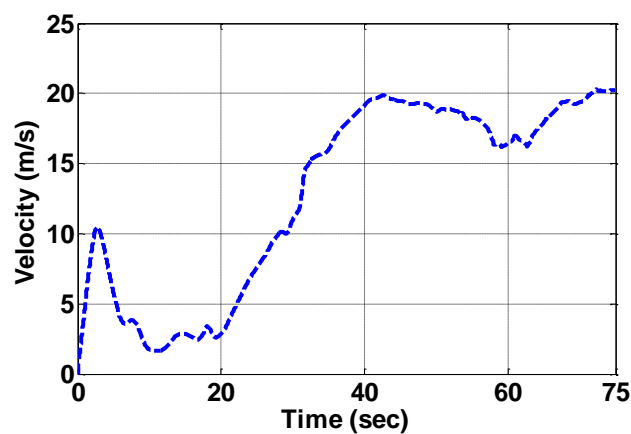


شکل ۱۶- درصد بازشدگی دریچه گاز خودروی سواری بر حسب زمان در روند شبیه‌سازی با داده‌های تجربی جاده

¹ Adams-Bashforth



شکل ۱۷- زاویه شیب جاده بر حسب زمان در روند شبیه‌سازی با داده‌های تجربی جاده

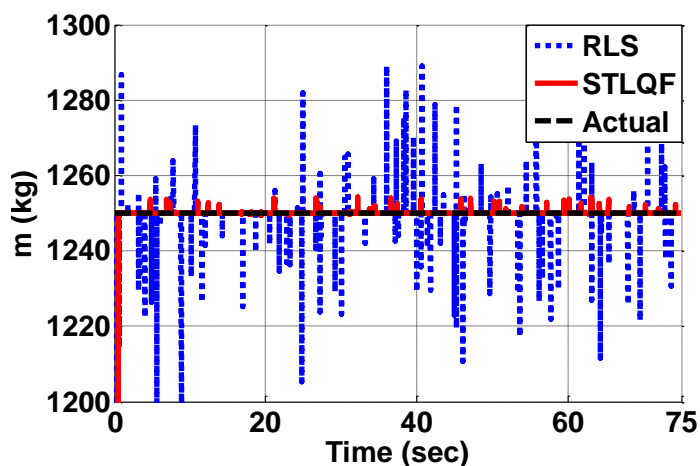


شکل ۱۸- سرعت خودروی سواری بر حسب زمان در روند شبیه‌سازی با داده‌های تجربی جاده

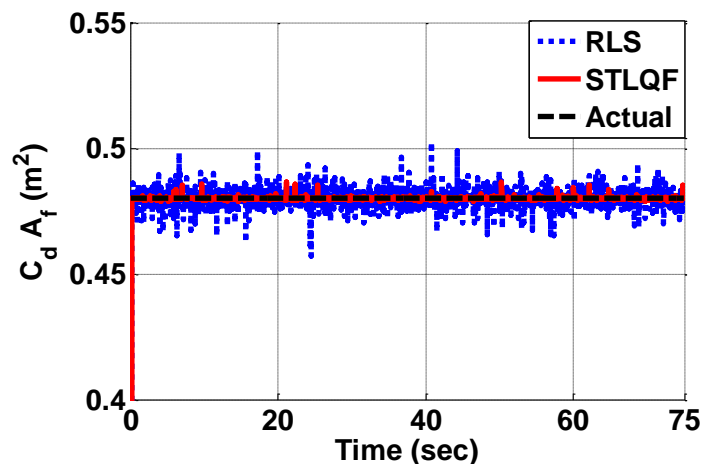
جدول ۳- پارامترهای الگوریتم STLQF

مقدار	توصیف پارامتر	پارامتر
۱	درجه تابع برداری V_{fen}^r	r
۱۱	طول پنجره زمان کوتاه	$2 \times WL + 1$
۰/۰۱	گام زمانی	T (sec)
+۵/۰۰	درایه‌های ماتریس متقارن و مثبت-معین A	a_{11}
+۳/۰۰		a_{12}
+۲/۰۰		a_{13}
+۵/۰۰		a_{22}
+۳/۰۰		a_{23}
+۵/۰۰		a_{33}

جهت ارزیابی دقت الگوریتم‌های تخمین گر RLS و STLQF در تخمین پارامترهای جرم خودرو، زاویه شیب جاده و ضریب درگ کلی آیرودینامیک خودرو، از سه معیار میانگین قدر مطلق خطا^۱، جذر میانگین مربعات خطا^۲ و مجموع مربعات خطا^۳ که نحوه محاسبه آنها به ترتیب در روابط (۳۱) تا (۳۳) آورده شده است استفاده می‌شود. در این روابط، x_i متغیر واقعی و \hat{x}_i متغیر تخمینی یا خروجی تخمین گر می‌باشد. نیز تعداد داده‌های تست را نشان می‌دهد. مقدار معیارهای خطای MAE، RMSE و SSE برای هر دو الگوریتم تخمین گر طراحی شده در جدول (۴) لیست شده است. لازم به ذکر است که در محاسبه مقدار تمامی معیارهای خطا، از نتایج تخمین در ۵ ثانیه اولیه شبیه‌سازی صرف نظر شده است.



شکل ۱۹- تخمین جرم خودرو با الگوریتم‌های تخمین گر RLS و STLQF

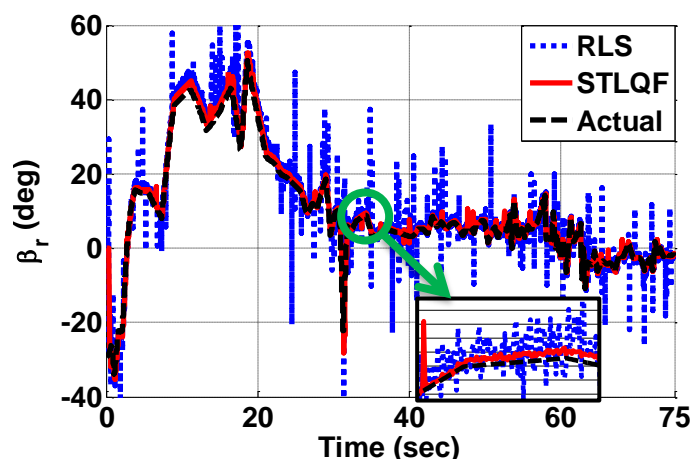


شکل ۲۰- تخمین ضریب درگ کلی آیرودینامیک خودرو با الگوریتم‌های تخمین گر RLS و STLQF

¹ Mean Absolute Errors (MAE)

² Root Mean Square Errors (RMSE)

³ Sum of Square Errors (SSE)



شکل ۲۱- تخمین زاویه شیب جاده با الگوریتم‌های تخمین گر RLS و STLQF

جدول ۴- مقایسه خطای تخمین الگوریتم‌های STLQF و RLS بر مبنای معیارهای MAE، RMSE، SSE و

خطای SSE		خطای RMSE		خطای MAE		خطای تخمین گر پارامتر
RLS	STLQF	RLS	STLQF	RLS	STLQF	
$6/65 e+4$	$1/14 e+4$	۳/۰۸۳۸	۱/۲۷۹۴	۱/۷۶۷۴	۰/۸۳۵۹	β_r (deg)
۰/۰۲۸۴	$4/9 e-4$	۰/۰۰۲۰	۰/۰۰۰۳	۰/۰۰۱۰	$8/4 e-5$	$C_d A_f$ (m ²)
$4/69 e+4$	۱۴۸/۸۵	۲/۵۸۸۶	۰/۱۴۵۸	۰/۲۴۱۶	۰/۰۱۵۸	m (kg)

$$MAE = \frac{1}{N} \sum_{i=1}^N \frac{|x_i - \hat{x}_i|}{x_i} \tag{۳۱}$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \hat{x}_i)^2} \tag{۳۲}$$

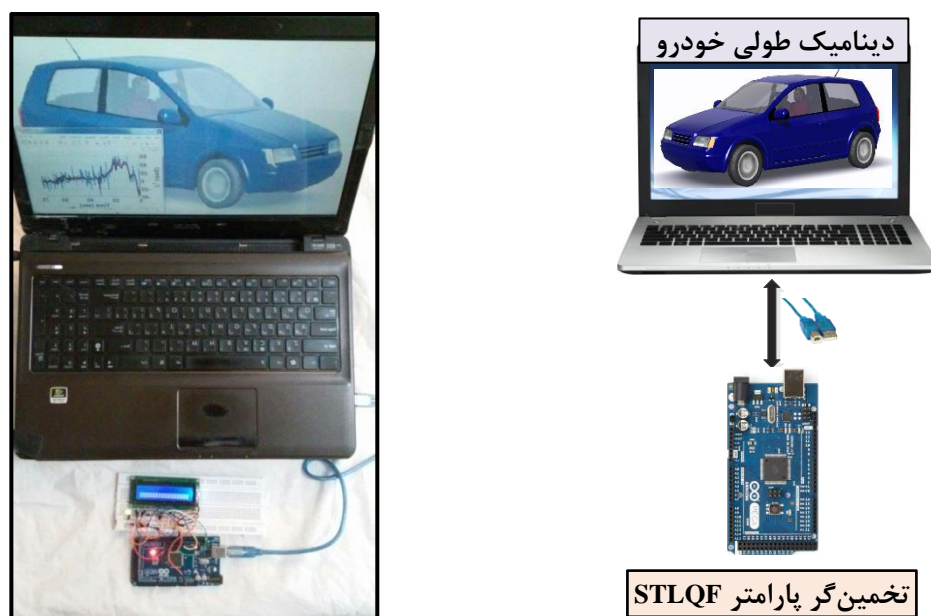
$$SSE = \sum_{i=1}^N (x_i - \hat{x}_i)^2 \tag{۳۳}$$

نتایج تخمین پارامترها دقت بالای الگوریتم STLQF را نشان می‌دهد، این در حالی است که خطای تخمین الگوریتم RLS نیز به میزان قابل ملاحظه‌ای بیشتر از تخمین گر STLQF است. قابلیت بالای الگوریتم STLQF در تخمین پارامترها به صورت بلادرنگ و با تأخیر زمانی اندک، امکان استفاده از نتایج این تخمین-گر در سیستم‌های کنترلی حلقه بسته و کاربردهای صنعتی را می‌دهد.

۵- شبیه‌سازی تخمین گر STLQF به صورت سخت‌افزار در حلقه

شبیه‌سازی سخت‌افزار در حلقه (HILS) یکی از متداول‌ترین روش‌هایی است که برای شبیه‌سازی سیستم‌های پیچیده به صورت برخط و قبل از ساخت مدل واقعی بکار می‌رود. این فرآیند به عنوان یک روش مقرون به صرفه در طراحی، توسعه، بهینه‌سازی و تست مدل‌های مختلف پیچیده در سیستم‌های تحقیقاتی و صنعتی استفاده می‌شود.

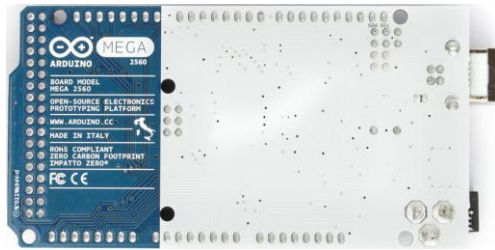
در این بخش، نحوه شبیه‌سازی الگوریتم تخمین پارامتر STLQF به صورت سخت‌افزار در حلقه با استفاده از بُرد آردوینو^۱ Mega2560 مطابق شکل (۲۲)، تشریح می‌گردد. از این طریق، تخمین‌گر STLQF می‌تواند با حداقل هزینه و در کوتاه‌ترین زمان ممکن برای استفاده در مدل یک خودروی سواری نمونه توسعه یافته و مورد ارزیابی قرار می‌گیرد. بُرد آردوینو Mega2560 که در شکل (۲۳) نشان داده شده است بر پایه میکروکنترلر ATmega2560 از خانواده AVR می‌باشد که دارای یک نوسان‌ساز کریستال ۱۶ مگاهرتزی، ۲۵۶ کیلو بایت حافظه فلش، یک درگاه USB با قابلیت اتصال به رایانه، یک دکمه ریست و نیز کلید امکانات مورد نیاز جهت بکارگیری میکروکنترلر موجود روی بُرد می‌باشد. همچنین آردوینو دارای یک محیط یکپارچه توسعه نرم‌افزار^۲ ساده است که در رایانه‌های عادی قابلیت اجرا دارد و امکان برنامه‌نویسی مستقیم را در میکروکنترلر به کمک زبان‌های C و C++ و در قالب پروتوکل STK500 به کاربر می‌دهد. ولتاژ کاری این بُرد ۵ ولت می‌باشد و تغذیه آن از طریق کابل USB، آداپتور مبدل AC به DC و باتری امکان‌پذیر است. جهت شبیه‌سازی تخمین‌گر STLQF به صورت سخت‌افزار در حلقه، این الگوریتم در میکروکنترلر برنامه‌نویسی شده و بُرد از طریق درگاه USB به رایانه متصل می‌شود. برای این منظور، الگوریتم تخمین‌گر STLQF که قبلاً در محیط MATLAB برنامه‌نویسی شده است، مطابق این روند به برنامه قابل استفاده در میکروکنترلر AVR برنامه‌نویسی می‌شود: ابتدا برنامه کدنویسی شده در محیط MATLAB به زبان C++ برنامه‌نویسی می‌شود. سپس با اعمال تغییراتی در ساختار کد C++، کد C++/Mex برنامه بدست می‌آید. تغییر عمده‌ای که در تبدیل کد از ساختار C++ به C++/Mex به وجود می‌آید، پیاده‌سازی آرایه‌های چند بعدی با استفاده از اشاره‌گر می‌باشد، برای نمونه استفاده از آدرس‌دهی دینامیکی `*u=new double [int n] به جای آدرس‌دهی استاتیکی برای یک آرایه یک بعدی: double u [cons int a]`.



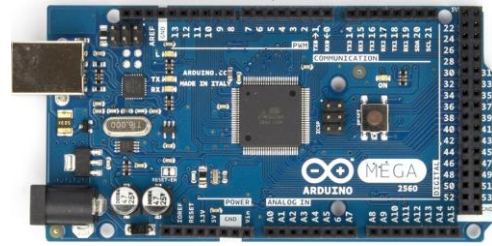
شکل ۲۲- شبیه‌سازی الگوریتم تخمین پارامتر STLQF به صورت سخت‌افزار در حلقه

¹ Arduino Board

² Integrated Development Environment (IDE)



نمای پشت بُرد آردوینو Mega2560



نمای مقابل بُرد آردوینو Mega2560

شکل ۲۳- شماتیک بُرد آردوینو Mega2560

مزیت کد C++/Mex این است که نرم‌افزار MATLAB امکان اجرای آن را همانند M-fileها دارد. در عین حال، یکی از معایب زبان MATLAB این است که سرعت اجرای به مراتب پایین‌تری نسبت به زبان C و C++ دارد؛ چرا که این زبان یک مفسر^۱ است و نه یک مترجم^۲. لذا استفاده از کد C++/Mex در محیط MATLAB بدلیل استفاده از ساختار C++ سبب می‌شود که نه تنها اجرای محاسبات بسیار سریع شود، بلکه به دلیل پیاده‌سازی کد C++/Mex در محیط MATLAB، امکان استفاده از ابزارهای این نرم‌افزار و نمایش خروجی‌ها به شکل گرافیکی فراهم می‌شود. در این مرحله الگوریتم STLQF که در ساختار برنامه C++/Mex کدنویسی شده است، در آزمون تخمین پارامتری که در بخش ۴-۲ تعریف شده بود مورد راستی-آزمایی قرار می‌گیرد. پس از اعتبارسنجی برنامه تخمین‌گر STLQF با کد C++/MEX تغییراتی جزئی در ساختار این کد اعمال می‌شود به نحوی که توسط مترجم آردوینو قابل ترجمه باشد. در ادامه، برنامه نهایی شده در میکروکنترلر AVR در بُرد آردوینو، مطابق شکل (۲۲)، بارگذاری می‌شود. سپس، مدل دینامیک طولی خودرو در محیط MATLAB در رایانه شبیه‌سازی می‌شود و خروجی‌های آن از طریق کابل USB به بُرد تخمین‌گر STLQF ارسال می‌شود. خروجی‌های این بُرد که شامل پارامترهای تخمین‌زده شده‌ی جرم خودرو، زاویه شیب جاده و ضریب درگ کلی آیرودینامیک خودرو می‌باشد محاسبه شده، به رایانه منتقل شده و در آن ذخیره می‌شوند. الگوریتم تخمین‌گر STLQF که به زبان C++ در محیط آردوینو برنامه‌نویسی شده است، در پیوست الف آورده شده است.

۵-۱- ارزیابی نتایج شبیه‌سازی سخت‌افزار در حلقه

در بخش ۴-۲، عملکرد الگوریتم تخمین‌گر STLQF مبتنی بر داده‌های شبیه‌سازی شده رایانه‌ای و تجربی ارزیابی شد. در شبیه‌سازی مذکور، مدل دینامیک طولی خودرو و تخمین‌گر STLQF در محیط MATLAB پیاده‌سازی شده بودند، که در اینجا به اختصار ساختار برنامه ۱ خوانده می‌شود. در این بخش کارایی تخمین‌گر STLQF که در بخش ۵ در ساختار C++/Mex، به اختصار ساختار برنامه ۲، و همچنین به صورت سخت‌افزار در حلقه با استفاده از میکروکنترلر AVR، به اختصار ساختار برنامه ۳، پیاده‌سازی شد ارزیابی می‌شود.

^۱ Interpreter

^۲ Compiler

به میزانی که خطا بین داده‌های تخمین پارامتر در ((ساختار برنامه‌های ۲ و ۳)) و ((ساختار برنامه ۱)) کمتر باشد، عملکرد تخمین‌گر STLQF در ساختار برنامه‌های ۲ و ۳ مطلوب ارزیابی می‌شود. جهت شبیه‌سازی تخمین‌گر STLQF به صورت سخت‌افزار در حلقه، سرعت دریافت و ارسال اطلاعات^۱ در محیط MATLAB و بُرد آردوینو Mega2560 برابر با حداکثر مقدار قابل قبول در بُرد Mega2560 که ۱۱۵۲۰۰ بیت بر ثانیه می‌باشد، تنظیم شده است.

با شبیه‌سازی آزمون تخمین پارامتر مشابه بخش ۴-۲ با هر سه ساختار برنامه ۱، ۲ و ۳، مقدار میانگین درصد خطای نسبی بین داده‌های تخمین زاویه شیب جاده در ((ساختار برنامه‌های ۲ و ۳)) و ((ساختار برنامه ۱)) که ساختار برنامه معیار می‌باشد محاسبه شده و در جدول (۵) لیست می‌شود.

علاوه بر این در جدول (۵)، سرعت برنامه به ازای اجرای یک نمونه^۲ در هر سه ساختار برنامه نشان داده می‌شود. لازم به ذکر است که سرعت اجرای برنامه کدنویسی شده در ساختار برنامه ۱، مربوط به برنامه MATLAB، و ساختار برنامه ۲، مربوط به ساختار C++/MEX، تابع واحد پردازش‌گر مرکزی^۳ رایانه است که دارای سرعت پردازش ۲/۴ گیگاهرتز و با مشخصات Intel Core i5 450M می‌باشد؛ این در حالی است که سرعت اجرای برنامه کدنویسی شده در ساختار برنامه ۳، سخت‌افزار در حلقه با استفاده از میکروکنترلر AVR، تابع پردازش‌گر میکروکنترلر AVR می‌باشد که ۱۶ مگاهرتزی است.

مطابق نتایج جدول (۵)، میانگین درصد خطای نسبی بین داده‌های تخمین زاویه شیب جاده در ((ساختار برنامه‌های ۲ و ۳)) و ((ساختار برنامه ۱)) بسیار ناچیز می‌باشد که نشان می‌دهد برنامه‌نویسی الگوریتم STLQF در ساختارهای C++/Mex و سخت‌افزار در حلقه در میکروکنترلر AVR به شکل صحیح صورت گرفته است. سرعت پردازش اطلاعات در ساختار برنامه ۲، در حدود ۲۸ برابر سرعت پردازش اطلاعات در ساختار برنامه ۱ است که حاکی از کارآمدی برنامه‌های کدنویسی شده در ساختار C++/Mex دارد.

همچنین هر چند در ساختار برنامه ۳، تخمین‌گر STLQF در میکروکنترلر AVR ارزان‌قیمت با سرعت پردازش ۱۶ مگاهرتز برنامه‌نویسی شده است، ولیکن زمان اجرای^۴ برنامه به ازای یک نمونه با دقت مناسبی نزدیک به زمان واقعی^۵ در کاربردهای عملی می‌باشد.

جدول ۵- مقایسه نتایج تخمین‌گر STLQF در سه ساختار برنامه MATLAB، C++/Mex و سخت‌افزار در حلقه

نتایج	ساختار برنامه ۱	ساختار برنامه ۲	ساختار برنامه ۳
میانگین درصد خطای نسبی بین داده‌های تخمین زاویه شیب جاده در ((ساختار برنامه‌های ۲ و ۳)) و ((ساختار برنامه ۱))	-	$7/3 \times 10^{-8}$	$4/6 \times 10^{-7}$
سرعت برنامه به ازای اجرای یک نمونه	۲/۳۶۱ میلی ثانیه	۰/۰۸۵ میلی ثانیه	۴۹۸/۵۲۱ میلی ثانیه

¹ Baudrate

² Sample

³ Central Processor Unit (CPU)

⁴ Run-Time

⁵ Real-time

۶- جمع بندی

در این مقاله، الگوریتم تخمین پارامتر با نام فرم درجه دوم خطی به صورت زمان کوتاه STLQF جهت تخمین همزمان پارامترهای مؤثر بر دینامیک طولی خودرو شامل جرم خودرو، ضریب درگ کلی آیرودینامیک خودرو و زاویه شیب جاده طراحی شد. برای این منظور، ابتدا معادلات حاکم بر مدل دینامیک طولی برای یک خودروی سواری پنج سرعته با موتور بنزینی و سیستم جعبه دنده خودکار استخراج شد. مدل بدست آمده، قابلیت شبیه‌سازی سیستم انتقال قدرت و اجزای اصلی آن شامل موتور، مبدل گشتاور، جعبه دنده خودکار، محورها و چرخ‌ها را داشت. برای اعتبارسنجی مدل دینامیک طولی، پاسخ‌های مدل فعلی در یک مانور با پاسخ‌های مدلی که در نمونه برنامه‌ای با عنوان sldemo_autotrans در نرم‌افزار Simulink شبیه‌سازی شده است مقایسه شدند. شبیه‌سازی هم‌زمان این دو مدل، از انطباق کامل نتایج مدل فعلی با مدل sldemo_autotrans حکایت دارد. سپس تخمین گر STLQF بر روی مدل دینامیک طولی خودرو که به صورت رگرسیون خطی بازنویسی شده بود پیاده‌سازی شد و کارکرد مؤثر آن در مقایسه با الگوریتم تخمین پارامتر RLS نمایش داده شد. در پایان، الگوریتم تخمین گر STLQF به صورت سخت‌افزار در حلقه شبیه‌سازی شد تا از این طریق، با حداقل هزینه و در کوتاه‌ترین زمان ممکن برای استفاده در یک خودروی سواری نمونه توسعه یافته و مورد ارزیابی قرار گیرد. نتایج شبیه‌سازی الگوریتم تخمین پارامتر STLQF به صورت سخت‌افزار در حلقه، عملکرد صحیح آن را برای کاربردهای زمان واقعی نشان می‌دهند. عمده مزیت‌ها و نوآوری‌های پژوهشی صورت گرفته در این مقاله را می‌توان در موارد زیر خلاصه کرد:

- جمع‌آوری داده‌های واقعی زاویه شیب جاده از یکی از خیابان‌های مرتفع شهر تهران
 - توسعه الگوریتم تخمین گر STLQF برای تخمین هم‌زمان و با دقت مناسب سه پارامتر جرم خودرو، زاویه متغیر شیب جاده و ضریب درگ کلی آیرودینامیک خودرو بدون تعریف فرضیات ساده‌کننده بر روی پارامترها
 - شبیه‌سازی تخمین گر STLQF به صورت سخت‌افزار در حلقه و امکان‌سازی توسعه این تخمین گر جهت استفاده در الگوریتم‌های کنترل دینامیک طولی خودرو
- برای ادامه این پژوهش، تخمین سایر عوامل مؤثر بر دینامیک طولی خودرو شامل ضریب مقاومت غلتشی جاده، سرعت باد و اصطکاک بین تایر خودرو و جاده پیشنهاد می‌گردد. همچنین استفاده از الگوریتم تخمین-گر STLQF در یک حلقه بسته کنترل تطبیقی برای سامانه‌های کنترل خودرویی که نسبت به وجود عدم قطعیت‌های پارامتری در دینامیک خودرو و جاده حساس هستند، پیشنهاد می‌شود.

مراجع

- [1] McIntyre, M. L., Ghotikar, T. J., Vahidi, A., Song, X., and Dawson, D. M., "A Two-stage Lyapunov-based Estimator for Estimation of Vehicle Mass and Road Grade", *IEEE Trans. Veh. Technol.*, Vol. 58, No. 7, pp. 3177–3185, (2009).
- [2] Rozyn, M., and Zhang, N., "A Method for Estimation of Vehicle Inertial Parameters", *Veh. Syst. Dyn.*, Vol. 48, No. 5, pp. 547–565, (2010).
- [3] Druzhinina, M., Stefanopoulou, A., and Moklegaard, L., "Adaptive Continuously Variable Compression Braking Control for Heavy-duty Vehicles", *J. Dyn. Syst. Meas. Control*, Vol. 124, pp. 406–414, (2002).
- [4] Hsu, L. Y., and Chen, T. L., "Vehicle Dynamic Prediction Systems with Online Identification of Vehicle Parameters and Road Conditions", *J. Sensors*, Vol. 12, No. 11, pp. 15778–15800, (2012).
- [5] Polanco, A. P., and Munoz, L., "Influence of Inertial Properties on the Comfort of Road Vehicles", *SAE Tech. Pap.* 2011-01-0436, (2011).
- [6] Tan, Y., Robotis, A., and Kanellakopoulos, I., "Speed Control Experiments with an Automated Heavy Vehicle", *Proc. 1999 IEEE Int. Conf. Control Appl.*, Vol. 2, pp. 1353–1358, Kohala Coast, USA, (1999).
- [7] Chen, Y., and Wang, J., "Adaptive Vehicle Speed Control with Input Injections for Longitudinal Motion Independent Road Frictional Condition Estimation", *IEEE Trans. Veh. Technol.*, Vol. 60, No. 3, pp. 839–848, (2011).
- [8] Huang, X., and Wang, J., "Adaptive Vehicle Planar Motion Control with Fast Parameter Estimation", *Proc. IEEE Conf. Decis. Control*, pp. 5034–5039, Maui, USA, (2012).
- [9] Ohnishi, H., Ishii, J., Kayano, M., and Katayama, H., "Study on Road Slope Estimation for Automatic Transmission Control", *JSAE Rev.*, Vol. 21, pp. 235–240, (2000).
- [10] Bae, H. S., Ryu, J., and Gerdes, J. C., "Road Grade and Vehicle Parameter Estimation for Logitudinal Control using GPS", *2001 IEEE Intell. Transp. Syst. Proc.*, pp. 166–171, (2001).
- [11] Zhang, K., and Frey, H. C., "Road Grade Estimation for On-road Vehicle Emissions Modeling using Light Detection and Ranging Data", *J. Air Waste Manage. Assoc.*, Vol. 56, No. 6, pp. 777–788, (2006).
- [12] Ren, Z., and Wang, H., "Road Slope Estimation Algorithm and Safety Analysis of Mountainous Road", *J. Inf. Comput. Sci.*, Vol. 11, No. 7, pp. 2253–2265, (2014).
- [13] Fathy, H. K., Kang, D. K. D., and Stein, J. L., "Online Vehicle Mass Estimation using Recursive Least Squares and Supervisory Data Extraction", *2008 Am. Control Conf.*, pp. 1842–1848, Seattle, USA, (2008).

- [14] Vahidi, A., Stefanopoulou, A., and Peng, H., "Recursive Least Squares with Forgetting for Online Estimation of Vehicle Mass and Road Grade: Theory and Experiments", *Veh. Syst. Dyn.*, Vol. 43, No. 1, pp. 31–55, (2005).
- [15] Vahidi, A., Druzhinina, M., Stefanopoulou, A., and Peng, H. P. H., "Simultaneous Mass and Time-varying Grade Estimation for Heavy-duty Vehicles", *Proc. 2003 Am. Control Conf.*, Vol. 6, pp. 4951–4956, Denver, USA, (2003).
- [16] Ramprabu, J., and Sindhuja, G., "Vehicle Stability Maintenance through Online Estimation of Parameter Uncertainties", *Int. J. Emerg. Trends Technol. Comput. Sci.*, Vol. 3, No. 5, pp. 132–136, (2014).
- [17] Kedar-Dongarkar, G., and Das, M., "Vehicle Parameter Estimation using Nested RLS Algorithm", *Midwest Symp. Circuits Syst.*, pp. 404–407, Columbus, USA, (2013).
- [18] Zarringhalam, R., Rezaeian, A., Melek, W., Khajepour, A., and Chen, S., "A Comparative Study on Identification of Vehicle Inertial Parameters", *Proc. 2012 Am. Control Conf.*, pp. 3599–3604, Montreal, Canada, (2012).
- [19] Hong, S., Lee, C., Borrelli, F., and Hedrick, J. K., "A Novel Approach for Vehicle Inertial Parameter Identification using a Dual Kalman Filter", *IEEE Trans. Intell. Transp. Syst.*, Vol. 16, No. 1, pp. 151–161, (2014).
- [20] Nada, E., Ahmed, A., and Abd-alla, M., "Modified Dual Unscented Kalman Filter Approach for Measuring Vehicle States and Vehicle Parameters", *Int. J. Eng. Res. Technol.*, Vol. 3, No. 8, pp. 1423–1430, (2014).
- [21] Winstead, V., and Kolmanovsky, I. V., "Estimation of Road Grade and Vehicle Mass via Model Predictive Control", *Proc. 2005 IEEE Conf. Control Appl.*, pp. 1588–1593, (2005).
- [22] Homaeinezhad, M. R., Tahbaz-zadeh Moghaddam, I., Khakpour, Z., and Naseri, H., "Short-time Linear Quadratic Form Technique for Estimating Fast-varying Parameters in Feedback Loops", *Asian J. Control*, Vol. 18, No. 4, pp. 1–14, (2015).
- [23] Marzbanrad, J., and Tahbaz-zadeh Moghaddam, I., "Self-tuning Control Algorithm Design for Vehicle Adaptive Cruise Control System through Real-time Estimation of Vehicle Parameters and Road Grade", *Veh. Syst. Dyn.*, Vol. 54, No. 9, pp. 1291–1316, (2016).
- [24] Liang, H., Chong, K. T., No, T. S., and Yi, S. Y., "Vehicle Longitudinal Brake Control using Variable Parameter Sliding Control", *Control Eng. Pract.*, Vol. 11, pp. 403–411, (2003).
- [25] Ljung, L., "*System Identification: Theory for the User*", Prentice Hall, (1999).
- [26] Marzbanrad, J., and Tahbaz-zadeh Moghaddam, I., "Prediction of Driver's Accelerating Behavior in the Stop and Go Maneuvers using Genetic Algorithm-Artificial Neural Network Hybrid Intelligence", *Int. J. Automot. Eng.*, Vol. 5, No. 2, pp. 986–998, (2015).

فهرست نمادهای انگلیسی

- m : جرم خودرو
- r_w : شعاع مؤثر چرخ
- R_g : ضریب دنده
- R_{fd} : شعاع مؤثر چرخ
- K_b : ثابت تبدیل فشار-گشتاور
- I_{ei} : مجموع لختی اینرسی موتور و پمپ
- $C_d A_f$: ضریب درگ کلی آیرودینامیک
- f : ضریب مقاومت غلتشی
- v : سرعت خودرو
- z : خروجی قابل اندازه‌گیری
- A : ماتریس متقارن، حقیقی و معین مثبت
- e : خطای مدل‌سازی
- w : خطای تعمیم یافته
- p : بردار ضرایب پارامترهای تخمین
- S_N : نویز سفید گوسی
- WL : نصف طول پنجره محتوی رگرسورهای سیستم
- T : گام زمانی
- σ_N : انحراف از معیار اغتشاش
- Θ : بردار پارامترهای سیستم دینامیکی
- Ψ : ماتریس رگرسورهای اصلی سیستم
- Φ : ماتریس رگرسورهای ثانویه سیستم

پیوست الف

کد شبیه‌سازی الگوریتم تخمین گر STLQF که به زبان C++ در محیط آردوینو برنامه‌نویسی شده است.

```
#include <math.h>
#define SWAP(a,b) { double temp=(a);(a)=(b);(b)=temp;}
double dq[11], ddq[11], Input[11], T; int WL; // Input Data
double p_Est[3]; // Output Data
double InData[35];
unsigned long Start, Elapse, End;
void setup()
{
  // for Serial Monitoring using Arduino
  Serial.begin(115200);
}
void loop()
{
  Start = millis();
```

```

// waits until the number of collected data be more than 34
if (Serial.available() > 34)
{
// Storing Input Data
for (int i = 0; i < 35; i++)
{
InData[i] = Serial.parseFloat(); //converts the string received into a double
// delay(1); // milisecond
}
// delay(1); // milisecond
for (int i = 0; i < 11; i++)
{
dq[i] = InData[i];
ddq[i] = InData[i + 11];
Input[i] = InData[i + 22];
}
WL = InData[33];
T = InData[34];
//=====
double beta_f = 0.015, g = 9.81, rho = 1.206;
//=====
//===== m,beta,Cd*Af regMat=====
const int p_num1 = 3;
const int WL1 = WL;
double regMat[WL1][p_num1 + 1];
for (int j = 0; j < WL1; j++)
{
regMat[j][0] = ddq[j];
regMat[j][1] = g / cos(beta_f);
regMat[j][2] = 0.5 * rho * pow(dq[j], 2);
regMat[j][3] = Input[j];
}
//=====
// Linear Quadratic Form (LQF) Parameters
double q11 = 1;
double q22 = 2;
double q33 = 3;
double q12 = -1;
double q13 = -1;
double q23 = 1;
// s[n] Parameters
// s[n]=a0*e[n]+a1*e[n-1]+a2*e[n-2]
double a0 = 1 + 1 / T + 1 / pow(T, 2);
double a1 = -(1 / T + 2 / pow(T, 2));
double a2 = 1 / pow(T, 2);
double rMat1[WL1][p_num1 * 2 + 1];
double tV[2];
double r0[p_num1 * 2];
double r1[p_num1 * 2];
double r2[p_num1 * 2];
double r3[p_num1 * 2];
double r4[p_num1 * 2];
double y0, y1, y2, y3, y4;
double G[p_num1 * 2][p_num1 * 2];
double U[p_num1 * 2];
//G = 0; U = 0;
for (int counter = 0; counter < p_num1 * 2; counter++)
{
U[counter] = 0;
for (int counter2 = 0; counter2 < p_num1 * 2; counter2++)

```

```

    {
        G[counter][counter2] = 0;
    }
}
// Estimation Loop
//=====
// Converting Original Regressors to Augmented Regressors\
// 1rd-Order Sinusoid Structure for Time-Varying Parameters
for (int k = 0; k < WL1; k++)
{
    tV[0] = (k - 1) * sin(T);
    tV[1] = 1.0;
    int counter = 0;
    for (int count = 0; count < p_num1; count++)
        for (int count2 = 0; count2 < 2; count2++)
        {
            rMat1[k][counter] = regMat[k][count] * tV[count2];
            counter++;
        }
    rMat1[k][p_num1 * 2] = regMat[k][p_num1];
}
//=====
for(int j = 4; j<WL1; j++)
{
    for (int count=0; count<p_num1*2; count++)
    {
        r0[count] = rMat1[j - 0][count];
        r1[count] = rMat1[j - 1][count];
        r2[count] = rMat1[j - 2][count];
        r3[count] = rMat1[j - 3][count];
        r4[count] = rMat1[j - 4][count];
    }
    y0 = rMat1[j - 0][p_num1 * 2];
    y1 = rMat1[j - 1][p_num1 * 2];
    y2 = rMat1[j - 2][p_num1 * 2];
    y3 = rMat1[j - 3][p_num1 * 2];
    y4 = rMat1[j - 4][p_num1 * 2];
    for(int counter1=0; counter1<p_num1*2; counter1++)
        for(int counter2=0; counter2<p_num1*2; counter2++)
        {
            G[counter1][counter2] += (q11*(a0*r0[counter1]+a1*r1[counter1]+a2*r2[counter1])) *
            (a0*r0[counter2]+a1*r1[counter2]+a2*r2[counter2] +
            (q22*(a0*r1[counter1]+a1*r2[counter1]+a2*r3[counter1])) *
            (a0*r1[counter2]+a1*r2[counter2]+a2*r3[counter2]) +
            (q33*(a0*r2[counter1]+a1*r3[counter1]+a2*r4[counter1])) *
            (a0*r2[counter2]+a1*r3[counter2]+a2*r4[counter2]) +
            (q12*(a0*r1[counter1]+a1*r2[counter1]+a2*r3[counter1])) *
            (a0*r0[counter2]+a1*r1[counter2]+a2*r2[counter2]) +
            (q12*(a0*r0[counter1]+a1*r1[counter1]+a2*r2[counter1])) *
            (a0*r1[counter2]+a1*r2[counter2]+a2*r3[counter2]) +
            (q13*(a0*r2[counter1]+a1*r3[counter1]+a2*r4[counter1])) *
            (a0*r0[counter2]+a1*r1[counter2]+a2*r2[counter2]) +
            (q13*(a0*r0[counter1]+a1*r1[counter1]+a2*r2[counter1])) *
            (a0*r2[counter2]+a1*r3[counter2]+a2*r4[counter2]) +
            (q23*(a0*r2[counter1]+a1*r3[counter1]+a2*r4[counter1])) *
            (a0*r1[counter2]+a1*r2[counter2]+a2*r3[counter2]) +
            (q23*(a0*r1[counter1]+a1*r2[counter1]+a2*r3[counter1])) *
            (a0*r2[counter2]+a1*r3[counter2]+a2*r4[counter2]));
        }
    for( int counter=0; counter<p_num1 *2; counter++)

```

```

    {
        U[counter] += q11*(a0*r0[counter]+a1*r1[counter]+a2*r2[counter])*(a0*y0+a1*y1+a2*y2) +
        q22*(a0*r1[counter]+a1*r2[counter]+a2*r3[counter])*(a0*y1+a1*y2+a2*y3) +
        q33*(a0*r2[counter]+a1*r3[counter]+a2*r4[counter])*(a0*y2+a1*y3+a2*y4) +
        q12*(a0*r1[counter]+a1*r2[counter]+a2*r3[counter])*(a0*y0+a1*y1+a2*y2) +
        q12*(a0*r0[counter]+a1*r1[counter]+a2*r2[counter])*(a0*y1+a1*y2+a2*y3) +
        q13*(a0*r2[counter]+a1*r3[counter]+a2*r4[counter])*(a0*y0+a1*y1+a2*y2) +
        q13*(a0*r0[counter]+a1*r1[counter]+a2*r2[counter])*(a0*y2+a1*y3+a2*y4) +
        q23*(a0*r2[counter]+a1*r3[counter]+a2*r4[counter])*(a0*y1+a1*y2+a2*y3) +
        q23*(a0*r1[counter]+a1*r2[counter]+a2*r3[counter])*(a0*y2+a1*y3+a2*y4);
    }
}
//=====
// Matrix Inversion Method
int i, j, k;
const int n2 = p_num1 * 2;
double polyC[n2];
double AB[n2][n2 + 1];
// Inversing Lemma
//=====
for (i = 0; i < n2; i++)
{
    for (j = 0; j < n2; j++)
    {
        AB[i][j] = G[i][j];
    }
    AB[i][n2] = U[i];
}
for (i = 0; i < n2 - 1; i++)
{
    int on = 0;
    if (AB[i][i] == 0)
    {
        for (j = i + 1; j < n2; j++)
        {
            if (on == 0 && AB[j][i] != 0)
            {
                for (k = 0; k < n2 + 1; k++)
                {
                    SWAP(AB[i][k], AB[j][k]);
                }
                on = 1;
            }
        }
    }
}
double B[n2];
for (i = 0; i < n2; i++)
{
    B[i] = AB[i][n2];
}
double A[n2][n2];
for (i = 0; i < n2; i++)
{
    for (j = 0; j < n2; j++)
    {
        A[i][j] = AB[i][j];
    }
}
for (k = 0; k < n2 - 1; k++)

```

```

{
  for (i = k + 1; i < n2; i++)
  {
    for (j = k + 1; j < n2; j++)
    {
      A[i][j] = A[i][j] - (A[i][k] / A[k][k]) * A[k][j];
    }
    B[i] = B[i] - (A[i][k] / A[k][k]) * B[k];
  }
}
polyC[n2 - 1] = B[n2 - 1] / A[n2 - 1][n2 - 1];
for (k = 0; k < n2 - 1; k++)
{
  i = n2 - k - 2;
  for (j = i + 1; j < n2; j++)
  {
    B[i] = B[i] - (A[i][j] * polyC[j]);
  }
  polyC[i] = B[i] / A[i][i];
}
double ti[WL1];
for (int counter = 0; counter < WL1; counter++)
  ti[counter] = counter * T;
double paramEsti[p_num1];
for (int i = 0; i < p_num1; i++)
{
  int pSize = 2;
  int pTo = (i + 1) * 2 - 1;
  double ans[WL1];
  for (int j = 0; j < WL1; j++)
  {
    ans[j] = 0;
    for (int i = 0; i < pSize; i++)
    {
      ans[j] += polyC[pTo - i] * pow(ti[j], i);
    }
  }
  double sum = 0;
  for (int i = 0; i < WL1; i++)
  {
    sum += ans[i];
  }
  paramEsti[i] = sum / WL1;
}
p_Est[0] = paramEsti[0];
p_Est[1] = asin(paramEsti[1]/paramEsti[0])-beta_f;
p_Est[2] = paramEsti[2];
// Send Output Data to Matlab and goes to next line
Serial.println(p_Est[0],16);
// delay(1); // milisecond
Serial.println(p_Est[1],16);
// delay(1); // milisecond
Serial.println(p_Est[2],16);
// Computing elapse time
End = millis();
Elapse = End - Start;
Serial.println(Elapse);
}
}

```

Abstract

Appropriate control of the vehicle longitudinal dynamic requires accurate and real-time knowledge of the parameters affecting the dynamics of the vehicle. The main contribution of this paper is in developing an estimator called short-time linear quadratic form (STLQF) which is a new parameter estimation technique to simultaneously estimate the time-varying parameters that affect on vehicle longitudinal dynamics with a low latency. These parameters are the vehicle mass, time varying road slope angle and overall aerodynamic drag coefficient of the vehicle in real time. Next, the efficiency of STLQF algorithm is shown by comparing the results of STLQF technique in an experimental test against the recursive least squares (RLS) parameter estimation method. The results of implementing STLQF parameter estimation algorithm, shows the better performance of STLQF estimator compared with RLS algorithm. In the end, the STLQF estimator is simulated in hardware in the loop configuration using an AVR microcontroller in order to rapid prototype of the STLQF method by minimum cost and high accuracy in the least possible time.