



**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA**

Olmer García Bedoya

**Análise de risco para
a cooperação entre o condutor e sistema de
controle de veículos autônomos**

CAMPINAS
2016

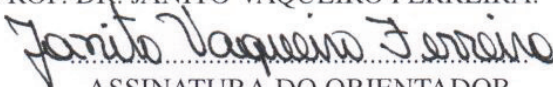
Olmer García Bedoya

**Análise de risco para
a cooperação entre o condutor e sistema de
controle de veículos autônomos**

Tese de Doutorado apresentada à Faculdade de Engenharia Mecânica da Universidade Estadual de Campinas como parte dos requisitos exigidos para obtenção do título de Doutor em Engenharia Mecânica, na Área da Mecânica dos Sólidos e Projeto Mecânico

Orientador: Prof. Dr. Janito Vaqueiro Ferreira
Coorientador:

ESTE EXEMPLAR CORRESPONDE À VERSÃO
FINAL DA TESE DEFENDIDA PELO ALUNO
OLMER GARCIA BEDOYA ORIENTADO PELO
PROF. DR. JANITO VAQUEIRO FERREIRA.



ASSINATURA DO ORIENTADOR

CAMPINAS
2016

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Luciana Pietrosanto Milla - CRB 8/8129

G165a Garcia Bedoya, Olmer, 1983-
Análise de risco para a cooperação entre o condutor e sistema de controle de veículos autônomos / Olmer Garcia Bedoya. – Campinas, SP : [s.n.], 2016.

Orientador: Janito Vaqueiro Ferreira.
Tese (doutorado) – Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Veículos autônomos. 2. Análise de trajetória. 3. Controle preditivo. 4. Engenharia automotiva. I. Ferreira, Janito Vaqueiro, 1961-. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Risk analysis for cooperation between the driver and the control system of an autonomous vehicle

Palavras-chave em inglês:

Autonomous robots

Analysis of trajectory

Predictive control

Automotive engineering

Área de concentração: Mecânica dos Sólidos e Projeto Mecânico

Titulação: Doutor em Engenharia Mecânica

Banca examinadora:

Janito Vaqueiro Ferreira [Orientador]

Pablo Siqueira Meirelles

Eurípedes Guilherme de Oliveira Nóbrega

Denis Fernando Wolf

Cairo Lúcio Nascimento Júnior

Data de defesa: 26-02-2016

Programa de Pós-Graduação: Engenharia Mecânica

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE MECÂNICA COMPUTACIONAL

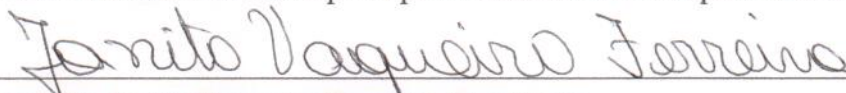
TESE DE DOUTORADO

**Análise de risco para
a cooperação entre o condutor e sistema de
controle
de veículos autônomos**

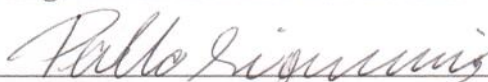
Autor: Olmer García Bedoya

Orientador: Prof. Dr. Janito Vaqueiro Ferreira

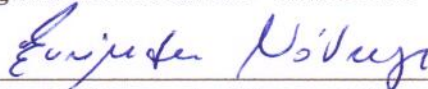
A Banca Examinadora composta pelos membros abaixo aprovou esta Tese:



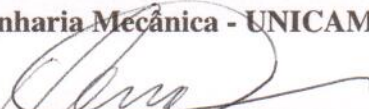
Prof. Dr. Janito Vaqueiro Ferreira, Presidente
Faculdade de Engenharia Mecânica - UNICAMP



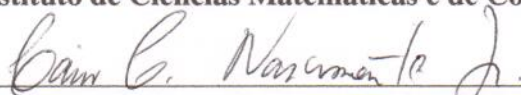
Prof. Dr. Pablo Siqueira Meirelles
Faculdade de Engenharia Mecânica - UNICAMP



Prof. Dr. Eurípedes Guilherme de Oliveira Nóbrega
Faculdade de Engenharia Mecânica - UNICAMP



Prof. Dr. Denis Fernando Wolf
Instituto de Ciências Matemáticas e de Computação (ICMC). USP de São Carlos



Prof. Dr. Cairo Lúcio Nascimento Júnior
Divisão de Engenharia Eletrônica -ITA

Dedicatória

A meu querido filho Sergio, minha esposa Marcela e minha grande família começando por meus Pais, Olmer e Olga, minha irmã e meus sobrinhos. Agradeço de todo coração e dou graças a Deus por fazer parte de minha vida, desejo estar sempre junto no caminho da vida.

Agradecimentos

À Deus, minha fonte de força, fé e determinação. E como em todos os dias, agradeço pelo dom da vida.

A toda minha família por toda a paciência e acreditar em mim, com seu apoio emocional e espiritual.

Ao meu orientador, Prof. Dr. Janito Vaqueiro Ferreira pelo tempo, dedicação e paciência durante todo o processo.

Aos membros das bancas de qualificação e defesa por seus valiosos aportes.

Aos professores Doutores Arthur de Miranda Neto e Pablo Siqueira Meirelles, por todas as contribuições a mim e ao projeto.

Aos amigos e companheiros de Laboratório de Mobilidade Autônoma (LMA) Flávio C. Bannwart, Tiberio Ferreira, Sergio Hirota, Matheus de Lima, Vinicius Falqueto, Larissa Santeno Helder Daiha, Carlos Daniel Reyes Bautista, Randerson Lemos, Ruben Dario Hernandez, Maria Fernanda Rodriguez e todos os que tem participado no desenvolvimento deste projeto.

A todo o pessoal do Laboratório De Estudos Dinâmicos (LABEDIN) e do Laboratorio de Ensaio de vibrações pela sua colaboração na automação do veículo.

A meu amigo, o Doutor Giovani Bernardes pela sua colaboração no tema de percepção.

A CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior e CNPQ - Conselho Nacional de Desenvolvimento Científico e Tecnológico, pelo indispensável apoio financeiro.

O those who do not know mathematics it is difficult to get across a real feeling as to the beauty, the deepest beauty, of nature. If you want to learn about nature, to appreciate nature, it is necessary to understand the language that she speaks in.

Richard P. Feynman

Resumo

GARCIA BEDOYA, Olmer. Análise de risco para a cooperação entre o condutor e sistema de controle de veículos autônomos . 2016. 239p. Tese (Doutorado). Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas.

Este trabalho tem como objetivo principal estudar estratégias para a cooperação entre o condutor e o sistema de controle de trajetórias de veículos autônomos por uma análise de risco. Inicialmente apresenta-se um estudo das arquiteturas dos veículos autônomos baseadas nas camadas de percepção, planejamento e controle. Baseado neste estudo definiu-se uma arquitetura de software e de hardware para a plataforma de testes VILMA01 (Veículo Inteligente do Laboratório de Mobilidade Autônoma) que possibilita a interação com o condutor. Depois é apresentada a camada do controle de trajetórias que consiste em manipular os graus de liberdade do veículo (direção, freio e acelerador) para leva-lo a uma posição desejada para cada instante de tempo. Para isso, este trabalho utiliza uma técnica de controle preditivo baseado nos modelos dinâmicos do veículo e da direção. Na sequência é apresentado o planejamento de trajetórias que consiste em saber para onde deve ir o veículo de acordo com a percepção e a missão. Nesta camada é apresentada a parte reativa também conhecida como planejamento local de caminho, onde a rota desejada representada em um espaço curvilíneo é selecionada a partir de indicadores de risco intrínseco e extrínseco de cada caminho. Com as camadas de planejamento e controle definidas é proposto um método pelo qual pode-se estimar durante o controle cooperativo a trajetória desejada do condutor, possibilitando uma decisão a ser tomada com base numa análise de risco para condicionamento do controle cooperativo ou do planejamento cooperativo. Finalmente, diferentes testes experimentais foram realizados os quais permitiram validar a automação, a instrumentação da plataforma VILMA01, e os conceitos de controle e planejamento cooperativos.

Palavras-chave: veículos autônomos, planejamento de trajetórias, controle preditivo, engenharia automotiva, controle cooperativo

Abstract

GARCIA BEDOYA, Olmer. Risk analysis for cooperation between the driver and the control system of an autonomous vehicle. 2016. 239p. Doutorado (Tese). Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas.

This work aims to study the conditioning of the cooperative path control through a risk analysis. First it is presented the study of the architectures of autonomous vehicles based on the layers of perception, planning and control. An architecture that includes interaction with the driver is proposed to VILMA01 (First Intelligent Vehicle of the Autonomous Mobility Laboratory), as well as their hardware and software architectures. Second, It is presented the path control layer consisting of manipulating the degrees of freedom of the car (steering, braking and acceleration) for bringing it to a desired position for each instant of time. In order to achieve that, the dynamic models of the vehicle and the steering system are used to apply the predictive control technique. Then, it is presented the path planning layer which consist of determining where the vehicle should go according to the perception and the mission. This layer shows the reactive part also known as local path planning, where the desired path represented in a curvilinear space is selected based on intrinsic and extrinsic risk indicators of each path. With the layers of planning and control already set, a method is proposed to estimate the trajectory desired by the driver during the cooperative control, allowing a decision to be made based on a risk analysis for conditioning of the planning or the control. Finally, different tests on VILMA01 are performed to validate the proposed work.

Keywords: Autonomous robots, autonomous navigation, model predictive control, automotive engineering, cooperative control.

Lista de Ilustrações

| | | |
|-----|--|----|
| 1.1 | Sistema <i>Intelligent Drive</i> de Mercedes-Benz. fonte web do mercedez benz . . . | 28 |
| 1.2 | Mortes por acidentes de trafego por 100000. fontes: Brasil(DATASUS), Estados Unidos(NHTSA), Iran (BAHADORIMONFARED <i>et al.</i> , 2013), Alemanha(destatis.de) e França(www.securite-routiere.gov.fr) | 29 |
| 1.3 | Mortes por acidentes de trafego em Brasil. fontes: ministério de saudê (DATASUS), a entidade gestora do seguro obrigatório (DPVAT), rodovias federais(DENATRAN) | 30 |
| 1.4 | a) Número de acidentes de tráfego pelo tipo de acidentes nas rodovias federais. fonte: rodovias federais(DENATRAN) b) Número de acidentes e mortes por tipo de acidente. | 30 |
| 1.5 | Causas de acidentes de tráfego. fonte (CET, 2012) | 31 |
| 1.6 | Estratégia de transferência de controle com indicadores de risco de trajetória | 32 |
| 1.7 | Exemplo análises de risco no condicionamento de controle cooperativo | 33 |
| 2.1 | Modelo de camadas de robótica móvel. Adaptado de (SIEGWART <i>et al.</i> , 2011a) | 37 |
| 2.2 | Arquitetura do VILMA autônomo no contexto da interação de áreas de conhecimento. | 37 |
| 2.3 | Arquitetura do sistema de automação. Comunicação: Ethernet (azul), RS232 (verde), RF (laranja), eletricas (preto) e CAN (Roxa) | 39 |
| 2.4 | Arquitetura do software para o veículo autônomo. Processamento em Linux (Preto), interação com o usuário (Cinza), firmware (Verde), embarcado (Azul). | 40 |
| 2.5 | Hardware conetado a VILMA01 através de software no ROS. | 43 |
| 3.1 | Controle de trajetória em duas camadas | 45 |
| 3.2 | Curvas de torque de assistência ao condutor do sistemas EPAS dependentes da velocidade . fonte (JONES E WILLIAMS, 2001) | 46 |
| 3.3 | Diagrama de blocos do controle preditivo generalizado (<i>MPC</i>) no caso sem restrições e com observador de estados | 55 |
| 3.4 | VDA test path. as linhas vermelhas são os limites de <i>VDA test</i> representam os retângulos azuis é o veículo durante as simulações | 60 |
| 3.5 | Exemplo correlação cruzada entre θ_p e θ | 61 |
| 3.6 | Diagrama de blocos da simulação para o controlador de trajetórias. | 62 |
| 3.7 | Erros na trajetória recorrida pelo veículo | 65 |
| 3.8 | Variáveis de estado do sistema de direção do veículo | 66 |
| 3.9 | Variáveis de estado do veículo | 67 |

| | | |
|------|---|-----|
| 3.10 | Sinais de controle do controle de trajetórias. | 68 |
| 3.11 | Ângulo de estercamento das rodas(δ_f) com $v_x = 20m/s$ e diferentes valores de ponderação das variáveis de controle $Q_{mpc}^{1,1}$ | 69 |
| 3.12 | Angulo de guinada do veículo (θ) com $v_x = 5m/s$ e diferentes valores de ponderação das variáveis de controle $Q_{mpc}^{4,4}$ | 70 |
| 3.13 | Variáveis de estado do veículo quando o estimador de estado tem giroscópio | 72 |
| 3.14 | Variáveis de estado de erro de trajetória quando o estimador de estado tem giroscópio mais um GPS | 73 |
| 3.15 | Sinal de controle do controlador do condutor e do controlador automático. | 74 |
| 3.16 | Sinal de controle do controlador do condutor e do controlador automático. | 75 |
| 4.1 | Reparametrização no arco da distancia das <i>splines</i> do caminho global | 77 |
| 4.2 | Localização de p_{carro} no espaço curvilíneo global em um segmento da estrada. Adaptada de (CHU <i>et al.</i> , 2012) | 79 |
| 4.3 | Função do caminho no espaço curvilíneo global. Adaptada de (CHU <i>et al.</i> , 2012) | 82 |
| 4.4 | Representação da função de custo pela consistência do caminho . Adaptada de (CHU <i>et al.</i> , 2012) | 85 |
| 4.5 | Risco de colisão dos caminhos candidatos. Adaptada de (CHU <i>et al.</i> , 2012) | 86 |
| 4.6 | Estrada de referencia para a simulação | 91 |
| 4.7 | Estrada de referência para o testes | 92 |
| 4.8 | Simulação do sensor LIDAR através de um mapa de linhas | 93 |
| 4.9 | Resultado de uma amostra do LIDAR no ambiente de simulação | 93 |
| 4.10 | Resultado de uma amostra do LIDAR com respeito ao ângulo de saída ϕ_i | 94 |
| 4.11 | Resultado da detecção de linhas do LIDAR no ambiente de simulação | 94 |
| 4.12 | Resultado da detecção das variáveis de estado do objeto em movimento | 96 |
| 4.13 | Caminhos candidatos para diferentes condições iniciais | 97 |
| 4.14 | Seleção de trajetória para diferentes funções de custo | 98 |
| 4.15 | Seleção de trajetória ante diferentes funções de custo na parte inicial da simulação | 98 |
| 4.16 | Função de custo para diferentes critérios | 99 |
| 4.17 | Trajectoria ante um objeto em movimento na mesma velocidade | 101 |
| 4.18 | Rotas possíveis para seguir o objeto em movimento | 101 |
| 4.19 | Valores do custo de cada um dos caminhos com um objeto em movimento na frente | 102 |
| 4.20 | Soma ponderada do custo dos caminhos possíveis para seguir um objeto em movimento | 102 |
| 4.21 | Soma ponderada do custo do caminhos possíveis para seguir um objeto em movimento | 103 |
| 4.22 | Soma ponderada do custo seguindo um objeto em movimento | 104 |

| | | |
|------|---|-----|
| 4.23 | Ponto crítico do caminho onde existe um objeto estático em uma curva seguindo um objeto em movimento | 105 |
| 5.1 | Diagrama de blocos da proposta de implementação do Risco para o condicionamento da cooperação | 107 |
| 5.2 | Sinais do controlador MPC e do controlador do condutor para referências diferentes $y_{mpc} = 3,5$ $y_{condutor} = 0$ | 108 |
| 5.3 | Predição de sinais de controle para referências diferentes $y_{mpc} = 3,5$ $y_{condutor} = 0$ | 109 |
| 5.4 | Predição da Curvatura para os diferentes esforços de controle | 110 |
| 5.5 | Sinais do controlador MPC e do controlador do condutor para referências diferentes $y_{mpc} = 0.3$ $y_{condutor} = 0$ | 114 |
| 5.6 | Posição atual estimada do veículo $(\hat{x}_{car}(k), \hat{y}_{car}(k))$ em pontos vermelhos e as trajetórias estimadas (azul) do que deseja fazer o condutor em um horizonte cinemático de $25m$ | 114 |
| 5.7 | Curvatura na simulação de trajetórias diferentes | 116 |
| 5.8 | Predição da trajetória desejada pelo condutor | 116 |
| 5.9 | Predição da trajetória desejada do condutor no horizonte cinemático | 117 |
| 5.10 | Simulação da medição do risco condicionamento da cooperação entre o condutor e o sistema autônomo. | 118 |
| 6.1 | Caracterização estática do motor da direção. | 120 |
| 6.2 | Localização do extensômetro no eixo do volante | 121 |
| 6.3 | Medidas do extensômetro durante o processo de calibração | 121 |
| 6.4 | Relação entre Nm e $\mu m/m$ do extensômetro. | 122 |
| 6.5 | Diagrama de blocos da instrumentação do sensor de torque | 123 |
| 6.6 | Acelerômetros durante percurso um filtrados e sem filtrar. | 124 |
| 6.7 | altitude, ângulos de rolagem e inclinação estimadas durante o percurso um | 125 |
| 6.8 | Altitude, ângulos de rolagem e inclinação estimadas da estrada para o percurso um | 125 |
| 6.9 | Comparação entre o gps e rota planajeda pelo Google Maps no recorrido um. | 126 |
| 6.10 | Comparação entre o gps e rota planajeda pelo Google Maps no recorrido dois. | 127 |
| 6.11 | Sensores de velocidade do VILMA | 128 |
| 6.12 | Velocidade longitudinal estimada percurso1 | 129 |
| 6.13 | Velocidade angular estimada com a velocidade das rodas | 129 |
| 6.14 | Modelo cinemático para o percurso 1 | 130 |
| 6.15 | Problema no ângulo das rodas por controle cooperativo | 130 |
| 6.16 | Ângulo das rodas estimada do percurso um | 132 |
| 6.17 | Estimação da posição do EKF com o modelo dinâmico | 133 |
| 6.18 | Variáveis de estado estimadas do percurso 1 | 133 |

| | | |
|------|---|-----|
| 6.19 | Diagrama de bloco do controle de seguimento de linhas da estradas | 134 |
| 6.20 | Controle Baseado em imagem | 137 |
| 6.21 | Zona morta no controle de velocidade do veículo | 138 |
| 6.22 | Tempo e velocidade do percurso dois | 140 |
| 6.23 | Ângulo estimado das rodas no percurso dois | 140 |
| 6.24 | Controle de velocidade longitudinal | 142 |
| 6.25 | Controle de trajetória cooperativo no percurso 2 | 143 |
| 6.26 | Controle de trajetória cooperativo no percurso 2 | 144 |
| A.1 | Velocidades angulares requeridas em testes dinâmicos CU short course e ISO 3888 Part 2-2002 com referências de Fishhook Rate e J-turn Rate. Duração do teste J-Turn entre 287-401ms e do teste Dishook entre 647-906ms(FORKENBROCK <i>et al.</i> , 2003b) | 161 |
| A.2 | Ângulo da direção vs. Torque da direção no simulador apresentado em (ILHWAN <i>et al.</i> , 2008) | 162 |
| A.3 | Torque da direção requerido em sistema EPAS ZF servoelectric(ZF, 2012) | 162 |
| A.4 | Curva Boost Curve com diferentes áreas de trabalho dependendo do tipo de viagem.(ROESTH, 2007) | 163 |
| A.5 | Teoria de operação do motor da direção. fonte (ANIMATICS, 2014) | 165 |
| A.6 | Localização do motor de direção. | 166 |
| A.7 | Modelo 3D do sistema de direção. | 167 |
| A.8 | Projeto mecânico da transmissão de movimento entre o motor e barra de direção. | 168 |
| A.9 | Automação do acelerado.r | 170 |
| A.10 | Distribuição da força máxima do pedal do freio pelas condições do pavimento. Fonte (NHT, 2003) | 173 |
| A.11 | Diagrama do sistema eletro-hidráulico do ECU do freio. Adaptado de (REIF, 2014) | 174 |
| A.12 | Diagrama hidráulico do Estado 0 de funcionamento do freio | 175 |
| A.13 | Diagrama hidráulico do Estado 1 de funcionamento do freio. | 175 |
| A.14 | Diagrama hidráulico do Estado 2 de funcionamento do freio. | 176 |
| A.15 | Diagrama hidráulico do Estado 3 de funcionamento do freio. | 176 |
| A.16 | Diagrama hidráulico do Estado 4 de funcionamento do freio. | 177 |
| A.17 | Arquitetura de processamento do MicroAutobox II. Fonte do folheto do fabricante. | 181 |
| B.1 | Software do sistema embarcado do microautobox. Linha preta (sinal elétrico) e azul (vetor de dados) | 184 |
| B.2 | Estados do envio de sinais para a troca de marcha. a troca de estados é comandada por dois sinais: comando e valor | 187 |
| B.3 | Estados no envio de pacotes ao ECU do freio pela rede CAN | 188 |

| | | |
|------|--|-----|
| B.4 | Diagrama da comunicação UDP do microautobox. | 190 |
| B.5 | Comportamento de uma variável recebido no protocolo maudp. Azul sinal onde foi enviada verde(o marrom) sinal sem erros, vermelha sinal com erro. | 192 |
| B.6 | Estados do sistema embarcado | 193 |
| C.1 | . Diagrama de interação entre os nós do ROS no experimento de testes de hardware. | 198 |
| E.1 | Hardware dos testes do sistema de automação do veículo | 203 |
| E.2 | Interface do programa SmartMotor Interface para programação do motor Animatics | 205 |
| E.3 | Configuração porta RS232 no smartMotorInterface | 205 |
| E.4 | Configuração porta RS485 no smartMotorInterface | 209 |
| E.5 | Monitorar as variáveis do motor no smartMotorInterface | 209 |
| E.6 | Programa ControlDesk para depuração do programa do sistema embarcado | 211 |
| E.7 | Programa <i>rqt</i> com <i>plugins</i> para monitorar informação do ROS | 212 |
| E.8 | Programa <i>rqt_bag</i> para reprodução de dados adquiridos previamente | 213 |
| G.1 | Desenho do modelo cinemático da bicicleta | 226 |
| G.2 | Diagrama de blocos do modelo dinâmico do veículo | 227 |
| G.3 | Desenho da dinâmica planar da bicicleta | 228 |
| G.4 | Ângulos de derrapagem na rodas | 230 |
| G.5 | Forças laterais das rodas dianteiras e traseiras | 232 |
| G.6 | Torque de alinhamento da roda dianteira equivalente do veículo. | 233 |
| G.7 | Diagrama de bloco dos sistemas do modelo mecânico da direção. | 234 |
| G.8 | Desenho do sistema mecânico da direção | 235 |
| G.9 | Desenho dos ângulos das uniões universais no sistema de direção | 235 |
| G.10 | Desenho esquemático do sistema hidráulico da direção | 236 |
| G.11 | Desenho do Modelo Dinâmico Simplificado da direção | 237 |
| G.12 | Modelo de bicicleta no sistema de coordenadas da trajetória | 238 |

Lista de Tabelas

| | | |
|------|--|-----|
| 3.1 | Índices de desempenho do controlador com os parâmetros padrão. | 64 |
| 3.2 | Índices de desempenho do controlador, sem o condutor ($\beta = 2000$), com respeito a velocidade desejada. dados de $\dagger Q_{mpc}^{4,4} = 1600$, $\star Q_{mpc}^{1,1} = 200$ | 69 |
| 3.3 | Índices de desempenho do controlador com diferentes sensores no estimador de estados. | 70 |
| 3.4 | Índices de desempenho do controlador com diferentes configurações do controlador do condutor a $v_x = 10m/s$ | 72 |
| A.1 | Parâmetros requeridos do motor de direção | 160 |
| A.2 | Motores comerciais pré-selecionados | 164 |
| A.3 | Sistemas de controle de motores | 164 |
| A.4 | elementos do sistema de transmissão de movimento da direção | 168 |
| A.5 | Sinais de comando da alavanca de comando de câmbios Dualogic. Fonte Manual de diagnóstico do câmbio Dualogic Fiat. | 171 |
| A.6 | Estados principais do freio para o funcionamento do freio autônomo. | 174 |
| A.7 | Norma técnica correspondente a cada camada do modelo OSI para rede CAN na função de serviços de diagnostico. | 177 |
| A.8 | Identificador de serviço do protocolo ISO 14229. Adaptado da norma técnica. | 180 |
| A.9 | Caraterísticas técnicas do sistema embarcado 1401/1511/1512 do MicroAutobox II. Fonte do folheto do fabricante. | 182 |
| A.10 | Sinais de entrada/saída utilizadas no MicroAutobox. | 183 |
| B.1 | Variáveis transmitidas pela ECU do freio pela rede CAN | 188 |
| B.2 | Variáveis dos sensores do carro | 189 |
| B.3 | Variáveis do sensores do freio | 189 |
| B.4 | Tipos de dados enviados pelo microautobox | 191 |
| B.5 | Tipos de dados enviados pelo microautobox | 191 |
| B.6 | Erros no sistema embarcado | 194 |
| D.1 | Testes funcionais na automação do veículo | 200 |
| D.2 | Testes funcionais nas funcionalidades do sistema embarcado do veículo | 201 |
| D.3 | Testes funcionais nas funcionalidades do sistema embarcado do veículo | 202 |

Lista de Abreviaturas e Siglas

Modelo Dinâmico e cinemático do veículo

| | | |
|---------------|---|---|
| v_x | - | Velocidade longitudinal do veículo |
| v_y | - | Velocidade lateral do veículo |
| θ | - | Ângulo de Guinada do veículo |
| γ | - | Velocidade angular do veículo |
| τ_{user} | - | Torque do condutor |
| v_m | - | Voltagem efetivo do motor de direção |
| f_{fy} | - | Força lateral do pneu dianteiro |
| f_{ry} | - | Força lateral do pneu traseiro |
| τ_{fa} | - | Torque de desalinhamento da roda dianteira |
| δ_j | - | Angulo da barra de união da direção |
| δ_f | - | Angulo de direção equivalente das roda dianteiras |
| m_v | - | Masa do veículo |
| J_v | - | Momento de inercia do veículo |
| L | - | Distancia entre eixos do veículo |
| l_f | - | Distancia entre eixo dianteiro e o CG |
| l_r | - | Distancia entre eixo traseiro e o CG |
| c_r | - | Rigidez lateral roda traseira |
| c_f | - | Rigidez lateral roda dianteira |
| t_p | - | Pneumatic trial |
| R | - | resistência de armadura |
| k_E | - | Ganho de voltagem do motor de direção |
| k_i | - | Ganho de corrente torque do motor de direção |
| r_g | - | Ratio de redução de velocidade do servomotor de direção |
| η | - | Eficiência dos engrenagem do servomotor de direção |
| r_h | - | Ganho linear de torque da direção hidráulica |

Vetores de Estado

| | | |
|-------------------|---|---|
| x_{car} | - | Variáveis de estado do modelo dinâmico do veículo |
| x_s | - | Variáveis de estado do sistema de direção do veículo |
| x_{ref} | - | Variáveis de estado do modelo cinemático do veículo |
| x_e | - | Variáveis de estado do modelo dinâmico do veículo referenciado a um caminho de referencia |
| x_c | - | Variáveis de estado da representação de estados discreta utilizado para controle de trajetórias |
| \hat{x}_k | - | Variáveis de estado da representação de estados discreta utilizado para o estimador de estados |
| x_{car2} | - | Variáveis de estado do objeto em movimento detetado pelo laser |

Espaço curvilíneo

| | | |
|------------|---|---|
| s | - | Variável independente do arco da distância |
| q | - | deslocamento lateral no espaço curvo |
| \dot{q} | - | derivada do deslocamento lateral no espaço curvo |
| κ | - | Curvatura do caminho |
| C_s | - | Custo de segurança do caminho sobre objetos estáticos |
| C_m | - | Custo de segurança de caminhos sobre objetos em movimento |
| C_κ | - | Custo de suavidade do caminho |
| C_c | - | Custo de consistência |

Siglas

| | |
|---------------|--|
| ADAS | - Advanced Driver Assistance Systems Direção Ativa. |
| AFS | - Active Front Steering |
| CG | - Centro de gravidade do veículo |
| EBD | - Distribuição Eletrônica de Frenagem. Electronic Brake Distribution |
| ECU | - Sistema de Controle Eletrônico. Electronic Control Unit |
| EKF | - Filtro de Kalman Estendido. Extended Kalman Filter |
| EPAS | - Electric Power Assisted Steering |
| ESP | - Programa Eletrônico de Estabilidade. Electronic Stability Program. |
| GPS | - Global Position System |
| JSON | - JavaScript Object Notation |
| LIDAR | - Light Detection And Ranging |
| LMA | - Laboratório de Mobilidade Autônoma |
| MA | - MicroAutobox |
| MPC | - Controle preditivo Generalizado. Model Predictive Control |
| ODE4 | - Ordinary Differential Equation |
| PC | - Computador Pessoal . Personal Computer |
| PID | - Controlador Proportional-Integral-Derivative |
| PWM | - Modulação por Largura de Pulso. Pulse-Width Modulation |
| ROS | - Robotics Operating System |
| SPLINE | - Curva definida matematicamente por dois ou mais pontos de controle |
| UDP | - User Datagram Protocol |
| VILMA | - Veículo Inteligente do LMA |

SUMÁRIO

| | |
|--|-----------|
| Lista de Ilustrações | 10 |
| Lista de Tabelas | 15 |
| Lista de Abreviaturas e Siglas | 16 |
| SUMÁRIO | 19 |
| 1 Introdução | 24 |
| 1.1 Veículos Autônomos | 25 |
| 1.2 Sistemas de assistência ao condutor (ADAS) | 26 |
| 1.3 Motivação da Pesquisa | 28 |
| 1.4 Objetivos | 31 |
| 1.5 Estrutura do Trabalho | 34 |
| 2 Arquitetura proposta para o VILMA01 | 36 |
| 2.1 Arquitetura de Hardware | 38 |
| 2.2 Arquitetura de Software | 38 |
| 2.3 Arquitetura de hardware e software no ROS | 42 |
| 2.4 Considerações finais | 43 |
| 3 Controle de trajetórias | 44 |
| 3.1 Bibliografia | 44 |
| 3.1.1 Controle de direção do veículo | 45 |
| Sistemas de assistência elétrica | 45 |
| <i>Steer-by-wire</i> | 47 |
| 3.1.2 Controle de trajetórias | 49 |
| 3.2 Controle Preditivo baseado em modelo | 50 |
| 3.2.1 Predição das Saídas de Controle | 51 |
| 3.2.2 Otimização analítica no caso sem restrições | 53 |
| 3.2.3 Modelo dinâmico linear do veículo | 56 |
| 3.3 Estimador de estados | 57 |
| 3.4 Indicadores de desempenho do controle de trajetórias | 59 |
| 3.5 Simulações do controle cooperativo de trajetórias | 61 |

| | | |
|----------|---|------------|
| 3.5.1 | Simulações a diferentes velocidades | 68 |
| 3.5.2 | Simulações com outros sensores no estimador de estados | 71 |
| 3.5.3 | Simulações com diferentes estratégias de controle do condutor | 74 |
| 3.6 | Considerações finais | 75 |
| 4 | Planejamento de Trajetórias | 76 |
| 4.1 | Construção do espaço curvilíneo global | 77 |
| 4.2 | Localização no espaço curvilíneo global | 79 |
| 4.3 | Geração dos caminhos candidatos | 79 |
| 4.3.1 | Transformação entre o espaço curvilíneo e o cartesiano | 80 |
| 4.3.2 | Função matemática dos caminhos candidatos | 81 |
| 4.4 | Seleção do caminho | 83 |
| 4.4.1 | Custo por risco intrínseco | 84 |
| 4.4.2 | Custo pelo risco de consistência | 84 |
| 4.4.3 | Custo pelo risco extrínseco sobre objetos estáticos | 85 |
| 4.4.4 | Custo pelo risco extrínseco sobre objeto em movimento | 86 |
| 4.5 | Perfil de velocidade | 88 |
| 4.5.1 | Condições de risco intrínseco | 89 |
| 4.5.2 | Condições de risco extrínsecos por obstáculos | 89 |
| 4.5.3 | Acelerações longitudinais | 90 |
| 4.6 | Simulação do planejamento de trajetórias | 90 |
| 4.6.1 | Simulação do LIDAR | 91 |
| 4.6.2 | Detecção de objetos | 93 |
| 4.6.3 | Geração de caminhos candidatos | 96 |
| 4.6.4 | Seleção de trajetórias | 97 |
| 4.7 | Considerações finais | 104 |
| 5 | Interação do condutor | 106 |
| 5.1 | Controle de trajetórias cooperativo para referências diferentes | 108 |
| 5.2 | Horizonte dinâmico | 109 |
| 5.3 | Horizonte cinemático | 110 |
| 5.4 | Detecção de trajetória diferente | 113 |
| 5.5 | Simulação | 115 |
| 5.6 | Considerações finais | 118 |
| 6 | Resultados Experimentais | 119 |
| 6.1 | Resultados da instrumentação do VILMA01 | 119 |
| 6.1.1 | Modelo estático do motor de direção | 119 |

| | | |
|----------|---|------------|
| 6.1.2 | Calibração do sensor de torque da direção | 120 |
| 6.1.3 | Acelerômetros e barômetro | 124 |
| 6.1.4 | Sensor GPS | 125 |
| 6.1.5 | Estimação da velocidade longitudinal do veículo | 128 |
| 6.1.6 | Modelo cinemático do veículo | 129 |
| 6.1.7 | Localização com o filtro de Kalman Estendido (EKF) | 131 |
| 6.1.8 | Recomendações | 132 |
| 6.2 | Controle de trajetórias do VILMA01 baseado em imagens | 134 |
| 6.2.1 | Percepção | 135 |
| 6.2.2 | Planejamento | 136 |
| 6.2.3 | Controle de movimento | 136 |
| 6.2.4 | Controle Cooperativo | 138 |
| 6.3 | Resultados do VILMA01 com condicionamento de controle cooperativo | 139 |
| 6.3.1 | Teste do Controle de Velocidade longitudinal | 141 |
| 6.3.2 | Teste de condicionamento de controle cooperativo | 141 |
| 6.3.3 | Teste de condicionamento de planejamento cooperativo | 142 |
| 7 | Conclusão | 145 |
| | Referências | 148 |
| | Referências Bibliográficas | 148 |
| | APÊNDICES | 159 |
| A | Automação do veículo autônomo | 160 |
| A.1 | Automação do volante da direção | 160 |
| A.1.1 | Seleção do atuador de direção | 160 |
| | Requerimentos de velocidade do atuador de direção | 161 |
| | Requerimentos de Torque do atuador de direção | 161 |
| | Opções Comerciais de Atuador | 163 |
| A.1.2 | Projeto Mecânico | 165 |
| A.1.3 | Firmware do servomotor | 167 |
| A.2 | Automação da acelerador do veículo | 170 |
| A.3 | Automação da troca de marcha | 170 |
| A.4 | Automação da frenagem do veículo | 171 |
| A.4.1 | Sistema eletro-hidráulico de frenagem | 172 |
| A.4.2 | Comunicação com o módulo de frenagem | 177 |
| A.5 | Sistema embarcado | 179 |

| | | |
|----------|---|------------|
| B | Software no sistema embarcado MicroAutobox | 184 |
| B.1 | Funções da direção | 185 |
| B.2 | Função do acelerador | 186 |
| B.3 | Função da troca de marcha | 186 |
| B.4 | Função do Freio | 187 |
| B.5 | Função de sensores | 188 |
| B.6 | Função da comunicação UDP | 189 |
| B.7 | Função da máquina de estados | 192 |
| C | Software desenvolvido em ROS | 195 |
| D | Testes da arquitetura de hardware e software da Automação | 199 |
| E | Programação, Execução e depuração da automação do veículo | 203 |
| E.1 | Modo de execução | 203 |
| E.2 | Modo Programação | 204 |
| E.2.1 | Motor de direção | 204 |
| E.2.2 | Sistema Embarcado do MicroAutobox | 206 |
| E.2.3 | Programas do computador com Linux no MicroAutobox | 207 |
| E.2.4 | Instrumentação do extensômetro da direção (MicroStrain) | 208 |
| E.2.5 | Sensor de GPS u-blox | 208 |
| E.3 | Modo Depuração | 208 |
| E.3.1 | Motor de direção | 208 |
| E.3.2 | Câmera Axis | 210 |
| E.3.3 | Sistema Embarcado do MicroAutobox | 210 |
| E.3.4 | Programas do computador com Linux no MicroAutobox | 211 |
| F | Desenho tecnico da automação da direção | 215 |
| G | Modelos Matemáticos do veículo | 225 |
| G.1 | Modelo Cinemático do Veículo | 225 |
| G.2 | Modelo Dinâmico do Veículo | 227 |
| G.2.1 | Dinâmica Planar do Veículo | 227 |
| G.2.2 | Modelo dos Pneus | 230 |
| G.2.3 | Sistema Mecânico da Direção | 231 |
| | Atuador da Direção | 232 |
| | Sistema Mecânico da Direção | 233 |
| | Sistema Hidráulico | 236 |
| | Representação de Estados do sistema de direção | 237 |

| | | |
|-----|--|-----|
| G.3 | Modelo dinâmico do carro referenciado ao modelo cinemático | 238 |
|-----|--|-----|

1 Introdução

Auxiliar humanos a partir de sistemas automatizados, semi-autônomos e autônomos é uma tendência decorrente da evolução tecnológica. Atualmente, a pesquisa na área de navegação autônoma e semi-autônoma tem motivado muitos pesquisadores de diferentes grupos devido ao desafio que representa. Digno de nota é o número de publicações nos últimos anos, incorporando os desenvolvimentos da área de telemática (MIRANDA NETO, 2007). A segurança intrínseca e extrínseca do veículo é a parte mais importante na hora de projetar um veículo autônomo terrestre (PARK *et al.*, 2010), assim como nos sistemas de assistência ao condutor.

A estratégia mais frequente para transferências de controle em situações de risco é ter um botão de emergência para dar o controle ao condutor, porém esta pode não ser a melhor solução. Mostra de isso é a patente de Google (CULLINANE *et al.*, 2014) que checa todas as variáveis de segurança prévio a fazer transferências entre automático e manual, ou perguntas como as que tem o gerente de ADAS e percepção da BMW: *The great problem is how can we handle critical situations where there is not enough time to give back -Control- to the driver* (LAURSEN, 2013). Nesta ordem de ideias, o presente trabalho procura pesquisar estratégias de cooperação entre o condutor e o sistema de controle do veículo autônomo especificamente no controle na direção do veículo. O sistema projetado deseja que o condutor não seja assistido só na atuação do volante, mas também na trajetória mais segura a realizar.

No ano 2008, interessado na robótica e na segurança veicular, o laboratório de mobilidade autônoma (LMA) foi formado no departamento de mecânica computacional da faculdade de engenharia mecânica (DMC-FEM) da UNICAMP. A criação do grupo possibilitou o desenvolvimento de projetos focados na área de percepção (MIRANDA NETO, 2007) (ARTHUR DE MIRANDA, 2011) (BERNARDES VITOR, 2014) (RODRIGUEZ RUIZ, 2014) e depois em algumas áreas de controle em veículos em escala (HERNÁNDEZ, 2012) (CÁRDENAS RUEDA, 2013). Em 2011 com a doação de um veículo pela Fiat deu início ao projeto do primeiro veículo autônomo do grupo chamado de VILMA (Veículo inteligente do LMA). O projeto iniciou com a necessidade de desenvolver a automação do veículo, estudar diferentes arquiteturas de hardware e software, assim como técnicas de controle e planejamento de trajetórias, para o desenvolvimento de um veículo de navegação terrestre.

A seguir é apresentada uma abordagem geral do desenvolvimento do tema, seguida de uma justificativa da pesquisa baseada nos acidentes no Brasil. Além disso, são apresentados os objetivos e a organização do trabalho.

1.1 Veículos Autônomos

A pesquisa em veículos autônomos iniciou na década dos anos 60 com trabalhos como o realizado na universidade de Stanford, o *Stanford Cart*, o qual era um veículo com motores elétricos e rodas de bicicleta e uma câmera de vídeo fixa onde as imagens captadas eram processadas para controlar o esterçamento das rodas (FERNÁNDEZ-MADRIGAL, 2012). Os trabalhos foram progredindo com o aumento da capacidade dos computadores para realização de processamento de imagens, aumentando o interesse de empresas privadas e de entidades governamentais.

Em consequência a isso, a agência DARPA (Defense Advanced Research Projects Agency) organizou nos anos 2004 e 2005 um desafio conhecido como Grand Challenge que consistiu em fazer um recorrido autônomo em um deserto na área de California. Na primeira vez, nenhuma das equipes conseguiu concluir com sucesso o percurso definido para a competição. Já em 2005 cinco equipes completaram o desafio, sendo a vencedora a equipe da Universidade de Stanford, Stanford Racing. Na ocasião do Grand Challenge 2005, o desafio foi a travessia do deserto de Mojave. A equipe vencedora completou a prova com uma velocidade média de 30.7 km/h. O último desafio ocorreu em novembro de 2007. Conhecido como DARPA Urban Challenge, este desafio caracterizou-se por veículos autônomos que administraram suas missões em uma falsa área urbana. As etapas do desafio foram subdivididas em uma série de passos de qualificação, que conduziam para um evento final que aconteceu em Victorville, CA. Foram trinta e cinco equipes semifinalistas, onze finalistas e o primeiro lugar ficou com a equipe Tartan Racing, Pittsburgh, PA, o segundo lugar com a equipe Stanford Racing, Stanford, CA e o terceiro lugar com a equipe Victor Tango, Blacksburg, VA.

Durante o desafio urbano foi possível testar a habilidade dos veículos para operarem com segurança e efetividade, interagindo com outros veículos dentro e ao redor de um ambiente urbano. O percurso teve uma distância total de aproximadamente 96 km, devendo ser completado num período de até 6 horas. Além do dever de seguir as leis de trânsito do local, as equipes tiveram como desafios: variações na qualidade do pavimento das estradas e vias, obstáculos como: barreiras, hidrantes, pedras, equipamento de construção, fios de alta tensão, além dos outros obstáculos estáticos encontrados num ambiente urbano. Uma dificuldade adicional foram os bloqueios erguidos durante o curso do evento, além das interferências com o sinal do sensor GPS (Global Position System) causadas por árvores e edifícios ao longo do caminho.

Apesar de não aparecerem novos desafios pelo DARPA, os desafios realizados possibilitaram a geração de uma bibliografia intensa muito importante para o desenvolvimento da área,

tanto no setor acadêmico como no setor industrial. Depois disso surgiram outros desafios na área por outras empresas como *Hyundai Challenge* (CHU *et al.*, 2012) e o *Grand Cooperative Driving Challenge* (GEIGER *et al.*, 2012). Empresas e pesquisadores seguiram investindo em veículos autônomos como o carro de Google, desenvolvido pelos principais pesquisadores que participaram do Darpa Challenge, o qual em 2014 já apresentava mais de 14000 milhas de condução autônoma, e também de resultados de pesquisa em universidades como o laboratório Vislab da universidade de Parma na Itália que fez um percurso de 13000km da Italia até China(BERTOZZI *et al.*, 2013).

Na área comercial a maioria dos fabricantes de veículos tem concentrado parte de seus recursos de pesquisa e desenvolvimento nesta área, os quais junto com políticas governamentais tem desenvolvido projetos como o SARTRE da União Europeia, que tem como objetivo permitir a navegação em comboio, ou seja, permitir que os veículos possam juntar-se a um comboio de carros que seguem de forma autônoma um veículo na frente para poupar combustível e liberar o condutor de dirigir.

No Brasil, a temática de veículos terrestres em ambientes externos tem sido desenvolvido em universidades e centros de pesquisa, entre os mais reconhecidos temos o Projeto VERO (CORDEIRO *et al.*, 2012) do Centro de Tecnologia da Informação Renato Archer (CTI), o Projeto CADU do Laboratório de Sistemas de Computação e Robótica da Universidade Federal de Minas Gerais (DIAS; ARRUDA, 2013; 2012) e o projeto CaRINA (FERNANDES *et al.*, 2014) na ICMC da universidade de São paulo em São Carlos/SP . Empresas começam a se estruturar, focando inicialmente em sistemas tele-operados na indústria agrícola, mas procurando a evolução para soluções autônomas, que representam o diferencial tecnológico a ser alcançado.

1.2 Sistemas de assistência ao condutor (ADAS)

Sistemas de assistência ao condutor são cada vez mais diversos e a um preço menor, sendo cada dia mais comuns os sistemas com percepção do meio externo que são consideradas como o passo intermediário para atingir uma navegação autônoma. Somente como exemplo do estado do arte - pois funções similares são encontradas em outras montadoras - o acessório *intelligent Drive suite* dos Mercedes Benz classe S, E e C (veículos de US\$42000) tem o valor US\$3000, permitindo controlar de forma autônoma e semi autônoma os freios, o acelerador e a direção através de uma malha de mais de 20 sensores que inclui câmeras stereo, sensores ultrassônicos e radares que permitem ter uma percepção nos 360 graus do veículo. As funções que utilizam percepção externa entre outras são:

- **Detector ativo de troca de Faixa.** Os sensores de radar verificam o tráfego ao redor do veículo, enquanto uma câmera verifica se as linhas de contorno da estrada, tanto contínua e descontínua são excedidos. Se uma alta probabilidade de colisão é detectada durante a mudança de faixa, o detector ativo de mudança de faixa vai intervir seletivamente nos freios das rodas do veículo para afastar o veículo da zona de perigo.
- **Freio assistido (*BAS PLUS*).** O sistema verifica os veículos e os pedestres ou objetos de forma geral à frente e otimiza automaticamente a força do freio assim que o motorista pressiona o freio. O sistema tem dois sistemas de radar. Um radar de longo alcance monitora a área à frente do veículo em até três faixas com um ângulo de varredura de 18° e um alcance de 200 m, igualmente este sensor também supervisiona a área de médio alcance à frente do veículo, com um ângulo de varredura de 60° e uma faixa de 60 m. O sistema também utiliza dois sensores de radar de curto alcance localizado atrás do pára-choques dianteiro com um ângulo de varredura de 80° e um alcance de 30 m. Este sistema de acordo com as informações da Mercedes Benz, pode reduzir na Alemanha em 36% os acidentes em rodovias.
- **Proteção ativa contra colisões.** Se a separação em relação a um veículo à frente é menor do que a distância de segurança durante alguns segundos, o sistema alerta ao condutor primeiro com um símbolo de advertência no painel de instrumentos que a separação não é adequada. Se a distancia continua diminuindo, um sinal sonoro é ativado por intervalos e o sistema de Freio Assistido é ativado.

Se, apesar de persistir o perigo de colisão, o motorista não age, o sistema inicia automaticamente uma frenagem parcial. A velocidade do veículo é substancialmente reduzida e o condutor é avisado por uma vibração. Dependendo da velocidade com que o veículo se aproxima do veículo anterior ou de um obstáculo, a intervenção autônoma no sistema de freio pode ser suficiente para evitar a colisão traseira ou atenuar as consequências do acidente.

- **Velocidade de cruzeiro adaptativa (*DISTRONIC PLUS*).** O sistema acelera e freia com potência equivalente à metade do máximo disponível para manter a distância de frenagem de segurança de forma independente. No limite de velocidade, este sistema leva em conta a situação do tráfego (congestionamento, circulação em comboio), e pode até chegar à parada total do veículo. Na classe E e S este sistema adicionalmente conduz, através de um controle de direção, a troca de faixa (deslocamento lateral).
- **Controle ativo de ângulo morto (*Active blind Spot*).** O sistema ajuda o motorista a monitorar áreas de pouca visibilidade. Avisava quando detecta um perigo e pode ajudar a prevenir acidentes por intervenção seletiva nos freios das rodas. A área de exploração do

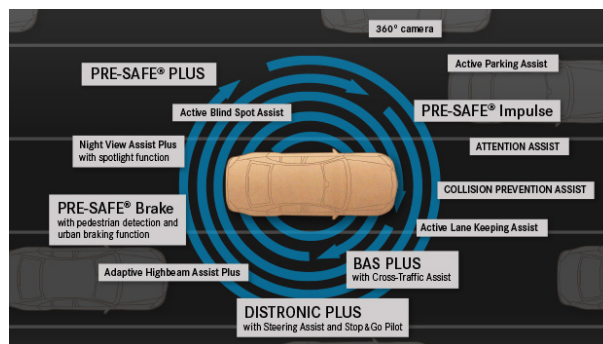


Figura 1.1: Sistema *Intelligent Drive* de Mercedes-Benz. fonte web do mercedez benz

sistema que se estende de cerca de 3,5 metros em ambos os lados do veículo e cerca de 3 metros da parte traseira do carro.

- **Unidade inteligente (*Intelligent Drive*).** Como é descrito na figura 1.1, o sistema incorpora a maioria das tecnologias descritas anteriormente sendo descrito pela Mercedes Benz como o passo prévio aos veículos autônomos.
- **Estacionamento Ativo com direção assistida (PARKTRONIC).** O sistema é responsável por dirigir a operação. Ou seja, manipulando o volante e os freios do veículo e mostrando ao motorista as intervenções necessárias, como trocar a caixa de velocidades. O motorista mantém o controle sobre todo o processo e apenas regula a velocidade com o acelerador e freio.

1.3 Motivação da Pesquisa

De acordo com as estatísticas de acidentes de 2007, mais de 1,2 milhão de pessoas morrem todos os anos nas estradas do mundo, e entre 20 e 50 milhões sofrem de lesões não fatais devido a acidentes rodoviários (ORGANIZATION, 2009). Apesar de não haver certeza quanto aos números absolutos de acidentes de trânsito que ocorrem anualmente no Brasil, os dados mostram que entre 1961 e 2000, o número de mortes em acidentes de trânsito aumentou cinco vezes, enquanto que o número de lesões foi multiplicado por quinze (DE PESQUISA ECONÔMICA APLICADA, 2003).

A diferença entre as estatísticas de acidentes de tráfego entre os países de alta renda e países em desenvolvimento são realmente significativos, por exemplo, em 2010, de acordo com a Organização Mundial de Saúde (OMS) os países entre os cinco com maior taxa de mortalidade por 100.000 habitantes são Niue (68,3), República Dominicana (41,7), a Tailândia (38,1),

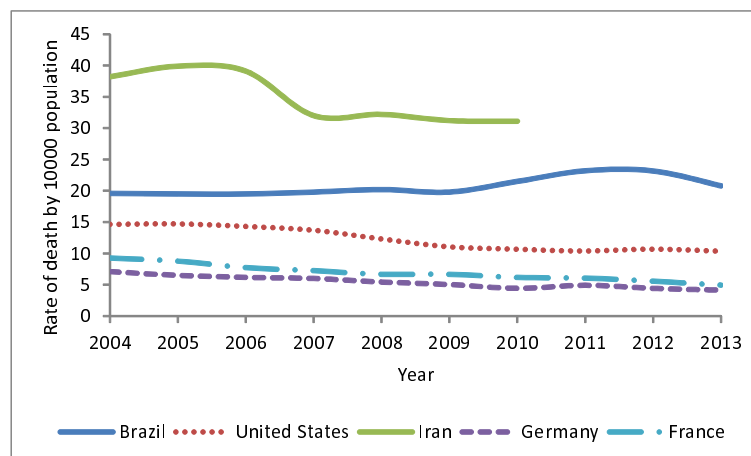


Figura 1.2: Mortes por acidentes de tráfego por 100000. fontes: Brasil(DATASUS), Estados Unidos(NHTSA), Iran (BAHADORIMONFARED *et al.*, 2013), Alemanha(destatis.de) e França(www.securite-routiere.gov.fr)

Venezuela (37,2) e Irã (34,1). Estes valores comparados com a Alemanha (4,7), a França (6,4) ou Estados Unidos (11,4), - onde o número de carros e quilômetros percorridos são maiores - devem ser relevantes no momento de adaptar a tecnologia para os países em desenvolvimento como o Brasil. Para mostrar a diferença de acidentes, embora as estatísticas por habitantes gerem controvérsia porque a quantidade de pessoas que conduzem é diferente em muitos países e o número de quilômetros percorridos por motorista também, a figura 1.2 apresenta os dados históricos dos países Brasil, Estados Unidos (EUA), Irã, França e Alemanha. É possível ver a grande diferença entre o nosso estudo de caso (Brasil) e países de alta renda.

As estatísticas de acidentes de trânsito no Brasil são geradas e apresentadas (ver figura 1.3) separadamente pelo Ministério da Saúde através da base de dados do DATASUS, a entidade gestora do seguro obrigatório DPVAT, rodovias federais (DENATRAN) e secretarias municipais. Para analisar as causas e os tipos de acidentes no Brasil podem ser usadas as estatísticas de acidentes nas rodovias federais e na cidade de São Paulo. Em rodovias federais os acidentes de trânsito aumentaram de 128.456 em 2007 a 188.925 em 2011 o que mostra um aumento de 47%. De acordo com a figura 1.4a, os tipos mais comuns de acidentes são colisão traseira, saída da pista e colisões laterais na mesma direção, que tiveram um aumento entre essas datas de 65%, 38% e 86% respectivamente. A maneira mais clara para analisar o problema é através da figura 1.4b, embora não apresente todos os tipos de acidentes, os acidentes mais frequentes (cerca de 55%) geram 17% das mortes em acidentes de trânsito e que mais do 50% das mortes são causados por acidentes do tipo atropelamento e colisão frontal que representam menos de 8% de todos os acidentes entre 2007 e 2011.

Sobre as causas do acidente (CET, 2012) são analisados dados de acidentes na cidade

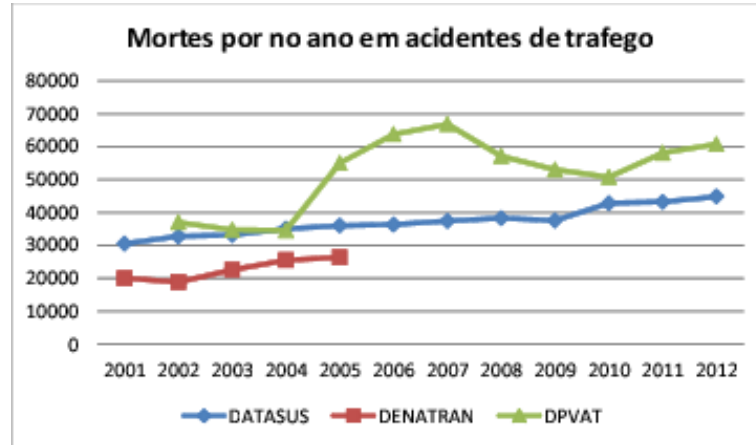


Figura 1.3: Mortes por acidentes de trafego em Brasil. fontes: ministério de saudê (DATASUS), a entidade gestora do seguro obrigatório (DPVAT), rodovias federais(DENATRAN)

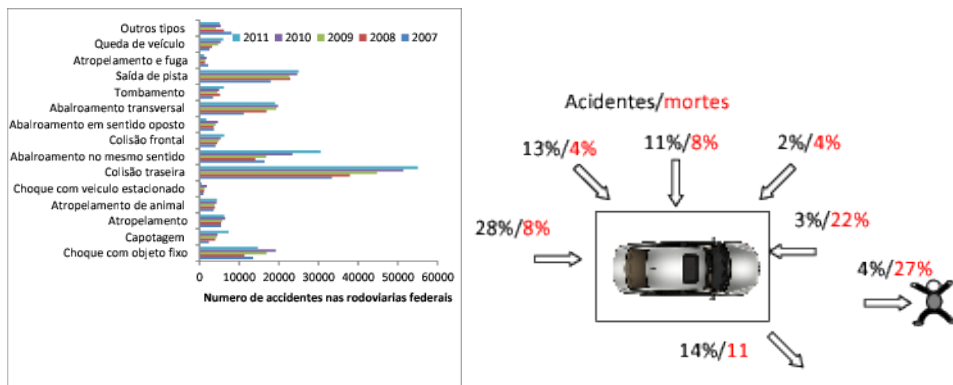


Figura 1.4: a) Número de acidentes de tráfico pelo tipo de acidentes nas rodovias federais. fonte: rodovias federais(DENATRAN) b) Número de acidentes e mortes por tipo de acidente.

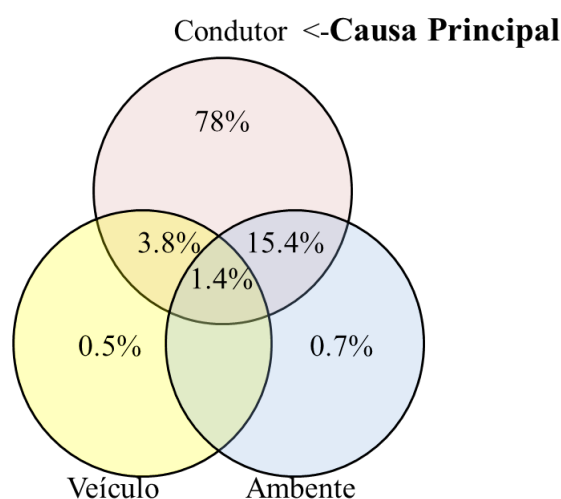


Figure 1.5: Causas de acidentes de tráfego. fonte (CET, 2012)

de São Paulo entre 2006 até o 1º trimestre de 2010, dos quais 69,3% foram atropelados e 30,7% foram colisões. O acidente ocorreu em 84,8% em vias arteriais (limites de velocidade ao redor 60km/h) e vias de trânsito rápido (limites de velocidade iguais ou maiores a 80km/h sem semáforos) onde tem-se maior volume de veículos. Ao contrário das estatísticas rodoviárias federais onde mais de 50% dos acidentes foram de veículos de passeio, na pesquisa só 36% dos acidentes fatais foram causados por veículos de passeio, seguido por acidentes de motocicleta até 31%. Sobre as causas de acordo com a figura 1.5, verificou-se que 98,6% teve uma combinação de erro humano, falha do veículo e problemas ambientais, dos quais 78% dos acidentes foram causados por fator humano. A maior causa de acidentes por fatores humanos está relacionada com a alta velocidade (44%), seguido por não respeito dos semáforos (14%), dados que tem correlação com que 35% das multas de trânsito que foram causados por excesso velocidade de acordo (CET, 2011). Outro ponto de interesse é que 75% dos acidentes com vítimas (incluindo os de atropelamento) ocorreram no meio do quarteirão e não nos cruzamentos.

1.4 Objetivos

Frente à grande taxa de acidentes, o presente trabalho quer incluir indicadores de risco na cooperação entre o condutor e o sistema de controle do veículo autônomo. No controle cooperativo os sistemas de atuação estão submetidos a dois sinais, um sinal é calculado pelo sistema automático e o segundo sinal é o aplicada pelo condutor, portanto é necessário que o sistema autônomo possa ter uma previsão da trajetória que o condutor deseja realizar e o risco de executar essa trajetória, com a finalidade de condicionar o controle cooperativo, realizar um planejamento cooperativo ou realizar a transferência total do controle ao motorista. A figura 1.6

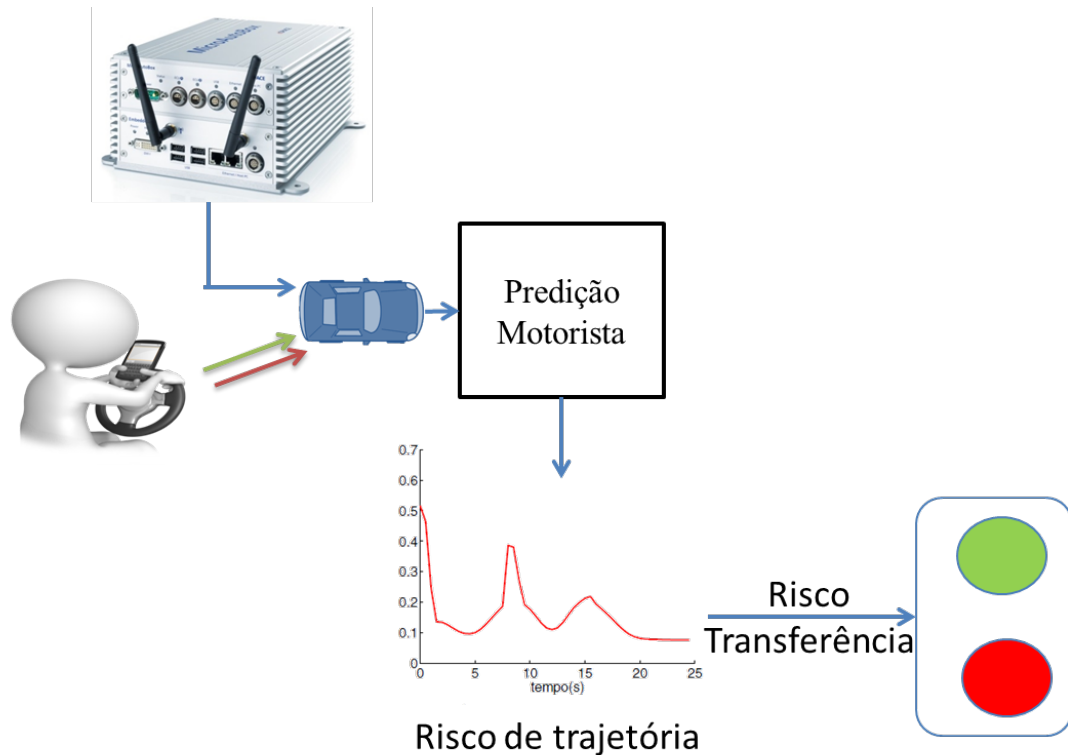


Figura 1.6: Estratégia de transferência de controle com indicadores de risco de trajetória

apresenta a proposta.

Note-se que nesta proposta é considerado a hipótese de que o sistema autônomo é mais confiável que o condutor, dado que possui toda informação do ambiente de forma redundante, não perde a concentração, tem maiores capacidades de detecção de imprevistos e reação a emergências. Em outras palavras as decisões do sistema são assumidas aqui como sendo sempre as decisões corretas. Noté-se que em todo momento o condutor terá a opção de pressionar o botão de emergência (ou transferência) e adquirir o controle total do veículo.

Como exemplo da proposta, a figura 1.7 apresenta um veículo autônomo (azul) que tem sensores que permitem conhecer o ambiente para planejar trajetórias seguras (azuis), mais por meio de horizontes de previsão baseados em modelos dinâmicos e cinemáticos do veículo é possível calcular a rota desejada pelo condutor (verde ou vermelha). Se a rota desejada pelo condutor está de acordo as regras de planejamento do sistema autônomo, ou seja, não apresenta risco (verde), o sistema autônomo vai replanejar a rota gerando um planejamento cooperativo com o desejado pelo condutor. Em caso contrário (rota vermelha), porque é uma regra do planejamento não poder ultrapassar pela direita, não vai permitir esta ação tentando continuar a rota segura planejada (azul) e vai gerar de forma antecipada algum tipo de sinal de alerta para o condutor.

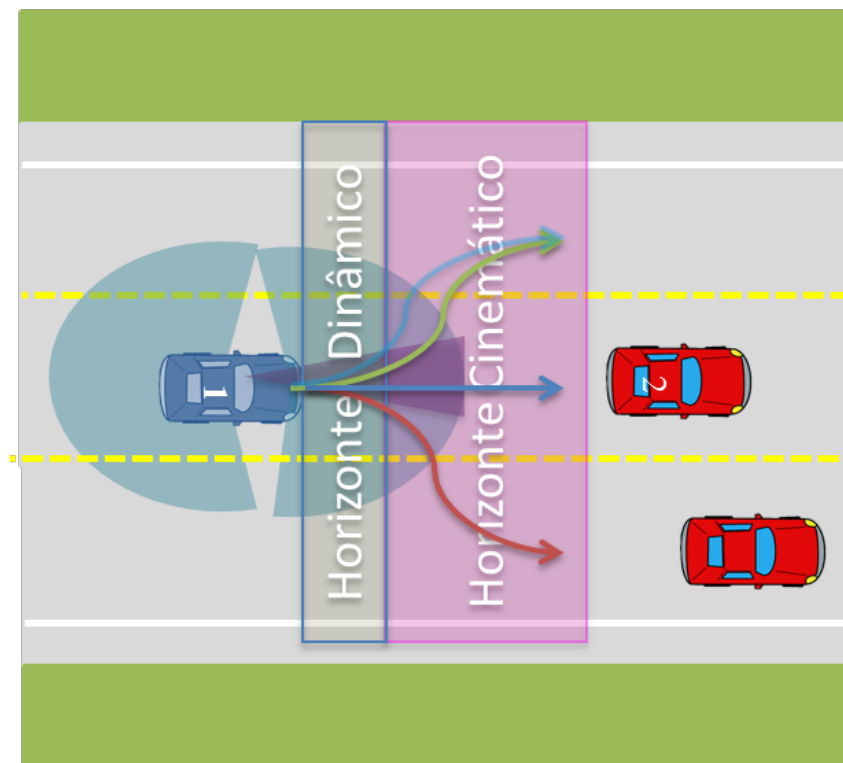


Figura 1.7: Exemplo análises de risco no condicionamento de controle cooperativo

Sobre esta proposta foi definido o objetivo principal de pesquisar estratégias de interação entre o condutor e os sistemas de controle autônomo na direção para o condicionamento de controle cooperativo, planejamento cooperativo e as transferências de controle entre Automático e Manual. Para desenvolver a proposta foram criados os seguintes objetivos específicos:

- Estudar modelos cinemáticos e dinâmicos do veículo e da automatização do mesmo.
- Propor uma estratégia de controle cooperativo de trajetórias do veículo.
- Propor estratégias para analisar o risco no planejamento e controle de trajetórias.
- Estudar a interação do condutor com o veículo para prever o desejado pelo condutor e poder gerar estratégias de planejamento cooperativo.
- Projetar os sistemas de automatização do volante da direção, da frenagem, troca de marcha e da aceleração de um veículo comercial.
- Propor arquiteturas de software e hardware para controlar o veículo nos diferentes modos de operação do mesmo.

1.5 Estrutura do Trabalho

Este trabalho está organizado em sete capítulos, uma seção dedicada às referências bibliográficas, seis apêndices e um anexo.

O Capítulo 2 apresenta a arquitetura de desenvolvimento do veículo autônomo VILMA01, assim como as arquiteturas de software e hardware projetadas.

O Capítulo 3 apresenta a teoria do controle de trajetórias baseado na técnica de controle preditivo, assim como a estimação das variáveis de estado baseada no filtro de Kalman estendido.

O Capítulo 4 apresenta a técnica de planejamento local de trajetórias em um espaço curvilíneo o qual inclui análises de risco das possíveis trajetórias locais do veículo.

O Capítulo 5 apresenta a proposta de interação com o condutor de como é estimado o horizonte de predição dinâmico e cinemático e o risco de transição.

O Capítulo 6 apresenta resultados no veículo VILMA01 em dois experimentos. O primeiro que consiste em análises da instrumentação disponível do veículo para obter sua localização. O segundo resultado apresenta um controle de trajetórias do tipo reativo baseado na percepção de uma câmera para seguir a linha lateral esquerda da estrada com estratégias de condicionamento de controle e planejamento cooperativo.

O Capítulo 7 apresenta as conclusões e algumas propostas de trabalhos futuros. Posteriormente, as referências citadas no texto.

O Apêndice A apresenta o projeto mecatrônico da automação de um veículo comercial.

O Apêndice B apresenta a arquitetura do software do sistema embarcado.

O Apêndice C apresenta o software implementado em ROS para o VILMA01.

O Apêndice D apresenta os resultados dos testes funcionais da automação.

O Apêndice E apresenta os manuais gerados para programação e depuração do sistema.

O Apêndice F apresenta os desenhos técnicos da automação da direção.

O Anexo G apresenta os modelos cinemáticos e dinâmicos do veículo e do sistema de direção utilizados no trabalho.

2 Arquitetura proposta para o VILMA01

Para implementar e testar a proposta de controle cooperativo foi necessário projetar um veículo robótico autônomo, o VILMA01, que tivesse a arquitetura adequada para desenvolver os testes necessários. Partiu-se do modelo de (SIEGWART *et al.*, 2011a) que apresenta três camadas: Percepção, Planejamento e Controle de movimento; e do modelo de (THRUN *et al.*, 2006), no qual os sistemas devem ser organizados em seis principais grupos funcionais: Interface com Sensores, Percepção, Controle, Planejamento, Interface com o Veículo, e Interface com o Usuário. De acordo com (CHEN *et al.*, 2008) estes tipos de arquiteturas facilitam a separação de funções e o desenvolvimento, permitindo a definição de interfaces entre os diferentes subsistemas.

A figura 2.1 apresenta as perguntas gerais de cada camada na robótica móvel. Nesta mesma figura foi adicionado uma nova interface entre o condutor (linha Pontilhada 1) e o robô com a respectiva pergunta de risco que busca ser respondida neste trabalho.

Nesta arquitetura proposta para que o condutor possa interagir em forma cooperativa com o sistema autônomo, a informação dele deve ter interfaces com as camadas de controle e planejamento (linha Pontilhada 2), as quais devem determinar o risco da ação desejada pelo condutor para poder condicionar a cooperatividade.

As perguntas de cada uma das camadas podem ser convertidas em diagramas de blocos das funcionalidades. Baseado no diagrama da figura 2.1 foi proposta a arquitetura apresentada na figura 2.2. Note-se que nesta mesma figura são mostrados os blocos das funcionalidades e as interações das camadas que foram projetados no veículo autônomo VILMA01. Apesar de todos os blocos serem necessários para o desenvolvimento de um veículo autônomo, ao longo deste trabalho vão ser discutidos em detalhes apenas os blocos em cinza, dado que eles são os blocos que interagem com a informação da interação com o condutor para responder a pergunta de risco no condicionamento de controle cooperativo. O que foi desenvolvido da arquitetura do VILMA01 mas não está relacionado diretamente a proposta será somente mencionado ao longo de texto e detalhado nos apêndices.

A seguir é descrita a arquitetura de software e hardware da automação do VILMA01.

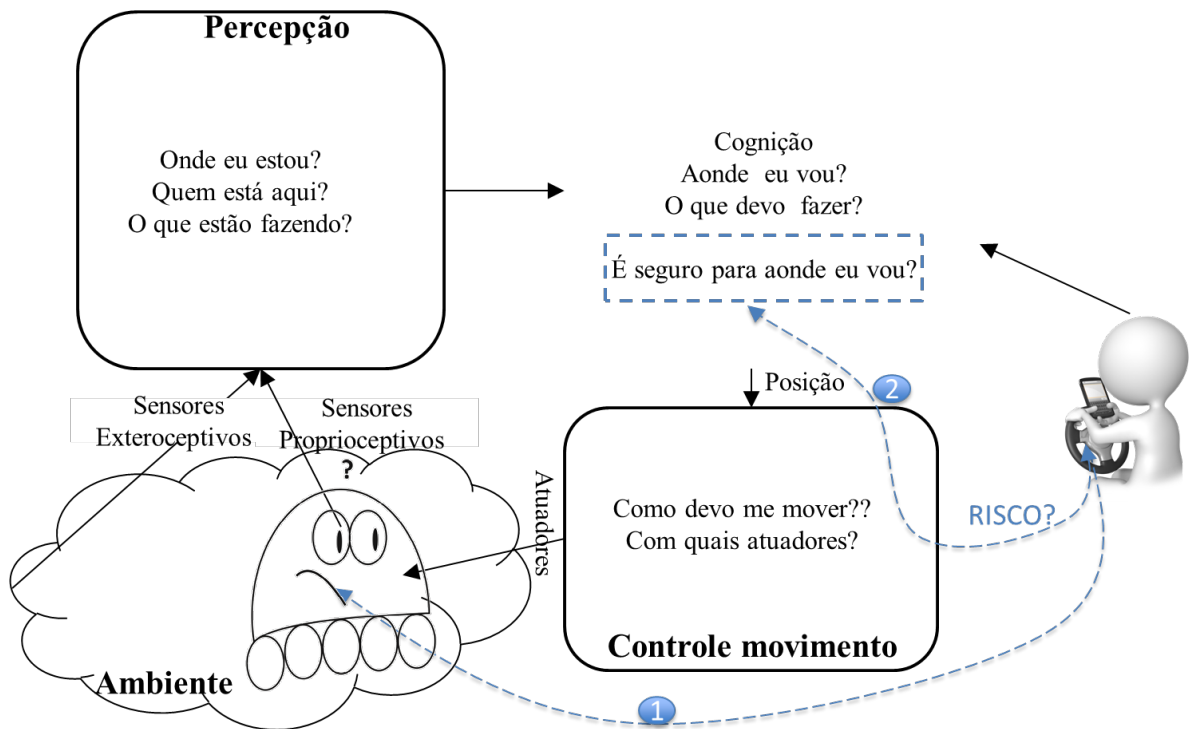


Figura 2.1: Modelo de camadas de robótica móvel. Adaptado de (SIEGWART *et al.*, 2011a)

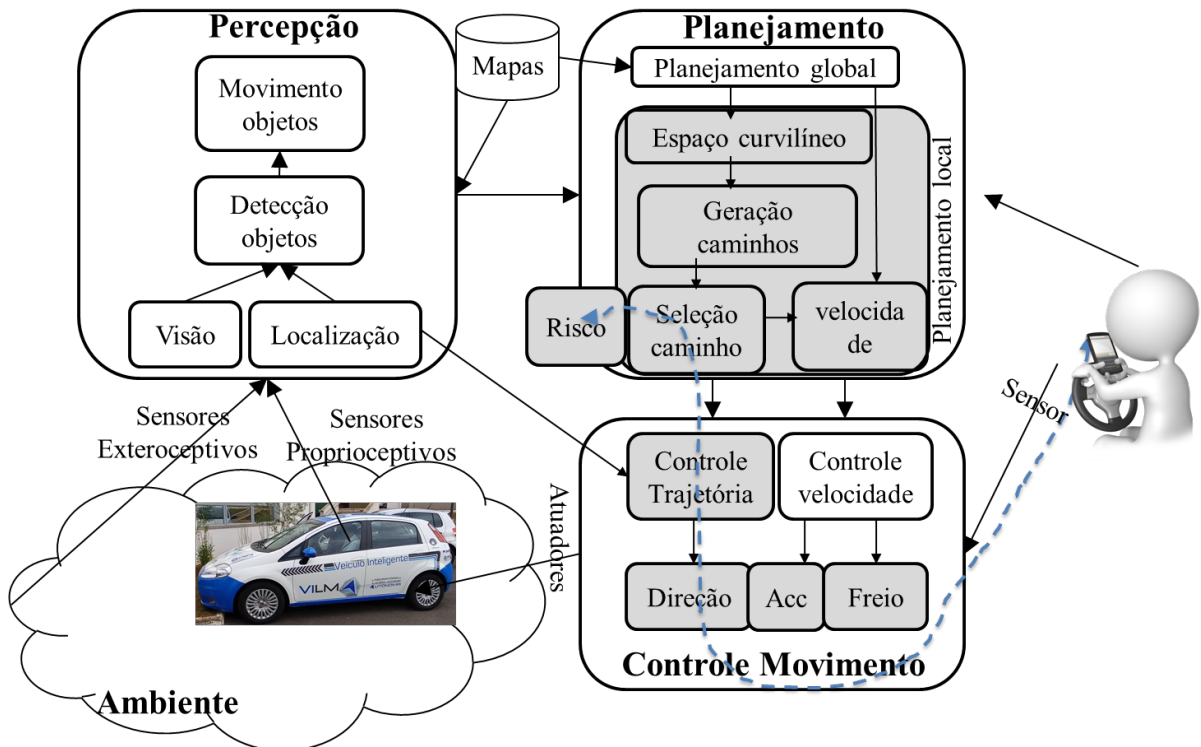


Figura 2.2: Arquitetura do VILMA autônomo no contexto da interação de áreas de conhecimento.

2.1 Arquitetura de Hardware

Apos a definição inicial dos grupos funcionais do veículo autônomo foi iniciado o projeto e o desenvolvimento da automação dos atuadores do veículo Fiat Punto 1,4V *Attractive* modelo 2011. A automação consistiu em automatizar a direção, o acelerador, o freio e a caixa de troca de marchas do veículo para serem comandados por um sistema embarcado. Na direção a automação foi realizada por meio de um motor elétrico acoplado ao eixo da direção sem fazer modificações mecânicas para não afetar a homologação do veículo. Os outros três sistemas de atuação do veículo foram automatizados a partir de circuitos eletrônicos e redes de dados. Uma descrição detalhada do projeto da automação é apresentado no apêndice A no qual também são apresentadas as características do sistema embarcado.

O projeto de automatização gerou a arquitetura de hardware do sistema de automação representada pela figura 2.3. Nela é possível distinguir que existem cinco formas de interação entre os blocos: Sinais elétrico (preto) que são entrada ou saídas digitais ou análogas, comunicação pela porta RS232 (verde) com o motor de direção, comunicação sem fio (laranja) ao sensor de torque da direção, comunicação com o freio do veículo pela rede CAN (roxa) com o sistema embarcado, e comunicação Ethernet (azul) do sistema embarcado com outros sistemas de processamento de dados. Note-se que a comunicação com o motor de direção tem redundância com a finalidade de melhorar a confiabilidade do sistema. Adicionalmente, para o posterior desenvolvimento do VILMA01 foi definido que todo dispositivo programável deveria ter um procedimento de depuração e programação. Um manual dos modos de programação, de depuração e de execução é apresentado no apêndice E.

2.2 Arquitetura de Software

A figura 2.4 apresenta a arquitetura de software proposta e implementada no VILMA01. A arquitetura foi projetada para que os softwares do veículo autônomo tivessem a capacidade de serem desenvolvidos em três arquiteturas diferentes de programação (verde, azul e preto) descritas a seguir.

- **Firmware (verde)**. Software desenvolvido diretamente no hardware dos equipamentos eletrônicos. No momento o único que está implementado diretamente foi o software do motor de direção, dado que os outros firmwares dentro da arquitetura são programados por terceiros mas devem ser configurados e portanto conhecidos.

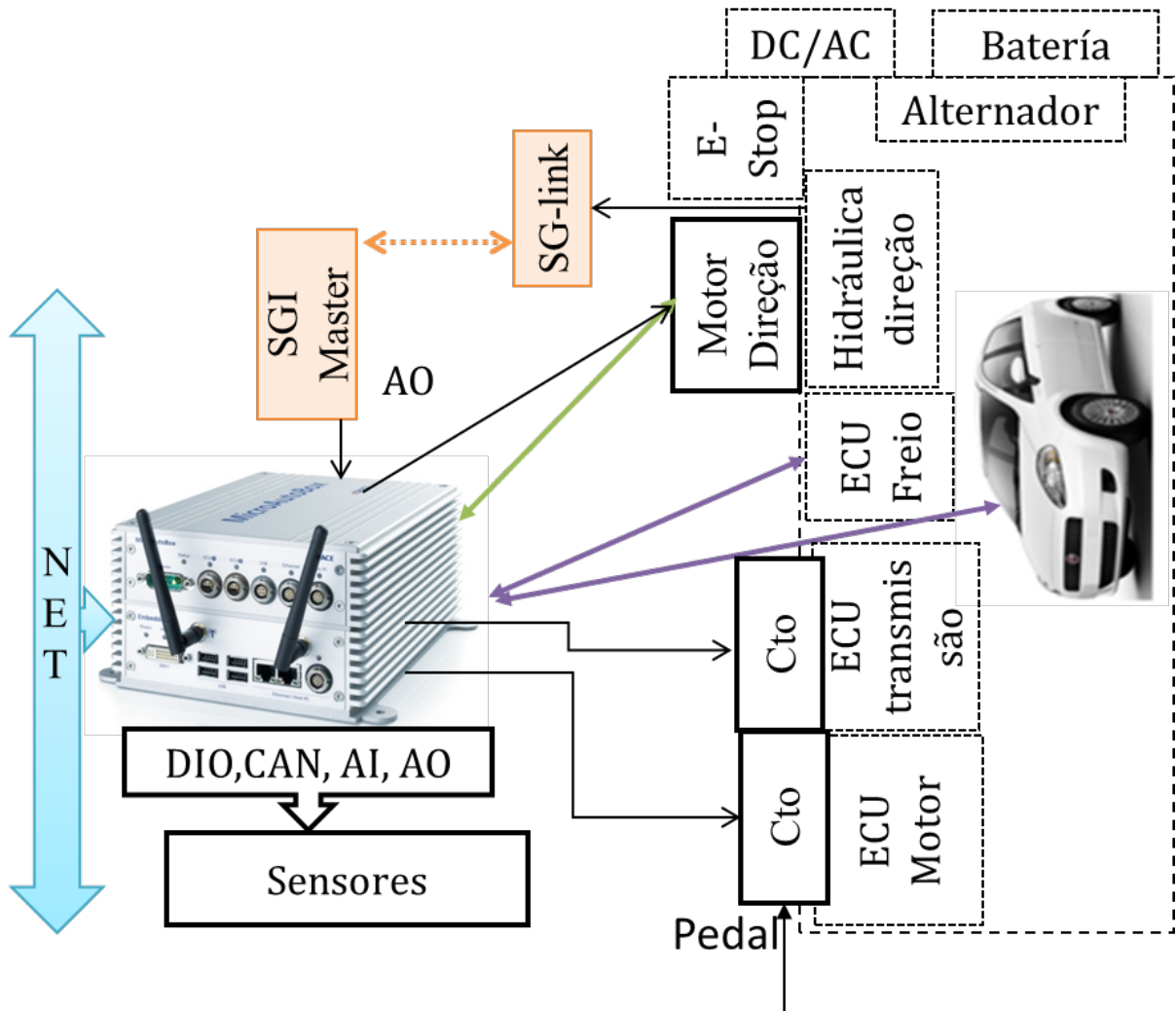


Figura 2.3: Arquitetura do sistema de automação. Comunicação: Ethernet (azul), RS232 (verde), RF (laranja), elétricas (preto) e CAN (Roxa)

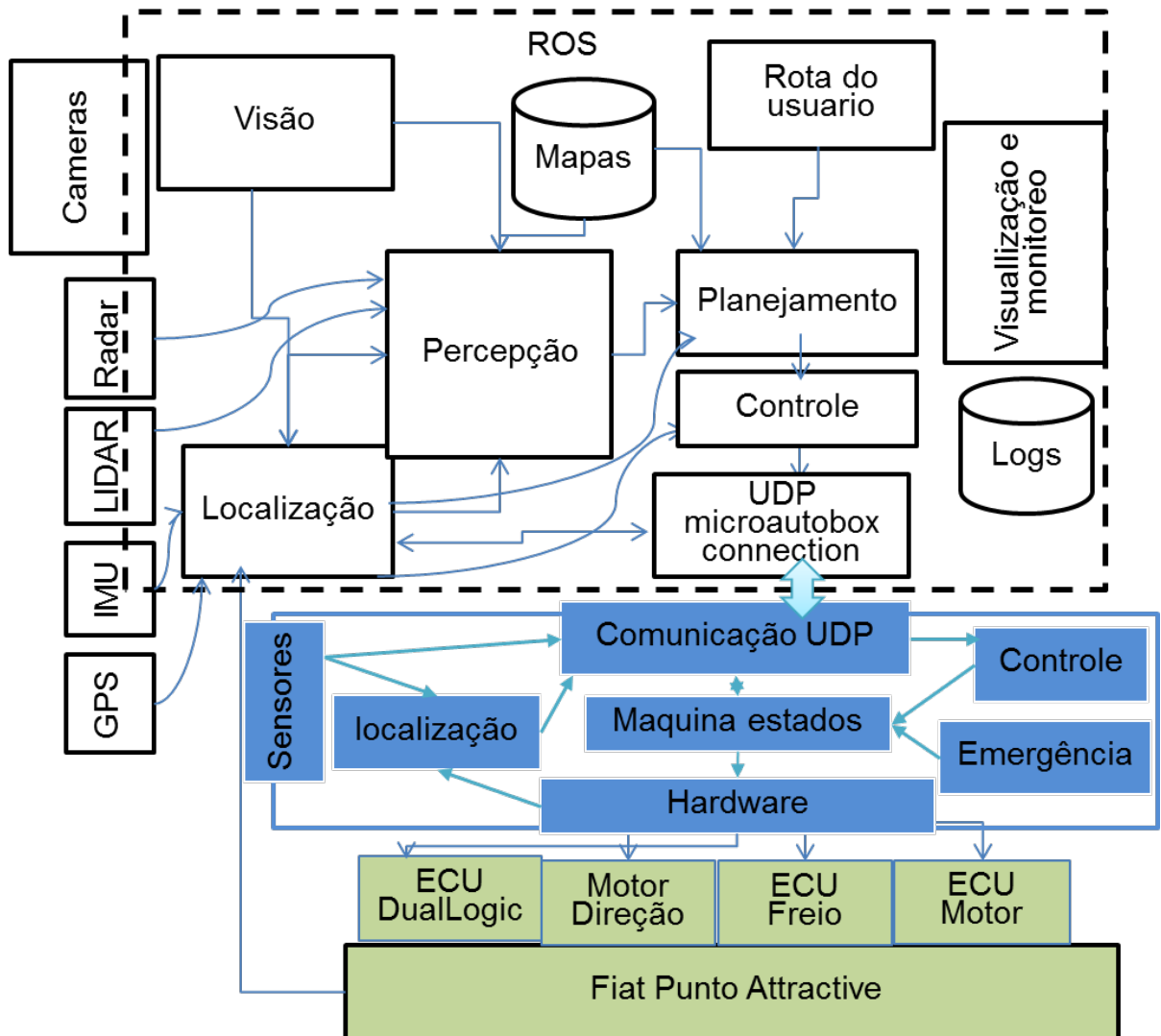


Figura 2.4: Arquitetura do software para o veículo autônomo. Processamento em Linux (Preto), interação com o usuário (Cinza), firmware (Verde), embarcado (Azul).

- **Software embarcado (azul).** Software implementado no sistema embarcado projetado para tarefas de tempo real encarregado de interagir com o hardware e o firmware. Os programas projetados nesta arquitetura são máquinas de estados da iteração do hardware de automação com o software, uma rotina de emergência, rotinas da camada de controle que tenham exigências de tempo real, rotina de localização e interação com sensores. Note-se que adicionalmente existe um programa que permite a comunicação com o software em Linux. Uma descrição detalhada do software implementado pode ser encontrado no apêndice B.
- **Software no Linux (preto).** Software implementados em arquiteturas de tipo PC com a possibilidade de serem distribuídas em diferentes computadores. O primeiro requisito para projetar a arquitetura foi definir o mediador (*middleware*) ou meta-sistema operacional onde vão ser executados todos os programas da arquitetura tipo PC. Para isso foram estudadas ferramentas para fazer a transferência de informação entre os módulos do sistema que fosse robusta, distribuída, fácil de utilizar e capaz de intercambiar informações de forma síncrona e assíncrona. Neste ponto as seguintes ferramentas foram analisadas:
 - **KogMo-RTDB.** Banco de dados centralizado *Real-Time Database* for Cognitive Automobiles (GOEBL E FÄRBER, 2007) desenvolvido por Matthias Goebel e Georg Farber patrocinados pelo projeto *Transregional Collaborative Research Centre 28 ‘Cognitive Automobiles’* e desenvolvido na universidade técnica de Munich. Sua principal vantagem é que foi projetado para veículos autônomos, com a particularidade de que os dados estão centralizados e são dirigidos por processos com *Hard Real Time* usando Xenomai. KogMo-RTDB tem sido usada pelo veículo AnnieWay (GEIGER *et al.*, 2012) que ganhou o Grand Cooperative Driving Challenge em 2011. Ele foi desenvolvido como software livre primeiro com contrato de licença LGPL e durante o projeto foi trocado para o contrato de licença Apache. Ele foi descartado porque a partir de 2012 o desenvolvimento tem sido baixo, eliminando qualquer possibilidade de suporte, adicionalmente ele tem sido testado em uma só arquitetura de hardware o que eliminava a flexibilidade.
 - **Microsoft RDS.** O Microsoft Robotics Developer Studio é baseado no CCR (Concorrência e Coordenação Runtime) de .NET em Windows para o gerenciamento de tarefas paralelas assíncronas. Esta técnica envolve o uso de transmissão de mensagens, um tempo de execução leve orientada a serviços, um DSS (Decentralized Software Services) e o gerenciamento de serviços múltiplos para alcançar comportamentos complexos. As características incluem: a ferramenta de programação visual, Microsoft Visual Programming Language para criar e depurar aplicações do robô, interfaces web-based baseados no Windows, simulação 3D (incluindo aceleração de

hardware), fácil acesso a sensores e atuadores de um robô, entre outras. A linguagem de programação principal é C#. Ele é livre para ser usado mais não pode ser modificado e o suporte da comunidade acadêmica é pouco e requer esquemas de licenciamento difíceis.

- **ROS.** O Robotic Operating System (QUIGLEY *et al.*, 2009) atua como um meta sistema operativo e uma caixa de ferramentas (*toolbox*). ROS fornece várias bibliotecas que são comumente usadas em problemas robóticos. Tem interfaces de comunicação entre programas (definidos como nós) como o modelo publisher-subscriber, cliente-servidor, cliente-servidor assíncrono, filas de dados, relógios e gerenciador de tarefas cíclicas. Além disso, também fornece múltiplas interfaces gráficas com ferramentas para ajudar o desenvolvimento. As ferramentas podem ser usadas no tempo de execução ou com dados armazenados para processamentos posteriores. Esta ferramenta é livre com contrato de licença Apache.

ROS foi o mediador selecionado para fazer os caminhos entre os programas principalmente por sua rápida curva de aprendizagem, o suporte da comunidade e dos desenvolvedores, a possibilidade de usar ferramentas e módulos já desenvolvidos e a possibilidade de ser integrado com ferramentas de programação para desenvolver software com requerimentos de tempo real como o OROCOS (BRUYNINCKX, 2001).

2.3 Arquitetura de hardware e software no ROS

Definido o ROS como *Middleware* do software no Linux foram implementados e testados os programas para ligar a instrumentação existente do veículo e o sistema embarcado do VILMA01. Na figura 2.5 são apresentados os nós utilizados no VILMA01. Os nós desenvolvidos neste trabalho são os que estão nos pacotes *vilma_ma_ros* e *vilma_perception*. O desenvolvimento e as funcionalidades dos programas do pacote *vilma_ma_ros* são apresentados no apêndice C e do segundo pacote *vilma_perception* são apresentados na seção 6.2.

Finalmente, dentro da arquitetura de software do PC foi integrada uma ferramenta de simulação do veículo para que todos os softwares possam ser testados sem a necessidade direta do sistema embarcado e do hardware do sistema. Foram testados os simuladores Gazebo(KOENIG E HOWARD, 2004), V-Rep(FREESE *et al.*, 2010) e MORSE (ECHEVERRIA *et al.*, 2011). Durante uma primeira etapa do projeto foi utilizado o primeiro software Gazebo no qual foi gerado um modelo virtual do veículo com sensores como IMU, Câmeras, GPS, LIDAR, velocidade das rodas e foram realizados dois trabalhos de mestrado com ele. No ano 2014 com a finalidade de

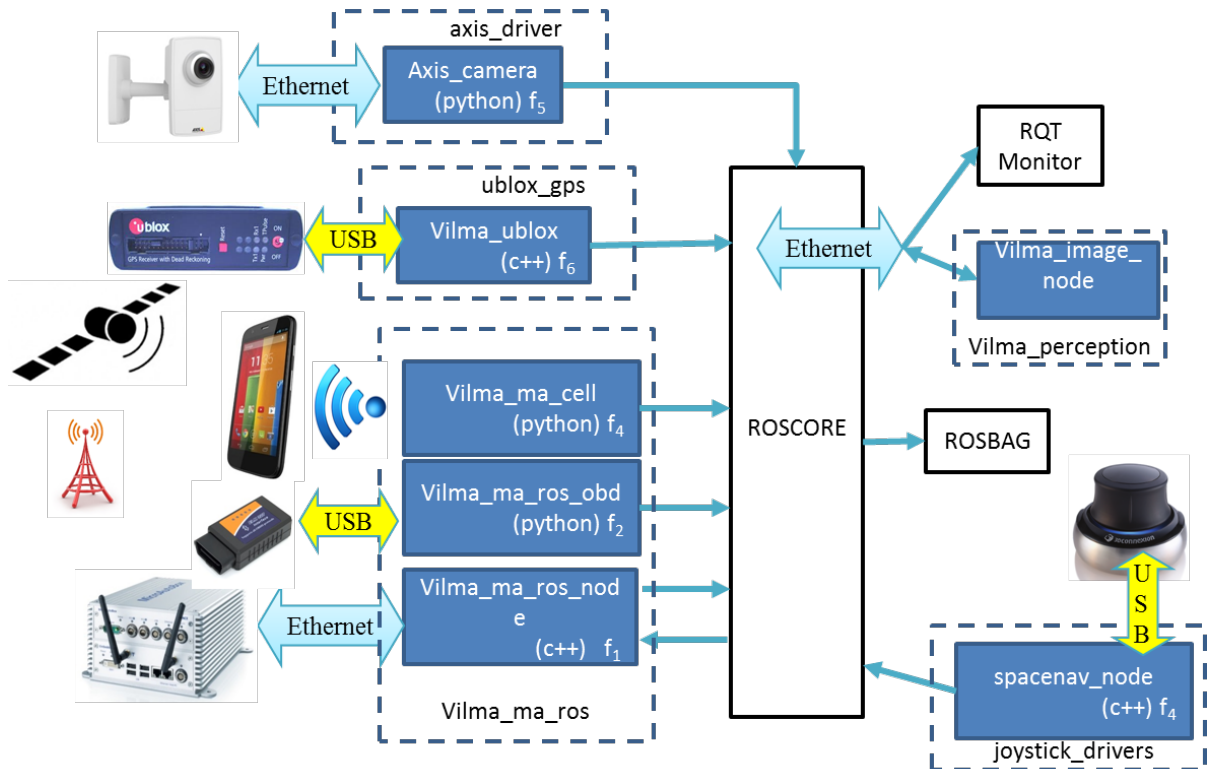


Figura 2.5: Hardware conectado a VILMA01 através de software no ROS.

buscar uma parceria com o grupo do veículo autônomo CARINA (FERNANDES *et al.*, 2014) o simulador foi trocado para o Morse, e um artigo sobre o uso de VILMA01 com este simulador foi submetido no Simpósio Brasileiro de Automação Inteligente (SBAI) (FERREIRA *et al.*, 2015). Estes simuladores tem servido de suporte em outros projetos, mas para o desenvolvimento desta proposta foi utilizado como software de simulação Matlab/Simulink por sua compatibilidade com o sistema embarcado e dado que a dinâmica veicular de estes simuladores é aproximada e não permitia validar as estratégias baseadas no modelo dinâmico do veículo.

2.4 Considerações finais

O presente capítulo permitiu definir a arquitetura do veículo autônomo e projetar o software e hardware requerido para automatizar um veículo comercial. Adicionalmente, neste capítulo foi definido que para propor estratégias de cooperação entre o condutor e sistema autônomo com análises de risco é necessário estudar as camadas de controle de movimento e de planejamento. O estudo destas duas camadas e as estratégias utilizadas em cada uma delas são apresentadas nos capítulos a seguir.

3 Controle de trajetórias

Este capítulo apresenta o controle de trajetórias que consiste em manipular os graus de liberdade do veículo, como a direção, o freio e a aceleração, para levar o veículo a uma posição $[x(t), y(t), \theta(t)]$ desejada para cada instante de tempo t . A metodologia escolhida para o projeto consiste em dividir o problema em dois controladores, o primeiro encarregado de fazer o carro seguir um perfil de velocidade e o segundo encarregado de manipular a direção do veículo para obter a orientação desejada e eliminação dos erros laterais.

O controle encarregado de seguir o perfil de velocidades não será apresentado neste trabalho dado que está sendo realizado por outros integrantes do grupo de pesquisa. Portanto, será apresentada somente a proposta de controle de trajetórias assumindo que a velocidade longitudinal do veículo é conhecida focando o trabalho no controle de orientação e erro lateral através da direção do veículo. Para cumprir os dois objetivos com um só grau de liberdade, na bibliografia são usadas técnicas de controle realimentada baseadas em modelos do veículo (SNIDER, 2009)(KEEN E COLE, 2006)(KEEN, 2008)(GOMES, 2006)(RAFFO *et al.*, 2009b). A técnica de controle utilizada neste trabalho é a de controle preditivo baseado em modelo (MACIEJOWSKI, 2002)(MACIEJOWSKI, 2002), MPC (*Model Predictive Control*) em inglês, no caso de modelos lineares e sem restrições no processo de otimização. Como arquitetura de controle, o controle em cascata para controlar primeiro a direção e sobre ele controlar a trajetória é a arquitetura mais encontrada na bibliografia (RAFFO *et al.*, 2009b), mas neste trabalho com a finalidade de levar em conta também o condutor do veículo e estudar as implicações de um controle cooperativo de trajetórias, a malha de controle de direção do veículo vai fazer parte de um só controlador multivariável.

3.1 Bibliografia

Na bibliografia estudada nesta área o controle de trajetórias é dividido em duas camadas ou duas malhas de controle retroalimentado como é mostrado na figura 3.1. A primeira malha fechada consiste em controlar o sistema de direção auxiliando ao condutor seja em torque (*kinetic assist*) ou em movimento (*kinematic assist*), conhecido como *Active front control*, que é o método pelo qual o controle de trajetórias atua. A segunda malha fechada consiste em fazer o controle de trajetórias de acordo com um modelo dinâmico do veículo e procurando seguir uma trajetória desejada.

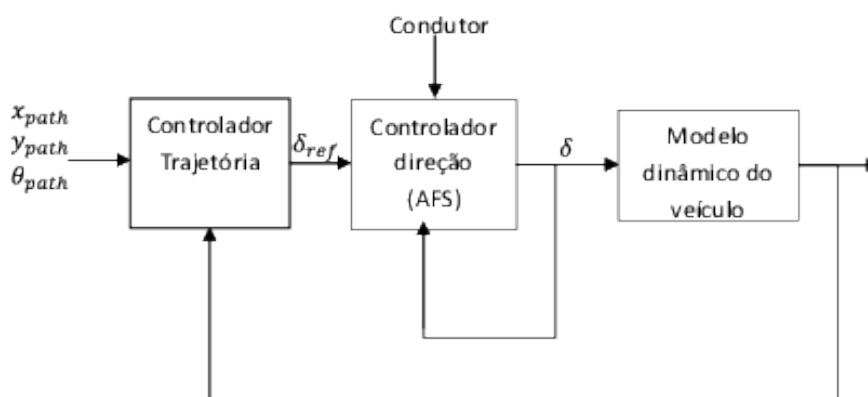


Figura 3.1: Controle de trajetória em duas camadas

3.1.1 Controle de direção do veículo

O controle da direção nos veículos depende dos sistemas de assistência ao condutor integrado no sistema de direção e podem ser classificados em três tipos: assistidos hidráulicamente, assistidos eletricamente e *steer-by-wire*. As primeiras procuram ajudar o usuário no torque requerido (assistência cinética) para executar qualquer movimento do veículo, por uma função não-linear em relação ao torque que o usuário gera, adicionando aproximadamente perto do 80% do torque exigido através do sistema hidráulico ou elétrico (YIH, 2005) (WONG, 2001). A seguir são apresentados brevemente descrições de trabalhos nesta área que tem servido como base para o desenvolvimento da proposta.

Sistemas de assistência elétrica

Os sistemas de assistência de potência elétrica, conhecidos como EPAS (*Electric Power Assisted Steering*), tem vantagens sobre os de assistência hidráulica. Uma vantagem está relacionada à economia em combustível dado que no sistema hidráulico a bomba está sempre ligada e no sistema elétrico o motor só é ligado quando requerido. Outra vantagem está relacionada às curvas de torque que no caso do sistema hidráulico são fixas e definidas pelo hardware e no sistema elétrico estas curvas são geradas por software, permitindo que as curvas de assistência dependam não só de variáveis como velocidades do veículo, mas também dos desejos do usuário entre outros muitos parâmetros. Um exemplo deste tipo de curvas é apresentado na figura 3.2 onde pode-se ver a curva do torque de assistência ao condutor dependente da velocidade (JONES E WILLIAMS, 2001).

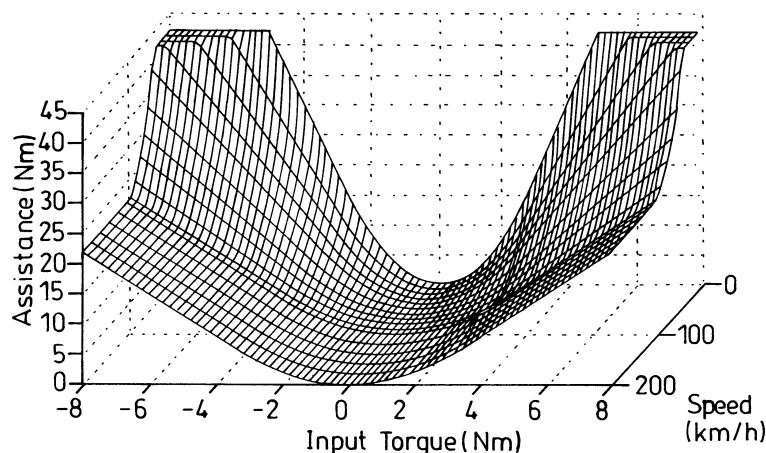


Figura 3.2: Curvas de torque de assistência ao condutor do sistemas EPAS dependentes da velocidade . fonte (JONES E WILLIAMS, 2001)

A maioria dos gráficos de assistência de torque (no inglês *Boost Curves*) foram patenteadas durante a década anterior, e uma amostra do projeto delas é apresentada por (CHEN *et al.*, 2011) onde um modelo para o sistema foi estabelecido, formado pela dinâmica do veículo, do sistema mecânico da direção e do sistema elétrico. Com base neste modelo, uma curva polinomial foi concebida e avaliada com o simulador ADAS para melhorar o desempenho do sistema. Os resultados mostraram uma redução do tempo de reação e uma diminuição do amortecimento garantindo estabilidade.

Uma segunda geração de EPAS, podem ser os chamados *Active Front Steering*(AFS), onde a assistência além de ser cinética (de torque) também tem assistência cinemática (de superposição de ângulo). (KLIER *et al.*, 2004) apresenta o modelo desenvolvido pela ZF em parceria com BMW, onde mostra por meio de módulos as funcionalidades e o processo de integração do sistema. Entre as funcionalidades possíveis têm-se funções cinemáticas como ajuda na troca de faixa ou para melhora na estabilização do veículo como controle da orientação ou rejeição às perturbações. Como entradas ao sistema têm-se as velocidades das rodas, a velocidade do veículo e o estado do sistema de freios.

(AVAK, 2004) apresenta o modelo de um sistema de direção AFS, em que um ângulo é sobreposto ao movimento do utilizador gerando uma desmultiplicação da variável que depende da velocidade linear do veículo. Estes sistemas mantêm a ligação mecânica entre o volante e as rodas, mas instala um motor elétrico na coluna de direção e a maior parte das vantagens de um sistema *steer-by-wire* pode ser obtido utilizando um sistema de superposição sem os problemas técnicos e legais relativas a segurança. Ele apresenta controladores analógicos e digitais usando

feedback e *feedforward*, que são projetados com modelo *single-track* de um carro e modelo mecânico e elétrico na direção.

Steer-by-wire

(YIH E GERDES; YIH *et al.*; YIH, 2005; 2004; 2005) apresenta uma estratégia *steer-by-wire* de controle que enfatiza as vantagens de *feedforward* no volante quando se lida com distúrbios conhecidos que são o torque de auto alinhamento e a dinâmica do mecanismo da direção. O controlador é implementado em um veículo com uma direção assistida hidráulica que foi convertido para *steer-by-wire*. O modelo matemático da direção estudado inclui o sistema hidráulico baseado em (PROCA E KEYHANI, 1998) o qual é complementado com o modelo dinâmico de bicicleta do veículo para o análise do sistema, mas dado a difícil obtenção de parâmetros do modelo da direção, um modelo de caixa preta tipo ARMAX é utilizado para projetar os controladores.

Um dos benefícios mais atraentes de *steer-by-wire* é a capacidade da direção ativa, pois através do estado do veículo pode-se manipular as forças no volante para se obter curvas de assistência que dependem da preferência do motorista e das condições de funcionamento. Alcançar, uma mudança contínua e suave na manipulação de qualidade requer tanto estimação de estado precisa e um controle preciso da interface háptica (o volante).

Outro benefício da *steer-by-wire* é a disponibilidade de informações do torque do atuador da direção pela corrente do mesmo, porque ele indiretamente leva a uma determinação dos estados do veículo, através do torque de auto-alinhamento dos pneus, o que está relacionado com as forças dos pneus.

(LEI-YAN *et al.*; LEI-YAN *et al.*, 2009; 2008) apresenta um modelo de malha fechada entre o condutor do veículo e a estrada de um sistema de direção *steer-by-wire* simulado em Matlab/Simulink. Os parâmetros do modelo são otimizados por algoritmos genéticos. Estratégias de controle do *steer-by-wire* de direção ativa, que incluem o *feedforward* e *feedback*, assim como o controle háptico para o volante receber as força da estrada. Diferentes relações de força do volante versus força da direção são otimizados através de algoritmos genéticos para otimizar as estratégias *feedforward*. O controle retroalimentado é feito através do sensor de velocidade de guinada o que melhora a estabilidade e a largura de banda. O torque transmitido ao volante está relacionado ao torque de alinhamento das rodas e da velocidade do carro emulando um sistema EPAS.

(BAJCINCA *et al.*, 2006) propõem uma configuração *steer-by-wire* composta de um controlador e um algoritmo para a estimativa das condições da estrada. As forças de atrito são modelados como um processo estocástico Gauss-Markov e portanto um filtro de Kalman estendido é projetado para a sua estimativa. Este sinal é utilizado como sensor para o controlador do *steer-by-wire*. Simulações com um modelo não-linear dinâmico do veículo são feitas para validar o abordagem.

Em (CETIN *et al.*; CETIN *et al.*, 2012; 2009) uma bancada foi construída para estudar e implementar estratégias de controle de impedância em um sistema *steer-by-wire*. Técnicas de estimativas adaptativas online são usadas para identificar os parâmetros dinâmicos de sistema de direção do veículo e da interface háptica do volante. Um modelo de veículo não-linear de 4 DOF, incluindo as velocidades longitudinal, lateral, e de guinada são obtidas usando a mecânica newtoniana para simular e testar as estratégias de controle de impedância desenvolvidos. Os resultados mostram melhoras na estabilidade do carro e na força que é enviada ao condutor.

Em (MAMMAR E KOENIG, 2002) o sistema de controle é desenvolvido usando uma combinação de controladores de *feedforward* e de *feedback*. No controlador de malha fechada o sensor de velocidade de guinada é usado e o sistema *feedforward* se baseia em uma função do comando de direção do condutor. Esta estratégia permite rejeição das forças laterais e de distúrbios de torque. A análise da estabilidade do modelo não-linear do carro mostra que, em uma estrada de baixo atrito, grandes ângulos de direção podem levar à rotação do veículo e redução da região de estabilidade, também pode-se encontrar uma redução do domínio de estabilidade, devido ao ângulo de direção do condutor de entrada. A redução torna-se mais pronunciada quando o ângulo de direção aumenta. O controlador implementado, do tipo H_∞ , é testado em simulação em três condições diferentes.

(MEHDIZADEH *et al.*, 2011) propõem que os sistemas de rastreamento de faixa (em inglês *lanekeeping*) trabalhem junto com o motorista, por um sistema de variação da relação virtual entre o volante e o ângulo da roda, procurando que o usuário ainda tenha a realimentação do torque de alinhamento das rodas como um sistema convencional. Baseado no princípio de veículo virtual, uma nova força de retroalimentação é projetada para excluir os efeitos do sistema de rastreamento de faixa. Uma das novidades do artigo é que ao invés de permitir que outros métodos possam desestabilizar o veículo, o novo método separa os sistemas de rodas e de direção do veículo e resolve com isso, qualquer problema de estabilidade.

3.1.2 Controle de trajetórias

(FALCONE *et al.*, 2007) apresentam o controle preditivo para o controle lateral e de orientação do veículo frente a uma troca dupla de faixa (descrito por uma equação matemática) em estradas com neve. Para o controle preditivo utiliza-se um modelo baseado no modelo dinâmico de bicicleta e um modelo de Pacejka das forças das rodas. Com os modelos são projetados dois tipos de controladores, o primeiro baseado no modelo não linear e o segundo no modelo linearizado, nos dois casos com restrições suaves na resolução do problema de otimização quadrática. O controle tem como saídas o eixo lateral, a orientação e, no caso do controle linearizado, a velocidade lateral do veículo. O controle não linear é testado em simulação e o controle linearizado é implementado em um sistema embarcado DSPACE usando como fatores de desempenho os erros máximos e RMS (*root mean square*) de orientação e lateral. Nos resultados foi possível controlar o veículo ate velocidades de $21m/s$ em condições de neve pela facilidade da técnica de prever o comportamento não linear das rodas.

(SNIDER, 2009) apresenta soluções para o problema de controle de trajetória para o qual usa: algoritmos de controle baseado nas relações geométricas do veículo como *Pure Pursuit* ou o método de Stanley; algoritmos baseados no modelo cinemático de bicicleta de controle retroalimentado não linear (*smooth feedback stabilization method*); e algoritmos baseados no modelo dinâmico de bicicleta com técnicas de controle ótimo *LQR* (*Linear Quadratic Regulator*) com *feedforward*. O artigo conclui que para incluir novas aplicações o próximo passo seria utilizar técnicas de controle preditivo.

(SHIM *et al.*, 2012) apresentam um algoritmo de planejamento de trajetórias e um controle preditivo de trajetórias através da direção e o torque dos pneus. Para o planejamento as posições x e y são parametrizadas com respeito ao tempo de forma polinomial de ordem seis para evitar obstáculos. Os parâmetros são encontrados levando-se em conta que o modelo cinemático do veículo tem um problema de valor limite que é resolvido analiticamente. Para o controle preditivo utiliza-se um modelo baseado na dinâmica do veículo de oito graus de liberdade e um modelo linear das forças dos pneus, e como entrada o torque dos pneus e a direção do veículo e como saídas de controle as posições x e y . O artigo apresenta resultados em simulações com diferentes cenários a velocidades próximas a $10m/s$.

Os artigos (RAFFO *et al.*; RAFFO *et al.*; GOMES *et al.*, 2009a; 2009b; 2006) apresentam controle de trajetórias utilizando a técnica de controle preditivo baseado em modelo e um planejamento baseado em primitivas, sendo destinado ao controle do comportamento cinemático e também do comportamento dinâmico do veículo. O controle do comportamento cinemático é

baseado no modelo linearizado da bicicleta e o controle dinâmico é baseado modelo dinâmico linear da bicicleta. O sistema utiliza o controle cinemático a velocidades menores de $9.4m/s$ e a velocidades maiores adiciona em cascada o controle dinâmico. O projeto apresenta a implementação do sistema em um carro de mini-baja com o controle em tempo real desenvolvido usando programação orientada a objetos em um DSP e um PowerPC. Provas de implementação do controle preditivo não linear baseado em um sistema linear geram tempos de resolução 40 vezes maiores e melhoras nos erros de 2% nas condições de trabalho do veículo.

(GRAY *et al.*, 2012) apresentam o planejamento e o controle de trajetórias de um veículo autônomo usando primitivas e controle preditivo respectivamente. No planejamento são usadas dois tipos de primitivas, a primeira chamada *trim* são trajetos baseados em estados de equilíbrio como giro a direita, linha reta, giro a esquerda ou deslizamento no gelo e a segunda chamada *maneuvers* são as que fazem as transições entre os estados discretos ou estados não estáveis (por exemplo para desviar de obstáculos) como troca de faixa ou troca de faixa dupla. Estas primitivas são encontradas a partir de uma otimização baseada no modelo não linear do veículo. Para o controle preditivo utiliza-se um modelo baseado na dinâmica de quatro rodas do veículo e um modelo de Pacejka das forças das rodas, e como entrada o torque das rodas e a direção do veículo, e como saídas de controle \dot{x} , y e o angulo de orientação (ψ) assim como $\dot{\psi}$. O artigo apresenta resultados em simulação e testes em um veículo real implementados com um sistema embarcadoDSPACE, um *GPS* diferencial e uma IMU.

3.2 Controle Preditivo baseado em modelo

O controle preditivo baseado em modelo é uma técnica que permite controlar uma planta em que sua dinâmica ou modelo matemático discreto é conhecido. O controle é possível com a retroalimentação dos valores presentes das variáveis de controle, $x(k)$, e dos valores anteriores do sinal do controle, $u(k-1)$. A técnica consiste em resolver um problema de otimização usando a função objetivo, $V(k)$, descrita geralmente por (3.1), que representa a norma quadrática da diferença entre a variável controlada e o valor de referência, acrescido da variação quadrática do sinal de controle de um horizonte de previsão, H_p .

$$V(k) = \sum_{i=H_w}^{H_p} \|\hat{\mathbf{y}}(k+i|k) - \mathbf{r}(k+i)\|_{Q(i)}^2 + \sum_{i=0}^{H_c-1} \|\Delta \mathbf{u}(k+i|k)\|_{R(i)}^2 \quad (3.1)$$

na qual,

- $\mathbf{r}(k + i)$, é o sinal de referência das saídas a controlar, conhecidas em um horizonte de predição futuro determinado por H_p amostras.
- $\mathbf{u}(k + i|k)$, são os esforços de controle a estimar $\forall i = 0 \dots H_c$.
- $\hat{\mathbf{y}}(k + i|k)$, são as saídas do sistema a controlar previstas no horizonte de predição H_p .
- Q é uma matriz de pesos para ponderar a importância da saída controlada na otimização.
- R é uma matriz de pesos para ponderar a importância da variação do esforço de controle na otimização.
- H_w é o retardo computacional, causado por exemplo por atrasos de comunicações.

Esta função objetivo no caso sem restrições tem solução analítica em um horizonte de controle, H_c . A seguir, baseado em (MACIEJOWSKI, 2002), é primeiro apresentado como prever as saídas de controle, $\mathbf{y}(k + i|k) \forall i = H_w \dots H_p$, baseadas em uma representação de variáveis de estado, e depois é apresentado o processo de otimização para minimizar a equação (3.1) e obter os valores de $\mathbf{u}(k + i), \forall i = 0 \dots H_c$.

3.2.1 Predição das Saídas de Controle

A fim de resolver o problema de otimização do controle é necessário prever os valores futuros das saídas a controlar, $\hat{\mathbf{y}}(k + i|k)$, a partir da estimação do estado atual, $\hat{\mathbf{x}}(k|k)$, com respeito as entradas futuras $\mathbf{u}(k + i|k)$. A previsão pode ser um problema muito difícil se as perturbações que atuam no sistema são desconhecidas, de maneira que primeiro é pressuposto que as perturbações não existem ou que são conhecidas (representadas por um vector de entrada $\mathbf{u}_2(k)$) e que todas as variáveis de estado podem ser medidas no instante k ou que podem ser estimadas por um observador de estado. Assim tudo o que tem que ser feito para prever as saídas é feito através de um processo iterativo com o modelo discreto. Portanto para prever o valor futuro das variáveis de estado no caso de $i = 1$ temos:

$$\begin{aligned} \hat{\mathbf{x}}(k + 1|k) &= A\hat{\mathbf{x}}(k) + B\hat{\mathbf{u}}_1(k|k) + B\mathbf{u}_2(k) \\ &= A\hat{\mathbf{x}}(k) + B\mathbf{u}_1(k - 1) + B\Delta\hat{\mathbf{u}}_1(k|k) + B\mathbf{u}_2(k) \end{aligned} \quad (3.2)$$

no caso de $i = 2$, temos

$$\begin{aligned}
\hat{\mathbf{x}}(k+2|k) &= A\hat{\mathbf{x}}(k+1) + B\hat{\mathbf{u}}_1(k+1|k) + B\mathbf{u}_2(k+1) \\
&= A\hat{\mathbf{x}}(k+1) + B\hat{\mathbf{u}}_1(k|k) + B\Delta\hat{\mathbf{u}}_1(k+1|k) + B\hat{\mathbf{u}}_2(k+1) \\
&= A(A\hat{\mathbf{x}}(k) + B\hat{\mathbf{u}}_1(k-1) + B\Delta\hat{\mathbf{u}}_1(k|k) + B\hat{\mathbf{u}}_2(k)) \\
&\quad + B\hat{\mathbf{u}}_1(k|k) + B\Delta\hat{\mathbf{u}}_1(k+1|k) + B\hat{\mathbf{u}}_2(k+1)
\end{aligned} \tag{3.3}$$

Generalizando $\forall i = 0 \dots H_p$ chega-se à equação matricial (3.4) onde pode ser visto que depende dos esforços de controle passados, dos valores das variáveis de estado atuais, das entradas de perturbação do sistema e da variação dos sinais de controle a estimar.

$$\begin{aligned}
\begin{bmatrix} \hat{x}(k+1|k) \\ \vdots \\ \hat{x}(k+H_c|k) \\ \hat{x}(k+H_c+1|k) \\ \vdots \\ \hat{x}(k+H_p|k) \end{bmatrix} &= \begin{bmatrix} A \\ \vdots \\ A^{H_c} \\ A^{H_c+1} \\ \vdots \\ A^{H_p} \end{bmatrix} \hat{\mathbf{x}}(k) + \begin{bmatrix} B \\ \vdots \\ \sum_{i=0}^{H_c-1} A^i B \\ \sum_{i=0}^{H_c} A^i B \\ \vdots \\ \sum_{i=0}^{H_p} A^i B \end{bmatrix} \mathbf{u}_1(k-1) \\
&+ \begin{bmatrix} B & 0 & \dots & 0 \\ AB & B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{H_c} B & A^{H_c-1} B & \dots & 0 \\ A^{H_c+1} B & A^{H_c} B & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ A^{H_p} B & A^{H_p-1} B & \dots & B \end{bmatrix} \begin{bmatrix} \mathbf{u}_2(k) \\ \vdots \\ \mathbf{u}_2(k+H_p) \end{bmatrix} \\
&+ \begin{bmatrix} B & \dots & 0 \\ AB+B & \dots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{H_c-1} A^i B & \dots & B \\ \sum_{i=0}^{H_c} A^i B & \dots & AB+B \\ \vdots & \vdots & \vdots \\ \sum_{i=0}^{H_p-1} A^i B & \dots & \sum_{i=0}^{H_p-H_c} A^i B \end{bmatrix} \begin{bmatrix} \Delta\hat{\mathbf{u}}_1(k|k) \\ \vdots \\ \Delta\hat{\mathbf{u}}_1(k+H_c-1|k) \end{bmatrix}
\end{aligned} \tag{3.4}$$

A predição das saídas controladas $\hat{z}(k+1|k)$ assumindo que só dependem das variáveis de estado podem ser calculadas pela equação (3.5), a qual na forma matricial é descrita pela equação (3.6).

$$\begin{bmatrix} \hat{\mathbf{y}}(k+1|k) \\ \vdots \\ \mathbf{y}(k+H_p|k) \end{bmatrix} = \begin{bmatrix} C_z & 0 & 0 & 0 \\ 0 & C_z & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & C_z \end{bmatrix} \begin{bmatrix} \hat{x}(k+1|k) \\ \vdots \\ \hat{x}(k+H_p|k) \end{bmatrix} \quad (3.5)$$

$$\begin{aligned} \mathcal{Y}(k) &= \begin{bmatrix} \hat{\mathbf{y}}(k+1|k) \\ \vdots \\ \hat{\mathbf{y}}(k+H_p|k) \end{bmatrix} \\ &= \Psi \mathbf{x}(k) + \Upsilon \mathbf{u}(k-1) + \Theta \Delta \mathcal{U}(k) + \Phi \mathcal{U}_2(k) \end{aligned} \quad (3.6)$$

na qual,

- Υ é uma matriz de $m(H_p - H_w + 1)xl_1$
- Θ é uma matriz de $m(H_p - H_w + 1)xl_1H_c$
- Ψ é uma matriz de $m(H_p - H_w + 1)xn$
- Φ é uma matriz de $m(H_p - H_w + 1)xl_2H_p$
- n é o número de estados, l_i é o número de entradas u_i , m é o número de saídas controladas

3.2.2 Otimização analítica no caso sem restrições

A função objetivo de (3.1) na forma matricial é descrita por:

$$V(k) = \| \mathcal{Y}(k+i|k) - \mathcal{T}(k) \|_{Q(i)}^2 + \| \Delta \mathcal{U}(k) \|_{R(i)}^2 \quad (3.7)$$

na qual,

$$\mathcal{T}(k) = \begin{bmatrix} \mathbf{r}(k + H_w | k) \\ \vdots \\ \mathbf{r}(k + H_p | k) \end{bmatrix} \quad (3.8)$$

e as matrizes \mathcal{Q} e \mathcal{R} são dadas por:

$$\mathcal{Q} = \begin{bmatrix} Q(h_w) & 0 & \cdots & 0 \\ 0 & Q(h_w + 1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & Q(h_p) \end{bmatrix} \quad (3.9)$$

$$\mathcal{R} = \begin{bmatrix} R(0) & 0 & \cdots & 0 \\ 0 & R(1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & R(h_c - 1) \end{bmatrix} \quad (3.10)$$

A partir da equação (3.6) pode ser definido $\mathcal{E}(k)$ como

$$\mathcal{E}(k) = T(k) - \Psi x(k) - \Upsilon u(k-1) - \Phi u_2(k) \quad (3.11)$$

portanto a equação (3.7)

$$V(k) = \|\Theta \Delta \mathcal{U}(k) - \mathcal{E}(k)\|_{\mathcal{Q}(i)}^2 + \|\Delta \mathcal{U}(k)\|_{\mathcal{R}(i)}^2$$

Resolvendo a norma quadrática temos

$$\begin{aligned} V(k) &= (\Theta \Delta \mathcal{U}(k) - \mathcal{E}(k)) \mathcal{Q} (\Theta \Delta \mathcal{U}(k) - \mathcal{E}(k)) + \Delta \mathcal{U}(k)^T \mathcal{R} \Delta \mathcal{U}(k) \\ &= \mathcal{E}(k)^T \mathcal{Q} \mathcal{E}(k) - \Delta \mathcal{U}(k) \mathcal{G} + \Delta \mathcal{U}(k)^T \mathcal{H} \Delta \mathcal{U}(k) \end{aligned} \quad (3.12)$$

na qual $\mathcal{G} = 2\Theta \mathcal{Q} \mathcal{E}(k)$ e $\mathcal{H} = \Theta^T \mathcal{Q} \Theta + \mathcal{R}$.

Para encontrar o valor ótimo de $\Delta \mathcal{U}(k)$ encontra-se o gradiente do $V(k)$ e iguala-se a zero. A partir da equação (3.12) tem-se que

3.2.3 Modelo dinâmico linear do veículo

Como foi explicado na seção 3.2, a técnica requer um modelo de estados linear e discretizado para funcionar. Portanto, a partir das equações do apêndice G, o sistema de equações das variáveis de estado $\mathbf{x}_c = [\mathbf{x}_s : \mathbf{x}_e]^T$ pode ser escrito pela equação (3.17) dado que τ_{fa} na equação (G.33) pode ser substituída em termos de \mathbf{x}_e através das equações (G.24), (G.20) e (G.37).

$$\dot{\mathbf{x}}_c = A_c \mathbf{x}_c + B_1 \mathbf{u}_1 + B_2 \mathbf{u}_2 \quad (3.17)$$

na qual, $\mathbf{u}_1 = \begin{bmatrix} v_m & \tau_{user} \end{bmatrix}^T$ são os esforços de controle e $\mathbf{u}_2 = \begin{bmatrix} r_p(s) & \dot{r}_p(s) \end{bmatrix}^T$ são as perturbações conhecidas do sistema de controle (*Feedforward Control*). Note-se que o torque do condutor é considerada como uma entrada de controle na qual é possível prever algumas características do sinal, como a frequência em que o usuário vai conseguir responder e quanta energia ele esta disposto a oferecer indiretamente.

O sistema da equação (3.17) pode ser discretizado pelo método da resposta ao impulso, portanto

$$\mathbf{x}_c(k+1) = A \mathbf{x}_c(k) + B \mathbf{u}_c(k) \quad (3.18)$$

na qual,

$$A = \sum_{k=0}^{\infty} \frac{A_c^k T_c^k}{k!}$$

$$B = \int_0^{T_c} (e^{A_c \tau} d\tau) B_c$$

na qual, $\mathbf{u}_c = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 \end{bmatrix}^T$, T_c é o período de amostragem. Se A_c não é singular $B = A_c^{-1}(A_c - I)B_c$.

Como saídas do controle do sistema temos que o objetivo principal do sistema de controle é manter θ_e e e_{cg} em zero, mas dado que \dot{e}_{cg} controla indiretamente o comportamento do v_y do veículo, que não pode ser por características físicas do veículo zero em todo momento, também vai ser considerada uma saída de controle. Finalmente depois de simulações e análises dos tempos de resposta foi necessário considerar uma referência para δ_f aproximada que permitiu que o ponto ótimo calculado pelo controlador esteja mais perto de uma resposta suave (*smooth*)

para o eixo da direção do veículo, o que por sua vez aumenta a segurança intrínseca do veículo. Portanto temos:

$$\mathbf{y} = \begin{bmatrix} e_{cg} \\ \dot{e}_{cg} \\ \theta_e \\ \delta_f \end{bmatrix} \quad (3.19)$$

3.3 Estimador de estados

Como estimador dos estados pela medição foi projetado um filtro de Kalman estendido (EKF). O filtro de Kalman pode ser descrito como o algoritmo recursivo para gerar uma estimativa ótima no sentido estatístico das variáveis de estado do sistema assumindo que o sistema e os aparelhos de medição apresentam um ruído Gaussiano e que o modelo de estados seja linear (SIEGWART *et al.*, 2011b), (CHOSSET, 2005). Já o filtro de Kalman estendido é uma aproximação para modelos não lineares.

Para o filtro de Kalman estendido é necessário ter uma representação de estado estocásticas representada por (3.20), que é obtido combinando as equações (G.17), (G.32) e as fórmulas de Pacejka dos pneus, e fazendo uma discretização, utilizando a aproximação de Euler de primeira ordem. O modelo de estado resultante é dado por:

$$\begin{cases} \hat{\mathbf{x}}_k(k) = \hat{\mathbf{x}}_k(k-1) + T_k f(\hat{\mathbf{x}}_k(k-1), \mathbf{u}_k(k)) + \mathbf{w}_k \\ \mathbf{z}(k) = h(\hat{\mathbf{x}}_k(k)) + \mathbf{v}_k \end{cases} \quad (3.20)$$

na qual,

$$\hat{\mathbf{x}}_k = \begin{bmatrix} \hat{\mathbf{x}}_s & \hat{v}_y & \hat{\gamma} & x & y & \theta \end{bmatrix}^T$$

$\mathbf{u}_k = [\tau_{user}, V_m]^T$ é o vetor de entradas

T_k é o período de amostragem.

\mathbf{w}_k é o vetor de entradas gaussianas que caracterizam o ruído do modelo.

Para este estimador são consideradas as seguintes entradas de ruído Gaussiana que tem

como características media zero e variança σ :

- Para as variáveis de estado $\hat{\mathbf{x}}_s$ são consideradas as mesmas entradas de (G.33), as quais tem um vetor de varianças $\mathbf{w}_{k,s} = \left[\sigma_{v_m} \quad \sigma_{\tau_i} \quad \sigma_{\tau_f} \right]^T$, portanto

$$B_{k,s} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{r_g k_i \eta}{J_s R} & \frac{1}{J_s} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{-1}{J_w} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.21)$$

- Para as equações das variáveis de estado \hat{v}_y e $\hat{\gamma}$ são considerados que existem varianças nas forças f_{fy} e f_{ry} , com valores σ_{fy} e σ_{ry} respetivamente, portanto para pequenos ângulos de δ_f temos

$$B_{k,,fy} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{m_v} & \frac{1}{m_v} \\ \frac{l_f}{J_v} & \frac{l_r}{J_v} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (3.22)$$

Como o método de discretização é o método de Euler de primeira ordem, a covariança em tempo discreto estará definida pela equação (3.23), definindo $B_k = [[B_{k,s} : B_{k,fy}]]$ e o vetor $\sigma_{\mathbf{wk}} = \left[\sigma_{v_m} \quad \sigma_{\tau_i} \quad \sigma_{\tau_f} \quad \sigma_{fy} \quad \sigma_{ry} \right]$.

$$Q_k = T_c^2 B_k B_k^T \sigma_{\mathbf{wk}} \quad (3.23)$$

z é o vetor de sensores no qual é atualizado a uma taxa de amostragem de $5ms$ composto pelo sensor Encoder do atuador da direção δ_j e $\dot{\delta}_j$ e a aceleração lateral a_y que existe no Microautobox. Finalmente, de acordo as simulações é necessário ter um sensor de orientação, portanto o vetor de medições será descrito por (3.24).

$$Z_{1,k} = \begin{pmatrix} \delta_j \\ \dot{\delta}_j \\ -\frac{\gamma(c_f l_f - c_r l_r)}{v_x} - c_f \delta_f + \frac{v_y (c_f + c_r)}{v_x} \\ \theta \end{pmatrix} \quad (3.24)$$

v_k é o vetor de entradas gaussianas que caracterizam o ruído dos sensores os quais tem um vetor de variâncias $\sigma_{v_k} = [\sigma_{\delta_f} \quad \sigma_{\dot{\delta}_f} \quad \sigma_{a_y} \quad \sigma_{\theta}]^T$, portanto a matriz de covariâncias do erro de medição é chamado de R_k , onde foi assumindo que os erro não estão correlacionados $R_k = \text{diag}(\sigma_{v_k})$.

Finalmente, o algoritmo realizado, apresentado no algoritmo 1, tem as matrizes F e H que são geradas a partir da linearização do modelo de estados (3.20) e $P_k(0)$ que é a covariância inicial da estimação.

$$H = \nabla_x h(x) \quad (3.25)$$

$$F = \nabla_x x_{k+1}(x, U) \left| \begin{array}{l} x = \hat{x}_k \\ U = u_k \end{array} \right. \approx \hat{x}_{k-1} + T_k \nabla_x f(x_{k-1}, U_k) \quad (3.26)$$

3.4 Indicadores de desempenho do controle de trajetórias

O processo de seleção dos parâmetros de controle, assim como a comparação com outras técnicas de controle requer ter um padrão de testes, para o qual foi selecionada a trajetória do teste *VDA-Test- Double lane change*, definido pela norma ISO-3888-2:2008, *Passenger cars — Test track for a severe lane change maneuver Part 1 – Double lane-change* (figura 3.4) a partir de diferentes testes padrão da industria automotiva (FORKENBROCK *et al.*, 2003a). Este teste foi concebido como um meio de avaliar subjetivamente o desempenho dinâmico do veículo . O

Data: $F, H, Q_k, R_k, P_k(0), \hat{\mathbf{x}}_k(0), z_k$

Result: $\hat{\mathbf{x}}_k$

if measure then

$$\begin{aligned} P_k &\leftarrow F P_k F' + Q_k; \\ K &\leftarrow P_k H' (H P_k H' + R_k)^{-1}; \\ P_k &\leftarrow P_k - K P_k C'; \\ \hat{\mathbf{x}}_k &\leftarrow \hat{\mathbf{x}}_k + T_k f(\hat{\mathbf{x}}_k, \mathbf{u}_k) - K (z - h(\hat{\mathbf{x}}_k)); \end{aligned}$$

else

$$\begin{aligned} P_k &\leftarrow F P_k F' + Q_k; \\ \hat{\mathbf{x}}_k &\leftarrow \hat{\mathbf{x}}_k + T_k f(\hat{\mathbf{x}}_k, \mathbf{u}_k); \end{aligned}$$

end

Algoritmo 1: Filtro de Kalman Estendido

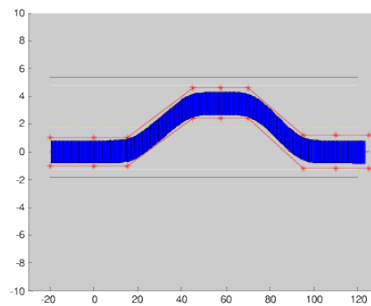


Figura 3.4: VDA test path. as linhas vermelhas são os limites de *VDA test* representam os retângulos azuis é o veículo durante as simulações

teste é subjectivo porque só quantifica uma pequena parte das características de manobrabilidade de um veículo e é altamente dependente do condutor (nosso caso o condutor e o sistema de controle). Esta dependência da experiência do condutor é o que faz o teste atraente para adaptação ao teste de habilidade de condução de veículo autônomos (KOON E WHITTAKER, 2006).

Selecionada a trajetória foi necessário definir indicadores de desempenho que permitam comparar resultados. Os indicadores são:

- $|e_{cg}|^{max}$ e $\overline{|e_{cg}|}$ são o módulo do valor máximo e a valor médio do erro lateral. O planejamento de trajetórias deve levar em conta um erro lateral médio sobre a trajetória para os fatores de segurança.
- $|\theta_e|^{max}$ e $\overline{|\theta_e|}$ são o módulo do valor máximo e a valor médio do erro de orientação. Erros de orientação grandes podem afetar os campos de visão dos sensores gerando zonas de difícil percepção.
- $|\dot{\delta}_j|^{max}$ e $|\ddot{\delta}_j|^{max}$ são os módulos dos valores máximos da velocidade e aceleração da

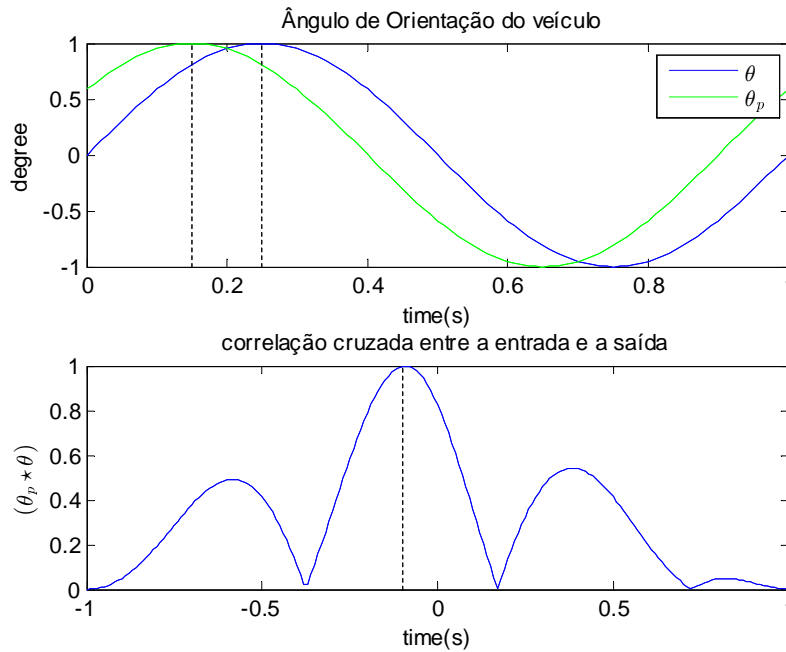


Figura 3.5: Exemplo correlação cruzada entre θ_p e θ

atuador da direção, pois estas variáveis estão relacionadas com as capacidades do atuador e indiretamente com o torque requerido.

- O tempo de defasagem entre a orientação desejada (θ_p) e a orientação do veículo (θ) é estimado como o tempo em que ocorre o valor máximo da correlação cruzada entre os dois sinais. Uma grande defasagem dos sinais pode ser gerado por problemas na estimação da dinâmica do modelo. Para compreender o indicador, na figura 3.5 estão θ_p e θ como sinais sinusoidais puros com uma defasagem de $0.1s$ e a correlação cruzada onde é possível ver que o valor máximo da correlação cruzada ocorre nesse tempo. A equação (3.27) apresenta como é realizado o cálculo.

$$|(\theta_p \star \theta)|^{Max} = \left| \sum_{m=0}^{t_f/T} \theta(k) \theta_p(k+m) \right|^{Max} \implies t_{defasagem} = k_{|(\theta_p \star \theta)|^{Max}} T \quad (3.27)$$

3.5 Simulações do controle cooperativo de trajetórias

Para validar o controle de trajetórias foram realizadas simulações usando Simulink/Matlab com ODE4 (runge-kutta) como método de solução das equações de movimento do veículo descrito no anexo G.2, implementado em uma frequência de ciclo de $1kHz$, com o objetivo de

ser compilado no sistema embarcado MicroAutobox. Neste mesmo processo é implementado o estimador das variáveis de estado, o filtro estendido de Kalman. Em outro processo a uma frequência de ciclo 100hz são implementadas as funções do sistema de controle automático de trajetórias e o controlador que emula o condutor, assim como o interpolador das *splines* geradas pelo gerador de trajetórias (ver capítulo 4). Este processo pode ser visualizado no diagrama de blocos da figura 3.6.

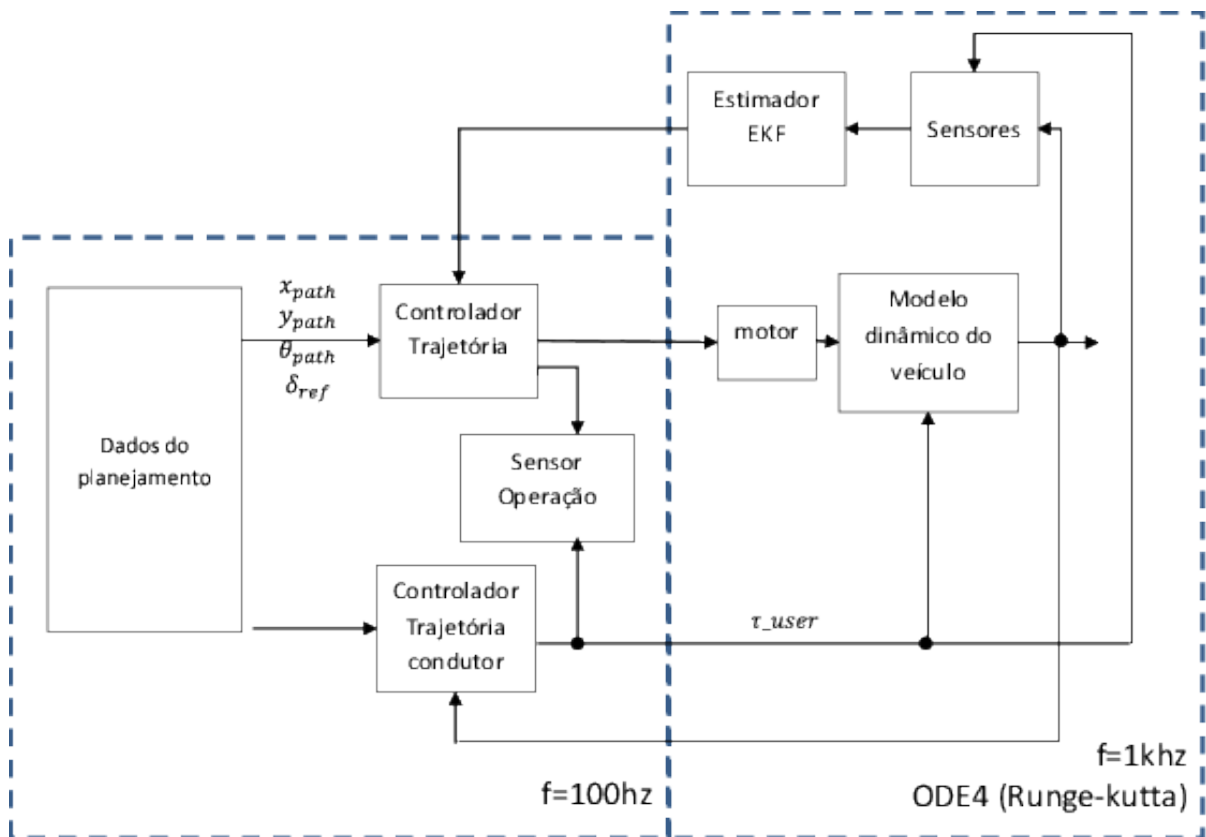


Figura 3.6: Diagrama de blocos da simulação para o controlador de trajetórias.

Os dados para a simulação são listados a seguir:

- Velocidade inicial e da trajetória $v_x = 10\text{m/s}$.
- Configurações do filtro estendido de Kalman. Com uma aproximação de Euler na discretização, o período do cálculo do filtro é $T_k = 1\text{ms}$, o mesmo utilizado para os sensores do motor de direção, diferente dos sensores de orientação e o acelerômetro lateral que utilizam um período de 5ms . Os ruídos do modelo e dos sensores são descritos na equação

(3.28) e (3.29).

$$\begin{aligned}\sigma_{\mathbf{vk}} &= \left[\sigma_{\delta_f} \quad \sigma_{\dot{\delta}_f} \quad \sigma_{a_y} \quad \sigma_{\theta} \right]^T \\ &= \left[(0.1^\circ)^2 \quad (1^\circ/s)^2 \quad (0.3m/s^2)^2 \quad (0.5^\circ)^2 \right]^T\end{aligned}\quad (3.28)$$

$$\begin{aligned}\sigma_{\mathbf{wk}} &= \left[\sigma_{v_m} \quad \sigma_{\tau_i} \quad \sigma_{\tau_f} \quad \sigma_{f_y} \quad \sigma_{r_y} \right]^T \\ &= \left[(0.1V)^2 \quad (0.5Nm)^2 \quad (100N)^2 \quad (100N)^2 \right]^T\end{aligned}\quad (3.29)$$

- Controlador de trajetória automático. Período de amostragem $T_c = 10ms$, horizonte de controle $H_c = 8$, horizonte de predição $H_p = 10$, para a matriz de ponderação das saídas do controle são normalizados os valores através do valor máximo desejado, de tal forma que todos tenham a mesma ponderação na resolução desejada, como é apresentado na equação (3.30).

$$\begin{aligned}Q_{mpc} &= \text{diag} \left(\left(\frac{1}{\theta_e^{max}} \right)^2, \left(\frac{1}{e_{cg}^{max}} \right)^2, \left(\frac{1}{\dot{e}_{cg}^{max}} \right)^2, \left(\frac{1}{\delta_f^{max}} \right)^2 \right) \\ &= \text{diag} \left(\left(\frac{1}{8^\circ} \right)^2, \left(\frac{1}{0,07m} \right)^2, \left(\frac{T_c}{0,01m} \right)^2, \left(\frac{1}{2.5^\circ} \right)^2 \right)\end{aligned}\quad (3.30)$$

Nota-se que o erro permitido no sinal de controle de direção é alto dado que nos só temos uma aproximação do sinal de referência verdadeiro calculado pelo modelo cinemático do veículo. Nota-se também que se a saída de controle é maior do que o valor máximo, a variável de controle terá maior ponderação nessa iteração da otimização. Para a matriz de ponderação da entrada, temos que $R_{mpc} = v_{ref}/v_{max} \text{diag}(\alpha, \alpha\beta)$, onde $v_{max} = 40$ é o máximo valor do voltagem do motor, $\alpha = 4$ é a ponderação com respeito às saídas a controlar, e $\beta = 5$ é a relação entre as duas entradas. Dado que o estimador de estados nesta simulação não tem sensor de posição, a variável e_{cg} é calculada através da integral de \dot{e}_{cg} , mas ela poderia ser gerada pelo método descrito no planejamento de trajetórias para encontrar q ou através da fórmula do cálculo da distância cartesiana entre o ponto de referência e o ponto do carro.

- Controlador do condutor. O controlador do condutor, tem inicialmente a mesma configuração do controlador automático, portanto, as variáveis de estado serão as mesmas do estimador de estados. Adicionalmente é assumido que o controlador do condutor não

| v_m | $\overline{ e_{cg} }$ | e_{cg}^{max} | $\overline{ \theta_e }$ | θ_e^{max} | $ (\theta_p \star \theta) ^{Max}$ | $ \dot{\delta}_j ^{max}$ | $ \ddot{\delta}_j ^{max}$ |
|-------|-----------------------|----------------|-------------------------|------------------|-----------------------------------|--------------------------|---------------------------|
| m/s | m | m | $^\circ$ | $^\circ$ | s | $^\circ/s$ | rad/s^2 |
| 10 | 0,0462 | 0,0971 | 0,415 | 1,18 | -0,12 | 98,57 | 41,32 |

Tabela 3.1: Índices de desempenho do controlador com os parâmetros padrão.

consegue medir o valor real do sinal do motor, portanto ele estima através dos dados calculados pelo controlador.

- Sensor de torque do condutor. O sensor do torque do condutor é assumido com um período de amostragem de $10ms$ e modelado com um erro de medição com distribuição padrão de $N \sim (0,0.05)Nm$.

Com estas condições o comportamento das variáveis de estado pode ser descrito pelas figuras 3.7, 3.8 e 3.9. Na primeira figura é apresentado o comportamento das variáveis de erro de trajetória, e_{cg} , \dot{e}_{cg} , θ_e e $\dot{\theta}_e$, onde pode-se ver que a variável sobre a qual se tem maior ponderação é o erro lateral, a qual tem um erro de deriva que pode ser causado principalmente pela falta de um sensor de posição absoluta, entretanto o erro de orientação como tem um sensor absoluto não tem erro de deriva perceptível. Na segunda figura são apresentados as variáveis do sistema de direção do veículo, onde existe um ruído grande nas velocidades causado principalmente porque o torque do usuário não foi filtrado previamente, ruído que não é percebido pelas variáveis de ângulo porque elas automaticamente filtram o ruído pelo processo de integração. Sobre o sinal de referência de ângulo δ_{ref} pode-se observar que o controle tem um tempo de resposta para seguir as trocas de sinal dado que o sistema dinâmico não está contemplado no planejamento de trajetórias. Na terceira figura são apresentadas as variáveis do veículo, onde se percebe que o atraso do ângulo de guinada θ , assim como um tempo de resposta em seguir as trocas de sinal da velocidade de guinada desejada $r(s)$ que são causados pela dinâmica própria do veículo que funciona como um filtro, o que por sua vez gera problemas na estimação da velocidade lateral v_y . Os indicadores de desempenho desta simulação são apresentados na tabela 3.1.

Como sinais de controle, temos o controle do condutor e a voltagem do motor, seu comportamento é descrito na figura 3.10, onde na primeira sub figura, podemos ver as duas variáveis em torque (através das equações do motor), e na segunda vemos em escalas diferentes a voltagem do motor e o valor da contribuição do condutor (em Nm) que está com aproximadamente um quinto (β) da parte do sinal de controle com respeito ao valor de voltagem gerado pelo motor (em V). A seguir são apresentadas diferentes simulações de testes do comportamento do sistema de controle, com diferentes velocidades, com diferentes condutores virtuais, com diferentes sensores na estimação e finalmente com diferentes sinais de referência entre o condutor

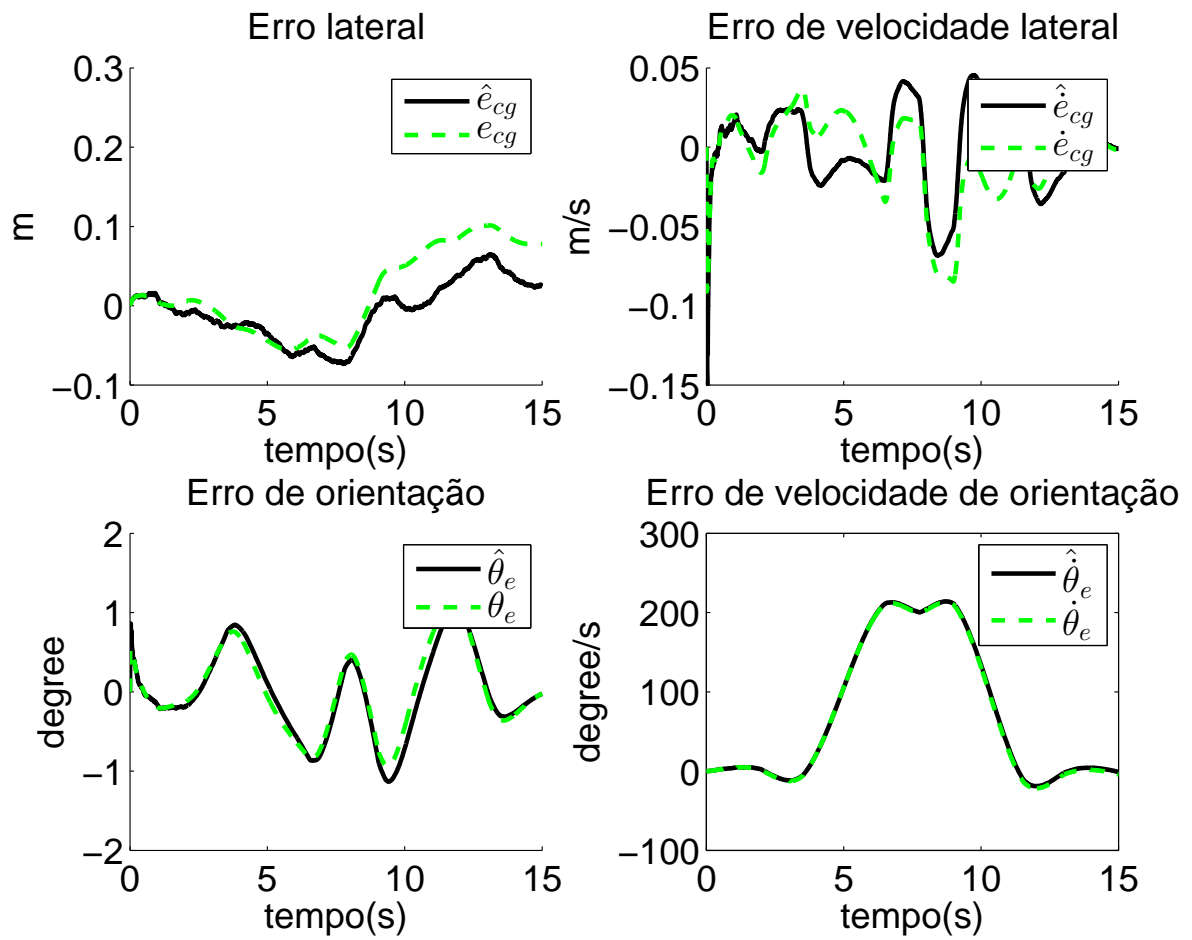


Figura 3.7: Erros na trajetória recorrida pelo veículo

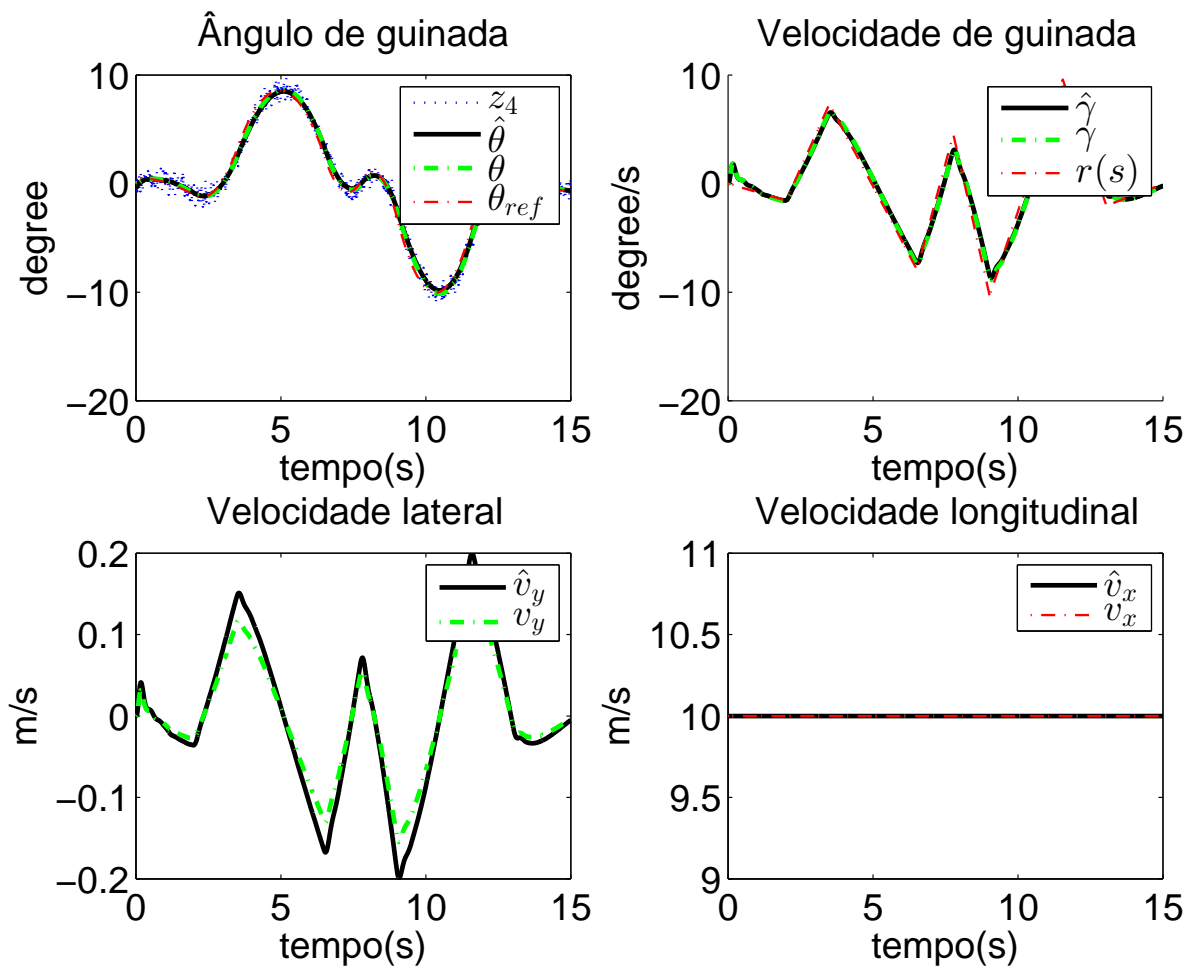


Figura 3.8: Variáveis de estado do sistema de direção do veículo

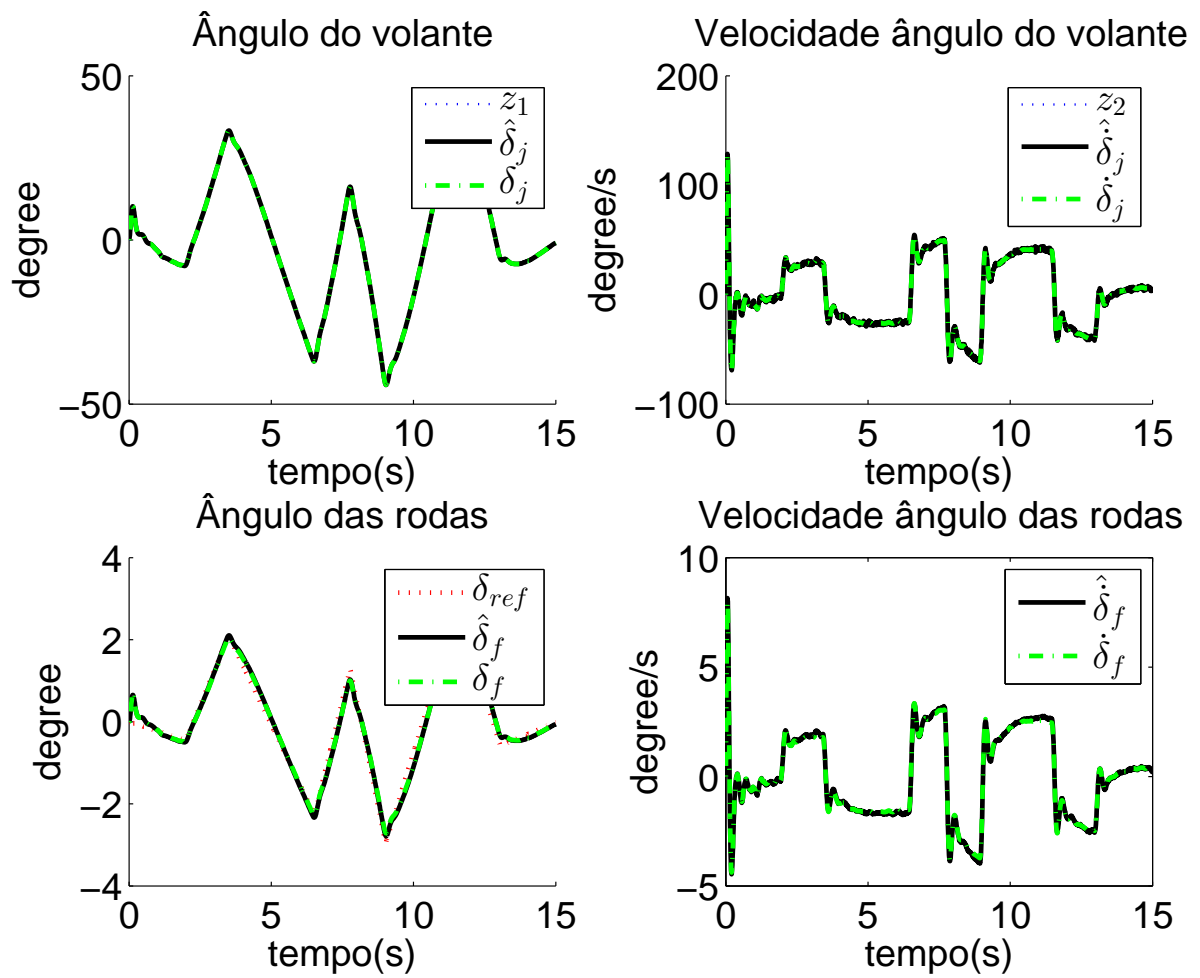


Figura 3.9: Variáveis de estado do veículo

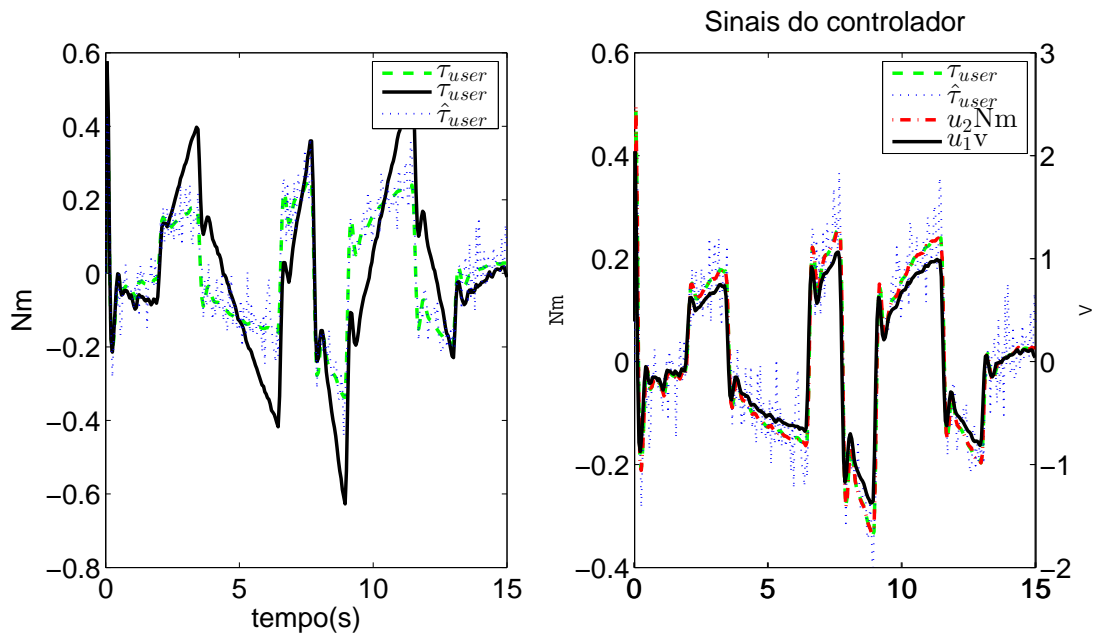


Figura 3.10: Sinais de controle do controle de trajetórias.

e controlador.

3.5.1 Simulações a diferentes velocidades

Na Tabela 3.2 pode-se ver o comportamento do controlador com diferentes velocidades desejadas na trajetória. O controlador vai requerer maiores esforços de controle (marcados pelos indicadores das velocidades do eixo do volante) a maiores velocidades longitudinais, o que faz com que o controlador a velocidades longitudinais maiores de 16,3 m/s começou a gerar diferenças nos indicadores de desempenho. Pode-se explicar esta diferença por duas causas, a primeira é que o motor da direção não vai conseguir atingir as acelerações descritas, assim como o tempo de resposta do sistema dinâmico da direção, e a segunda é que o perfil do ângulo da roda gerado pelo planejamento de trajetórias é baseado em um modelo cinemático que diverge a velocidades altas do modelo dinâmico usado tanto no controlador como no modelo de simulação do carro.

A figura 3.11, mostra como o comportamento desta variável controlada pode ser trocada acrescentando o valor de $Q_{mpc}^{1,1}$ de 50 para 200, mas isso aumenta o atraso com respeito a referência ao ângulo de guinada do veículo. Sobre este indicador também pode ser ressaltado que quanto menor a velocidade maior é o tempo de atraso, o que pode ser explicado porque a dinâmica do veículo é mais lenta a menor velocidade, o que faz com que exista uma defasa-

| v_m | $\overline{ e_{cg} }$ | e_{cg}^{max} | $\overline{ \theta_e }$ | θ_e^{max} | $ (\theta_p \star \theta) ^{Max}$ | $ \dot{\delta}_j ^{max}$ | $ \ddot{\delta}_j ^{max}$ |
|-------|-----------------------|----------------|-------------------------|------------------|-----------------------------------|--------------------------|---------------------------|
| m/s | m | m | $^\circ$ | $^\circ$ | s | $^\circ/s$ | rad/s^2 |
| 5 † | 0.0472 | 0,138 | 0.367 | 0.925 | -0.16 | 151.3 | 81.8 |
| 5 | 0.0306 | 0.108 | 0.529 | 1.39 | -0.28 | 62.18 | 33.04 |
| 10 | 0.0462 | 0.0972 | 0.414 | 1.18 | -0.12 | 93.14 | 36.72 |
| 13.3 | 0.0293 | 0.0673 | 0.396 | 1.04 | -0.08 | 118.2 | 37.4 |
| 15 | 0.0106 | 0.0332 | 0.344 | 0.919 | -0.07 | 173.8 | 37.67 |
| 16.7 | 0.0303 | 0.0651 | 0.331 | 0.996 | -0.06 | 156.9 | 67.81 |
| 20 | 0.123 | 0.243 | 0.374 | 1.16 | -0.04 | 266.3 | 68.87 |
| 20* | 0,115 | 0.269 | 0.754 | 1.95 | -0.1 | 159.2 | 48.85 |
| 25 | 0.248 | 0.653 | 0.854 | 2.77 | 0 | 392.7 | 69.55 |

Tabela 3.2: Índices de desempenho do controlador, sem o condutor ($\beta = 2000$), com respeito a velocidade desejada. dados de † $Q_{mpc}^{4,4} = 1600$, * $Q_{mpc}^{1,1} = 200$

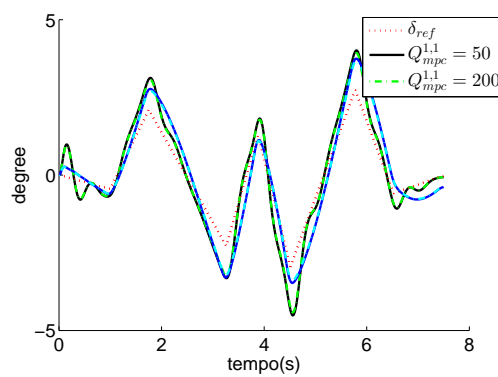


Figura 3.11: Ângulo de estercamento das rodas(δ_f) com $v_x = 20m/s$ e diferentes valores de ponderação das variáveis de controle $Q_{mpc}^{1,1}$

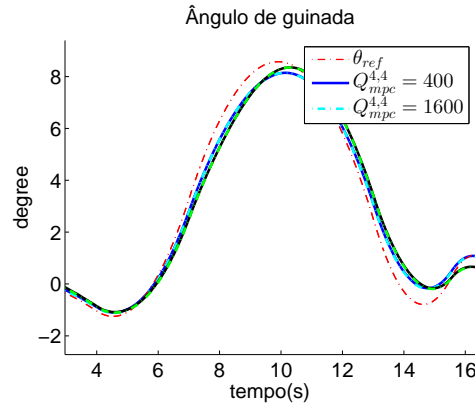


Figura 3.12: Ângulo de guinada do veículo (θ) com $v_x = 5m/s$ e diferentes valores de ponderação das variáveis de controle $Q_{mpc}^{4,4}$

| Configuração | $\overline{ e_{cg} }$ | e_{cg}^{max} | $\overline{ \theta_e }$ | θ_e^{max} | $ (\theta_p \star \theta) ^{Max}$ | $ \dot{\delta}_j ^{max}$ | $ \ddot{\delta}_j ^{max}$ |
|---------------------------------|-----------------------|----------------|-------------------------|------------------|-----------------------------------|--------------------------|---------------------------|
| | m | m | $^\circ$ | $^\circ$ | s | $^\circ/s$ | rad/s^2 |
| Giroscópio $T_s = 20ms$ | 0,046 | 0,0969 | 0,415 | 1,18 | -0,12 | 103,4 | 46,43 |
| Giroscópio GPS $T_s = 1s$ | 0,0277 | 0,065 | 0,402 | 1,05 | -0,12 | 101,8 | 45,49 |

Tabela 3.3: Índices de desempenho do controlador com diferentes sensores no estimador de estados.

gem maior entre a sinal desejado e o comportamento do veículo. A figura 3.12 mostra como aumentando o valor de $Q_{mpc}^{4,4}$ de 400 para 1600, o que significa quanto maior a ponderação de controle da orientação menor é o atraso do ângulo de guinada, mas gera necessidade de maiores esforços de controle e faz com que o máximo erro lateral seja amplificado. Do exposto anteriormente pode-se concluir que existe correlação entre a velocidade do veículo e o valor das matrizes de ponderação Q_{mpc} e R_{mpc} , portanto deve-se utilizar um valor diferente de acordo com a velocidade, assim como também de acordo com o objetivo principal do controlador, seja ele erro lateral, erro de orientação ou esforço do controlador. Por exemplo em estradas de alta velocidade o maior objetivo deve ser o erro lateral para funcionar como um ACC (*Active Cruise Control*), mas na cidade onde ele deve reagir rapidamente a objetos na rua o atraso no ângulo de guinada pode ser um valor mais importante. Adicionalmente a estas simulações foram feitas simulações com perfis de aceleração ou desaceleração constante sem ser percebido alguma troca no comportamento do sistema.

3.5.2 Simulações com outros sensores no estimador de estados

O primeira opção é ser inserido um giroscópio a uma frequência de 50Hz, portanto a cada 20 iterações o vetor $\mathbf{z}(k)$ trocará sua dimensão de 4 para 5, mudando a matriz H que ficará descrita pela equação (3.31). Finalmente, também trocará a matriz R_k dado que o vetor $\sigma_{\mathbf{v}k}$, tem o erro, assumido gaussiano, do giroscópio de desvio padrão de $1^\circ/s$.

$$H_{gyro} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{c_f}{m} & 0 & -\frac{c_f+c_r}{m \text{ vel}} & -\frac{c_f l_f - c_r l_r}{m \text{ vel}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (3.31)$$

Quando o GPS é inserido no estimador de estados com um desvio padrão de $1m$ em cada eixo que se adiciona ao vetor $\sigma_{\mathbf{v}k}$, a matriz H será descrita pela equação 3.32. Neste caso, o e_{cg} é calculado através do erro lateral entre a posição estimada (\hat{x}, \hat{y}) e a desejada (x_{ref}, y_{ref}) , pela equação (3.33), que consiste em utilizar a projeção dos dois vetores depois de ser rotacionados com o angulo da referência, θ_{ref} , e utilizar esta componente projetada perpendicular à trajetória nesse ponto. Como pode-se ver na figura 3.14, o erro de deriva diminui no erro lateral. Inclusive se o carro depois da manobra continuar reto o estimador consegue diminuir o erro de deriva quase a zero, mas ele requer bastante tempo porque o tempo de amostragem do sensor do GPS é de $1s$.

$$H_{gyro+gps} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{c_f}{m} & 0 & -\frac{c_f+c_r}{m \text{ vel}} & -\frac{c_f l_f - c_r l_r}{m \text{ vel}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \quad (3.32)$$

$$e_{cg} = (\hat{y} \cos \theta_{ref} - \hat{x} \sin \theta_{ref}) - (y_{ref} \cos \theta_{ref} - x_{ref} \sin \theta_{ref}) \quad (3.33)$$

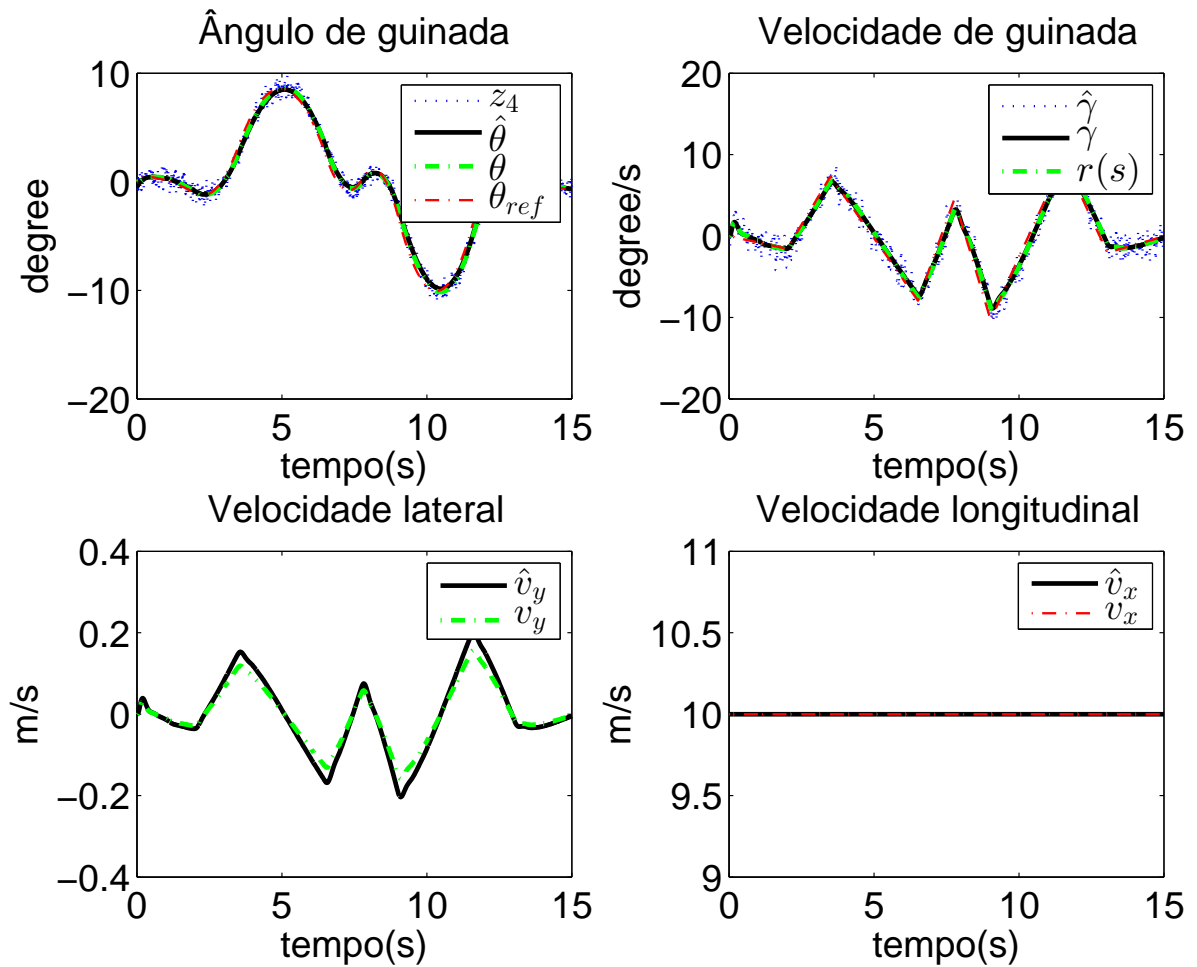


Figura 3.13: Variáveis de estado do veículo quando o estimador de estado tem giroscópio

| Configuração | $\overline{ e_{cg} }$ m | e_{cg}^{max} m | $\overline{ \theta_e }$ $^\circ$ | θ_e^{max} $^\circ$ | $ (\theta_p \star \theta) ^{Max}$ s | $ \dot{\delta}_j ^{max}$ $^\circ/s$ | $ \ddot{\delta}_j ^{max}$ rad/s^2 |
|---|------------------------------|-----------------------|-------------------------------------|------------------------------|--|--|--|
| $\beta = 1$ | 0,0461 | 0,0973 | 0,417 | 1,18 | -0,12 | 115,8 | 59,65 |
| $\beta = 0,1$ | 0,046 | 0,0976 | 0,42 | 1,2 | -0,12 | 172,9 | 91,5 |
| v_m não existe $\beta = 1$ | 0,046 | 0,079 | 0,422 | 1,19 | -0,12 | 131,9 | 63,38 |
| $Q_{mpc}^{4,4} = 0$ $,\beta = 5$ | 0,0462 | 0,968 | 0,414 | 1,28 | -0,12 | 96,12 | 39,54 |
| x e o real $,\beta = 5$ | 0,461 | 0,975 | 0,416 | 1,18 | -0,12 | 95,94 | 39,52 |
| x e o real $,\beta = 5$ $Q_{mpc}^{4,4} = 0$ | 0,461 | 0,971 | 0,416 | 1,18 | -0,12 | 94,37 | 38,35 |

Tabela 3.4: Índices de desempenho do controlador com diferentes configurações do controlador do condutor a $v_x = 10m/s$

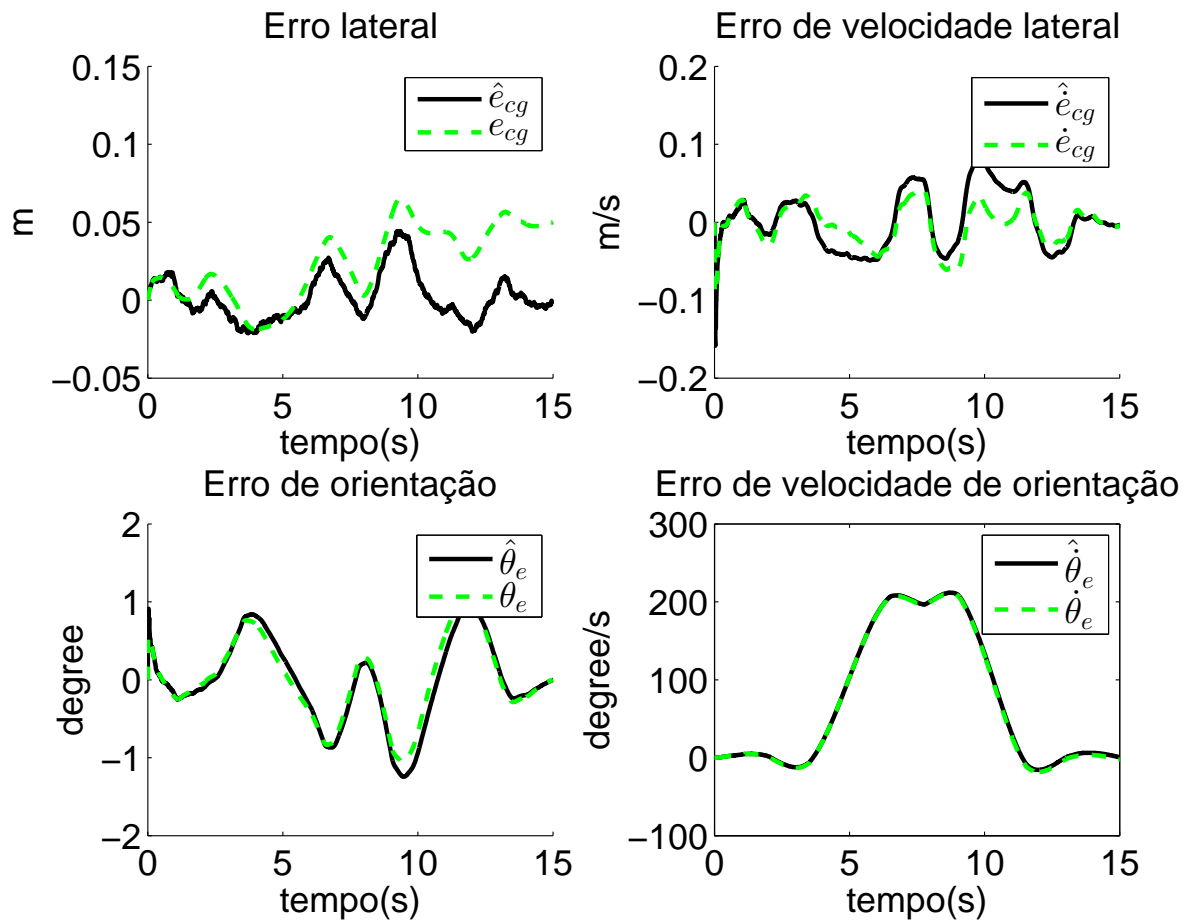


Figura 3.14: Variáveis de estado de erro de trajetória quando o estimador de estado tem giroscópio mais um GPS

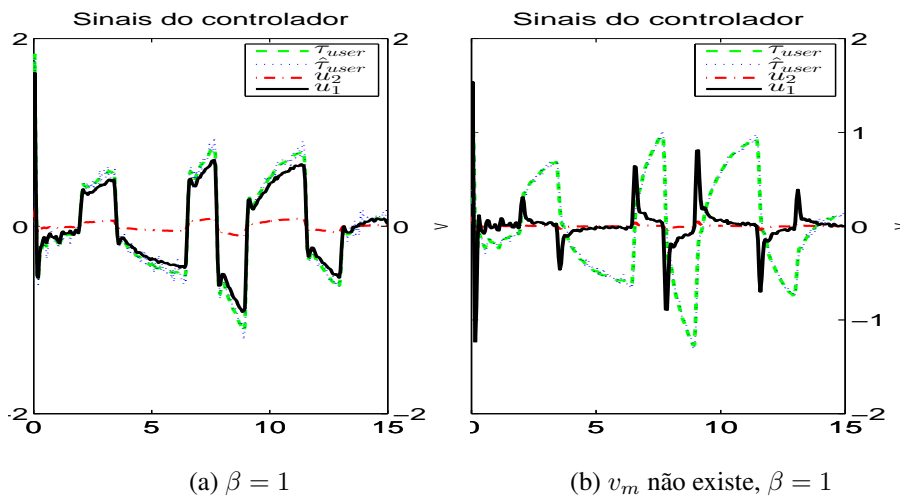


Figura 3.15: Sinal de controle do controlador do condutor e do controlador automático.

3.5.3 Simulações com diferentes estratégias de controle do condutor

Na Tabela 3.4 pode-se ver o comportamento do controlador com controladores do condutor com objetivos de controle diferentes, mas com o mesmo sinal de referência. O comportamento em todos os indicadores de desempenho é muito similar. Porém, se analisamos o sinal de controle, podemos ver, de acordo a figura 3.15a, que para o caso $\beta = 1$, os sinais de controle tem a mesma magnitude, portanto o sistema de controle automático consegue se adaptar ao requerido pelo condutor. Isto pode ser explicado porque ao conhecer o sinal de torque do condutor, τ_{user} , ele vai funcionar como um sinal *feedforward*, porém os valores futuros vão estar aumentando de uma forma menor da real porque o β do controle automático é maior. Outro caso que pode acontecer é que o controlador do condutor não perceba o sinal do motor dentro do projeto do controlador (v_m não existe); Nesse caso, descrito pela figura 3.15b, o sinal de controle do condutor tem o valor completo do sinal de controle requerido, no entanto o controlador do motor ainda consegue ajudar nas trocas rápidas de magnitude requeridas.

Na figura 3.16a, podemos ver as respostas de um controlador do condutor sem o objetivo de controlar a orientação do carro, o que significa que $Q_{mpc}^{4,4} = 0$. Nesse caso vemos como a forma do sinal de τ_{user} é diferente (verde versus vermelha na figura), mas a ordem de magnitude dos sinais é a mesma. Finalmente, na figura 3.16b, pode-se observar o sinal de controle quando o controlador do condutor é retroalimentado com o sinal real, \mathbf{x} , e não pelo sinal estimado, $\hat{\mathbf{x}}$. Neste caso observamos que o sistema começa comportando-se da mesma forma, mas quando o erro de deriva da estimação de e_{cg} começa a crescer, os sinais começam a divergir para compensar a diferença de estimação. Dado que o controlador automático tem mais informação porque

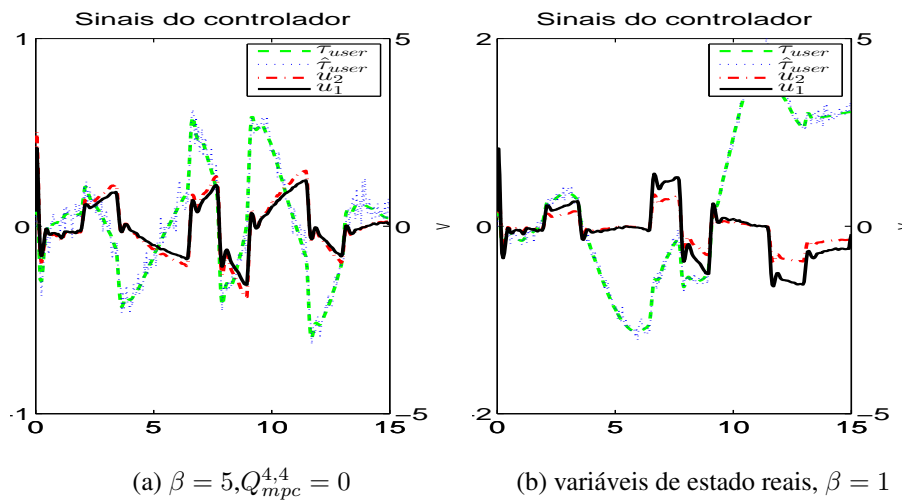


Figura 3.16: Sinal de controle do controlador do condutor e do controlador automático.

conhece o torque do usuário, ele vai dominar o comportamento do sistema com a trajetória.

3.6 Considerações finais

O presente capítulo apresentou a proposta de controle de trajetórias do veículo autônomo pela técnica de controle preditivo baseado em modelo e um estimador de estados baseado pela técnica de filtro de Kalman estendido. Esta técnica permitiu projetar um controlador cooperativo entre o condutor e o motor da direção da automação do veículo. A técnica selecionada foi simulada no sistema embarcado através de *software in the loop* usando um modelo dinâmico não linear para o veículo e um controlador preditivo para simular o condutor baseado na proposta (KEEN, 2008). O controlador é avaliado ante diferentes velocidades do veículo, diferentes sensores para a estimação de estados e o comportamento do controlador automático com controladores do condutor com objetivos de controle diferentes, mas com o mesmo sinal de referência. Nos capítulo a seguir é apresentado como é gerada o sinal de referência (a trajetória desejada) pelo veículo autônomo de acordo a indicadores de risco.

4 Planejamento de Trajetórias

Veículos autônomos, que genericamente são entendidos como robôs autônomos, tem aspectos de sua cognição diretamente relacionada com uma mobilidade robusta que é competência da navegação (SIEGWART *et al.*, 2011b). De acordo com a literatura, costuma-se dividir a navegação entre planejamento e reação. O planejamento global é responsável pelo planejamento e considerando que os veículos comerciais atuam em ambientes estruturados e obedecem a restrições não holonômicas, o planejamento local é responsável pela área de reação da navegação. O primeiro envolve a geração de uma trajetória que guiará o veículo à uma localização alvo, isso é realizado a partir de informações prévias do ambiente (mapa) e do usuário de maneira que seja possível considerar fatores como distância, velocidade e outros. Essa abordagem voltada para aplicações automotivas foi mais recentemente estudado por (CHU *et al.*; LI *et al.*; WERLING *et al.*; LEE *et al.*; HOWARD E KELLY, 2012; 2014; 2010; 2014; 2007). Neste trabalho partimos do pressuposto que o caminho global ou o planejamento da missão já se encontra disponível.

O capítulo apresenta o algoritmo sobre planejamento de trajetórias implementado no trabalho, o qual é baseado em (CHU *et al.*, 2012), da equipe vencedora do 2010 *Autonomous Vehicle Competition* organizada pelo grupo automotivo Hyundai Kia da Coreia (CHU *et al.*, 2012) propor um planejamento local de caminhos que fornece ao veículo autônomo a capacidade de desviar de obstáculos estáticos quando seguindo um caminho global predefinido. O presente capítulo utiliza a abordagem de (CHU *et al.*, 2012) que propõem um planejamento local de caminhos que fornece ao veículo autônomo não só a capacidade de desviar de obstáculos estáticos, como também de desviar de obstáculos em movimento. A mesma metodologia de planejamento será utilizada para entender o risco do condutor para assumir o comando ao veículo que está em modo autônomo ou cooperativo, permitindo ou não através da análise as transições de automático ou cooperativo para manual, garantindo assim que o veículo venha seguir sempre uma trajetória segura.

Diferentes critérios podem ser empregados para selecionar o melhor caminho de acordo com a aplicação desejada. O algoritmo de Chu (CHU *et al.*, 2012) usa um critério de custo de segurança dos caminhos candidatos. Como a estimativa do centro da estrada pode ser ambígua e/ou heurística para se avaliar a segurança dos caminhos em alguns cenários, o algoritmo proposto não especifica diretamente o centro da estrada. Em vez disso, a segurança de cada caminho é quantitativamente avaliada com uma convolução da distância entre os caminhos avaliados e os caminhos com obstáculos estáticos, o que descarta a necessidade de se estimar no detalhe as distâncias de todos os pontos pertencentes aos caminhos com obstáculos. A avaliação de obstáculos móveis é adicionada neste trabalho seguindo a mesma metodologia. Também são

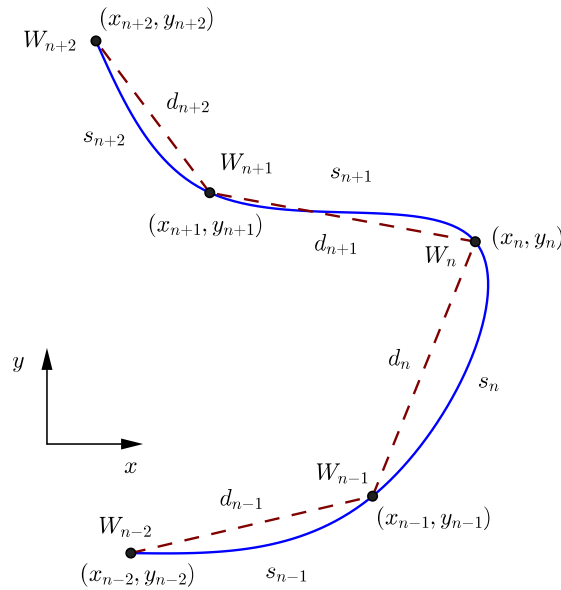


Figura 4.1: Reparametrização no arco da distancia das *splines* do caminho global

analisadas a suavidade e convergência dos caminhos a partir de outras funções custos, sendo as funções de segurança, suavidade e consistência minimizadas juntas em um algoritmo de otimização. Essa abordagem permite ao veículo autônomo uma condução mais estável.

De acordo a metodologia apresentada em (CHU *et al.*, 2012), a seguir será apresentada o algoritmo implementada na seguinte ordem: Construção do espaço curvilíneo global, localização do veículo no espaço curvilíneo global, geração do caminho candidato, seleção de um caminho candidato e determinação do perfil de velocidade. Finalmente, serão apresentados os resultados da implementação.

4.1 Construção do espaço curvilíneo global

O primeiro requerimento consiste em transformar um conjunto discreto de pontos em uma função contínua do caminho. Os pontos do caminho gerados no planejamento de caminho global são interpolados por *splines* cúbicas bidimensionais parametrizadas no arco da distância, distância de viagem do caminho global, s_i , para formar um espaço de trabalho. As curvas parametrizadas são frequentemente utilizadas para modelar estradas e para se definir caminhos de referência.

Para se gerar as *splines* cúbicas parametrizadas no arco da distância a partir dos pontos do

caminho global, primeiramente constrói-se uma curva parametrizada em um parâmetro qualquer d , neste caso usando a distancia euclideana entre os *waypoints*, e depois procedesse com sua reparametrização para o arco da distância, s . Para realizar a parametrização no arco da distância, os arco das distância entre os pontos do caminho global precisam ser calculados, o que é feito a partir do calculo numérico da equação 4.1 pelo método de Runge-Kutta

$$s_i = \int_{t_i}^{t_f} \left(\sqrt{\left(\frac{dx(d)}{dt}\right)^2 + \left(\frac{dy(d)}{dt}\right)^2} \right) dt \quad (4.1)$$

na qual $x(d)$ e $y(d)$ são as *splines* cúbicas a partir da distância euclidiana entre cada ponto do caminho. Na figura 4.1 pode-se ver o procedimento.

Conhecidos todos os valores dos arcos das distâncias entre os pontos do caminho procedesse com a reparametrização, que é o processo de construção de uma nova curva a partir das *splines* cúbicas usando-se como variável independente os valores dos arcos das distância (s_i) calculados (WANG *et al.*, 2002). O processo gera a equação (4.2).

$$\begin{cases} x_b(s) = a_{x,i} (s - s_i)^3 + b_{x,i} (s - s_i)^2 + c_{x,i} (s - s_i) + d_{x,i} \\ y_b(s) = a_{y,i} (s - s_i)^3 + b_{y,i} (s - s_i)^2 + c_{y,i} (s - s_i) + d_{y,i} \end{cases} \quad (4.2)$$

Portanto, se o parâmetro s é dado, a posição correspondente $[x_b, y_b]^T$ no espaço de trabalho é diretamente calculado a partir da equação continua. Nota-se que a presente aproximação do arco da distância da curva tem alguns problemas de convergência, os quais foram mostrados num artigo apresentado no congresso Workshop on Engineering Applications em Bogota (DE LEMOS *et al.*, 2015). A tangente e a função da curvatura do caminho são calculadas a partir da primeira e da segunda derivadas da *splines* cúbicas parametrizadas no arco da distância de acordo com a equação (4.3).

$$\kappa_b = \frac{\frac{dx_b}{ds} \left(\frac{d^2 y_b}{ds^2} \right) - \frac{dy_b}{ds} \left(\frac{d^2 x_b}{ds^2} \right)}{\left(\left(\frac{dx_b}{ds} \right)^2 + \left(\frac{dy_b}{ds} \right)^2 \right)^{3/2}} \quad (4.3)$$

Como a *spline* cúbica parametrizada possui a segunda derivada contínua, a curvatura de todos os seus segmentos também é contínua. Esta é uma condição necessária dado que o modelo cinemático requer esta condição. Para entender esta afirmação suponha-se que o veículo se move

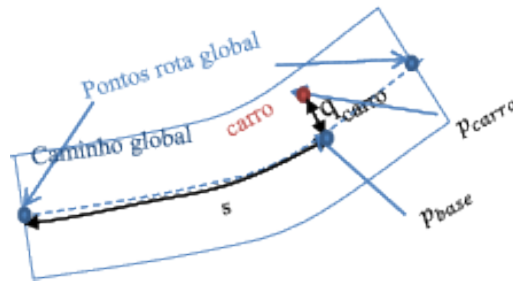


Figura 4.2: Localização de p_{carro} no espaço curvilíneo global em um segmento da estrada. Adaptada de (CHU *et al.*, 2012)

ao longo da curva com velocidade unitária, então a curvatura se define como uma medida da rapidez da troca do vetor unitário tangente a curva, desde que do ponto de vista geométrico seja descrito através do cálculo infinitesimal $\kappa = d\theta/ds$.

4.2 Localização no espaço curvilíneo global

Neste etapa a posição do veículo sobre o espaço curvilíneo global é determinada através da transformação da localização do veículo de coordenadas cartesianas para as coordenadas curvilíneas. O aspecto mais importante deste mapeamento é o cálculo do ponto mais próximo sobre o espaço de trabalho (p_{base}) ao ponto que está representado em coordenadas cartesianas (p_{carro}). O ponto mais próximo da posição do veículo é calculado numericamente utilizando-se o método de otimização de Newton (WANG *et al.*, 2002). Neste processo, o arco da distância (s) do ponto mais próximo ao veículo sobre o espaço curvilíneo é encontrado pela minimização da distância entre p_{carro} e p_{base} , chamada q_{carro} como mostrado na figura 4.2.

4.3 Geração dos caminhos candidatos

Nesta etapa, os caminhos candidatos são gerados pela integração do modelo cinemático do veículo a partir das informações de velocidade e orientação. Manobras suaves para desvio de obstáculos são projetadas no sistema de coordenadas curvilíneo do espaço de trabalho ao invés do sistema cartesiano. Dessa forma, o algoritmo de geração é dividido em duas partes, a primeira é encarregada da transformação entre os sistemas de coordenadas considerando-se restrição cinemáticas do veículo, e a segunda, é encarregada em descrever corretamente os caminhos no espaço curvilíneo.

4.3.1 Transformação entre o espaço curvilíneo e o cartesiano

O arco da distância do espaço de trabalho, o qual é a distância viajada pelo veículo no caminho global, se torna o eixo horizontal do sistema de coordenadas globais e o deslocamento lateral do veículo se torna o eixo vertical. Para a transformação entre os sistemas de coordenadas cartesianas e curvilíneas, a curvatura do caminho κ é determinada como apresentado em (WERLING *et al.*, 2010)(BARFOOT E CLARK, 2004) pela equação (4.4)

$$\kappa = \frac{S}{Q} \left(\kappa_b + \frac{(1 - q\kappa_b) \frac{d^2q}{ds^2} + \kappa_b \left(\frac{dq}{ds}\right)^2}{Q^2} \right) \quad (4.4)$$

na qual,

$$S = \text{sgn}(1 - q\kappa_b) \quad (4.5)$$

$$Q = \sqrt{\left(\frac{dq}{ds}\right)^2 + (1 - q\kappa_b)} \quad (4.6)$$

Caso o deslocamento lateral do veículo q seja igual ao raio de curvatura $1/\kappa_b$, o caminho percorrido pelo veículo passa pelo centro de curvatura do espaço de trabalho. Se o deslocamento lateral q for maior que o raio de curvatura do espaço curvilíneo global, $1/\kappa_b$ a curvatura e a direção do caminho gerado terão sinais contrários. Os caminhos candidatos que pertencem a este caso são descartados, porque eles violam as restrições não holonômicas do veículo. Então, a geração de caminhos factíveis é garantida apenas quando o deslocamento lateral do caminho gerado é menor que a curvatura do espaço curvilíneo global. A curvatura dos caminhos está diretamente relacionada com a dinâmica do veículo, a qual pode ser descrita a partir de algum modelo veicular. Neste caso o modelo cinemático da bicicleta é utilizado descrito pela equação (G.1), na qual $\kappa = \tan(\delta_f)/L$ pela geometria de Ackerman (BARFOOT E CLARK, 2004). A curvatura do veículo é limitada pelo ângulo de esterçamento máximo (δ_{max}), então se $\kappa \geq \tan(\delta_{max})/L$ para o caminho candidato, este caminho é descartado.

De acordo com a equação (G.9), os movimentos do veículo estão sujeitos a apenas dois graus de liberdade, velocidade e esterçamento. A velocidade do veículo v_x pode ser representada em termos da distância de translação, como apresentado em (BARFOOT E CLARK, 2004) pela

equação 4.7.

$$v_x = SQ \frac{ds}{dt} \quad (4.7)$$

Substituindo a equação (4.7) dentro da equação (G.9) apresentada no anexo G.1 e transformando em termos da curvatura, as equações diferenciais que modelam os movimentos do veículo podem ser representadas em função do arco da distância da curva do espaço global pela equação (4.8).

$$\frac{dx_p}{ds} = Q \cos \theta_p \quad \frac{dy_p}{ds} = Q \sin \theta \quad \frac{d\theta}{ds} = Q\kappa \quad (4.8)$$

Esta abordagem permite gerar os caminhos independentemente da velocidade desenvolvida pelo veículo ao longo da curva do espaço de trabalho. Então, os caminhos podem ser determinados pela integração do modelo da equação(4.8) com respeito ao arco da distância s sobre a curva do espaço de trabalho.

4.3.2 Função matemática dos caminhos candidatos

Para gerar um caminho, a curvatura precisa ser determinada, o que é feito pelo deslocamento lateral q deste caminho sobre a curvatura da curva do espaço de trabalho. As derivadas primeira e segunda do deslocamento lateral também são necessárias para se calcular a curvatura do caminho e elas devem ser contínuas na longitude dos arcos a serem integradas. Então, a função de deslocamento lateral é projetada de modo a assegurar mudanças suaves no deslocamento lateral. A manobra para a mudança do deslocamento lateral no sistema de coordenadas curvilíneas é mostrado na figura 4.3. O deslocamento lateral q é descrito pelo polinômio cúbico da equação (4.9)(CHU *et al.*, 2012).

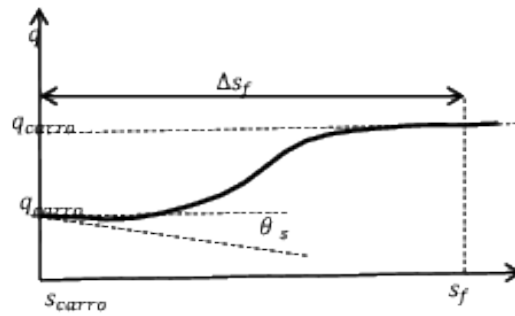


Figura 4.3: Função do caminho no espaço curvilíneo global. Adaptada de (CHU *et al.*, 2012)

$$\begin{aligned}
 q(s) &= \begin{cases} a\Delta s^3 + b\Delta s^2 + c\Delta s + q_{carro}, & s_i \leq s < s_f \\ q_f, & s_f \leq s \end{cases} \\
 \frac{dq}{ds}(s) &= \begin{cases} 3a\Delta s^2 + 2b\Delta s + c, & s_i \leq s < s_f \\ 0, & s_f \leq s \end{cases} \\
 \frac{d^2q}{ds^2}(s) &= \begin{cases} 6a\Delta s + 2b, & s_i \leq s < s_f \\ 0, & s_f \leq s \end{cases}
 \end{aligned} \tag{4.9}$$

Dado que $\Delta s = s - s_{carro}$

Na figura 4.3 o arco da distância s_{carro} e o deslocamento lateral q_{carro} são as coordenadas da posição atual do veículo no espaço curvilíneo global que são computados no estágio de localização. A diferença angular θ_s é definida como a diferença entre a orientação do veículo e o ângulo da reta tangente à curva do espaço de trabalho na posição atual. O arco da distância s_f e o deslocamento lateral q_f na posição final da mudança de deslocamento lateral são parâmetros de projeto para a geração de múltiplos caminhos. Desta forma são obtidas as condições de contorno da equação (4.10)

$$\begin{aligned}
 q(s_{carro}) &= q_{carro}, & \frac{dq}{ds}(s_{carro}) &= \tan \theta_s \\
 q(s_f) &= q_f, & \frac{dq}{ds}(s_f) &= 0
 \end{aligned} \tag{4.10}$$

Baseado nestas condições de contorno, os coeficientes do polinômio cúbico podem ser determinados analiticamente. O algoritmo de planejamento de caminhos gera um número finito de caminhos candidatos com diferentes deslocamento laterais q_f . O conjunto de todos os caminhos cobrem a estrada por completo cada qual com um diferente deslocamento lateral. As mudanças laterais no caminhos permitem ao veículo desviar de obstáculos. As manobras das mudanças

dos deslocamentos laterais são ajustadas pela especificação do arco da distância da curva do espaço de trabalho Δs_f como mostrado na figura 4.3. As distâncias da mudança do deslocamento lateral Δs_f e a velocidade do veículo v_x afetam significativamente a taxa da mudança do deslocamento lateral. Um pequeno Δs_f induz uma mudança degrau na mudança do deslocamento lateral dos caminhos gerados, enquanto que um Δs_f maior produz mudanças suaves no deslocamento lateral. Para mesmas distâncias de deslocamento lateral, velocidades maiores produziram maiores taxas de deslocamento lateral, o que aumenta a aceleração lateral. Acelerações laterais elevadas podem degradar a estabilidade do veículo e a sensação de conforto dos passageiros. Então, velocidades elevadas requerem distâncias maiores para as mudanças no deslocamento lateral. As distâncias das mudanças do deslocamento lateral Δs_f são determinadas em função da velocidade pela equação (4.11)

$$\Delta s_f = k_v v_x + \Delta s_{min} \quad (4.11)$$

na qual k_v é o ganho proporcional para a velocidade atual do veículo, e Δs_{min} é a distância mínima para as mudanças de deslocamento lateral. Os termos k_v e Δs_{min} são parâmetros de projeto heurísticos que devem ser escolhidos cuidadosamente de acordo com estilo de direção (esportiva ou urbana por exemplo) desejada e as limitações cinemáticas do sistema de esterçamento. Por exemplo, uma direção esportiva necessita de mudanças rápidas no deslocamento lateral, o qual é alcançado com pequenos valores de k_v . O intervalo de trabalho para o deslocamento lateral ($q_{max} - q_{min}$) foi selecionado para ser o comprimento da estrada, valor que depende principalmente da geometria da pista com uma resolução de Δq , o qual precisa ser selecionado do balanço entre o tempo computacional e a resolução do espaço solução, o qual melhora a convergência para uma solução ótima.

4.4 Seleção do caminho

O risco de colisão definido a partir da segurança, suavidade e consistência do caminho são parâmetros usados para selecionar os caminhos. Um algoritmo de busca localiza os caminhos que interceptam obstáculos descartando-os. Depois, o caminho que minimiza a combinação linear das funções de custos com fatores de peso é selecionado sendo descrita pela equação (4.12)

$$J[i] = w_s C_s[i] + w_m C_m[i] + w_\kappa C_\kappa[i] + w_c C_c[i] \quad (4.12)$$

na qual i é o índice do caminho, C_s é o custo de risco extrínseco sobre objetos estáticos, C_m é o custo de risco extrínseco sobre objetos em movimento, C_κ é o custo pelo risco intrínseco ou de suavidade e C_c é o custo pelo risco de consistência. Cada custo tem um fator de peso que determinam as características dos caminhos selecionados.

4.4.1 Custo por risco intrínseco

A suavidade do caminho é importante na condução autônoma, porque está associada ao risco interno do veículo. Em particular, caminhos não suaves aumentam o risco de acidentes porque possíveis perdas de controle do veículo associadas a uma dinâmica mais rápida aumenta os valores de indicadores de risco como transferência de carga lateral (GLASER *et al.*; KAMNIK *et al.*, 2010; 2003), velocidade excessiva (GLASER *et al.*, 2010), invasão de faixa e perda de contato entre a roda e o asfalto (MULLER *et al.*; AHN *et al.*; LECHNER *et al.*; LI *et al.*; GHANDOUR *et al.*, 2003; 2009; 2006; 2006; 2011).

A integração do quadrado da curvatura ao longo da extensão do caminho (DANIEL *et al.*, 2010) (CHU *et al.*, 2012) (HORN, 1983) foi escolhido como critério de suavidade do caminho de acordo a equação (4.13)

$$C_\kappa[i] = \int \kappa_i^2 ds_p = \int \kappa_i^2(s) Q(s) ds \quad (4.13)$$

na qual s_p é o comprimento ao longo do caminho, s é o arco da distância da curva do espaço de trabalho, e κ_i é a curvatura do caminho avaliado i .

4.4.2 Custo pelo risco de consistência

Os custo de segurança e suavidade dos caminhos são calculados a partir das informações do ambiente que são coletadas a cada intervalo de tempo. Por esta razão, o custo de segurança e suavidade não previnem que os caminhos gerados não sejam significativamente diferentes do caminho anterior, o que pode acrescentar a quantidade de energia requerida pelo veículo, assim como o risco interno. Mudanças abruptas entre caminhos causam excessivo esforço de controle e desperdício de energia. Para se prevenir mudanças significativas entre a geração de caminhos candidatos a consistência dos caminhos gerados é avaliada.

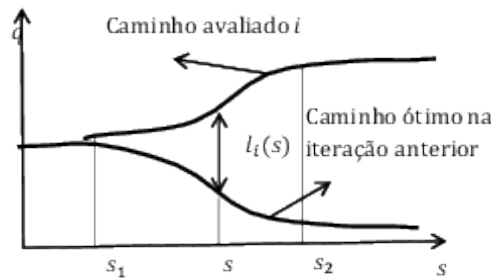


Figura 4.4: Representação da função de custo pela consistência do caminho . Adaptada de (CHU *et al.*, 2012)

O conceito de consistência significa que no estado atual, ações ou decisões sobre um sistema qualquer dependeram dos estados, ações ou decisões anteriores (CHU *et al.*, 2012). Assim, para a geração de caminhos, a seleção do caminho do grupo de candidatos deve ser influenciada pelas características do caminho anterior. Uma forma de se quantificar a similaridade entre o caminho anterior e os candidatos atuais é a integração da distância $l_i(s)$ entre os pontos do caminho atual sobre o caminho anterior ao longo da região que sobrepõem. O custo de consistência dos candidatos é calculado pela equação (4.14)

$$C_c[i] = \frac{1}{\Delta s_f} \int_{s_1}^{s_2} l_i ds \quad (4.14)$$

na qual l_i é a distância euclidiana entre o ponto sobre o caminho atual de índice i e um ponto sobre o caminho do passo anterior com o mesmo arco da distância sobre a curva do espaço global. Uma explicação do cálculo é mostrada na figura 4.4.

4.4.3 Custo pelo risco extrínseco sobre objetos estáticos

A função de colisão fornece apenas informações binárias, porque ela vai descartar os caminhos com obstáculos. No entanto, o risco de colisão é necessário para cada caminho candidato. O risco de cada caminho é quantitativamente avaliado por meio de uma convolução entre as funções de colisão binária $c[i]$ no espaço discreto dos caminhos i e uma função de probabilidade com distribuição gaussiana $g[i]$ (CHU *et al.*, 2012). Para inferir o risco do caminho, a seguinte regra é utilizada: O risco de colisão de um caminho afeta o risco de colisão dos caminhos vizinhos e é insignificante ao risco de colisão dos caminhos mais distantes. Assim, o risco de um caminho é definido pela equação (4.15).

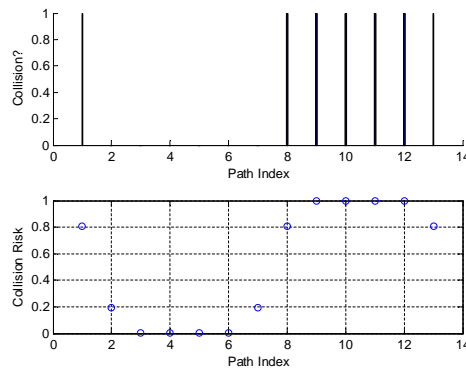


Figura 4.5: Risco de colisão dos caminhos candidatos. Adaptada de (CHU *et al.*, 2012)

$$C_s[i] = \sum_{k=0}^N c[k]g[i - k] \quad (4.15)$$

na qual,

$$g[i] = \frac{1}{\sqrt{2\pi}\sigma_s} \exp\left(-\frac{(\Delta q \cdot i)^2}{2\sigma_s^2}\right) \quad (4.16)$$

e, $c[i]$ é o parâmetro de detecção de colisão do caminho de índice i , Δq é o resolução do deslocamento lateral para o caminho, e σ_s é o desvio padrão do risco de colisão. No caso da convolução para um caminho fora do índice, se o valor do parâmetro de detecção de colisão é mantido sempre como verdadeiro o veículo tentará navegar pelo centro da via para não sair da estrada. O desvio padrão do kernel da gaussiana $g[i]$ é um parâmetro heurístico que está relacionado com o intervalo efetivo do processo de detecção de colisão para cada caminho que de acordo com o tipo de sensor e o método de detecção gerará um fator de segurança ao redor do objeto detetado. A figura 4.5 na parte inferior apresenta um exemplo de convolução entre as variáveis de colisão binárias dos caminhos mostrado na parte superior com uma distribuição Gaussiana para o risco de colisão do caminho afetando o risco dos caminhos vizinhos. Pode-se ver que os caminhos 2 a 7 não se prolongam em direção a obstáculos mas sua probabilidade de acidente não é nula porque ele se encontra perto de caminhos que colidem.

4.4.4 Custo pelo risco extrínseco sobre objeto em movimento

O risco de colisão é dependente da variação das variáveis cinemáticas do modelo como velocidade, posição e orientação, portanto diferentes parâmetros devem ser utilizados para com-

putar o custo sobre objetos em movimento para determinar se o veículo consegue ultrapassar o objeto em movimento. De acordo com essa seleção o custo pode querer ser minimizado ou maximizado. O critério adotado para identificar o risco máximo foi definido como identificar em quais $q(\Delta s_f)$ estão localizados os objetos em movimento, aproximando sua posição com um estimador de estados, para encontrar quais caminhos candidatos se interceptam à trajetória do objeto em movimento, $c_m[i]$. Depois de encontrados os caminhos que se encontram com o objeto em movimento é avaliada o risco de todos os caminhos pela convolução discreta descrita pela equação (4.17) (Note-se que a equação(4.15) tem a mesma forma mas os significados dos parâmetros são diferentes)

$$C_m[i] = \sum_{k=0}^N g_m[k]c_m[i - k] \quad (4.17)$$

dado que,

$$g_m[i] = \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left(-\frac{(\Delta q.i)^2}{2\sigma_m^2}\right) \quad (4.18)$$

$c_m[i]$ é o objeto detectado que é interceptado pelo caminho de índice i , Δq é a resolução do deslocamento lateral do caminho e σ_m é o desvio padrão do risco de colisão. O desvio padrão do kernel da Gaussiana $g_m[i]$ é um parâmetro projetado que depende da magnitude da covariância da velocidade estimada do objeto em movimento e da velocidade do objeto em movimento, dado que a alta velocidade a capacidade para trocar de direção é elevada.

A velocidade do objeto em movimento multiplicado por um fator de segurança (k_o) e que o veículo autônomo consiga alcançar a velocidade máxima em uma distância menor a Δs_f determinará se é melhor o veículo continuar no caminho ou não. Finalmente dado que a via pode ser mão dupla ou estar em uma estrada com curvas, um estado discreto g_{global} é definido. Se esta condição for violada o caminho será eliminado (o carro passará o objeto em movimento) ou entrará em estado de emergência.

Note-se que C_m é um risco a minimizar se o veículo decide adiantar o objeto em movimento, em caso contrario vai ser $C_m[i] = 1 - C_m[i]$. O critério utilizado para determinar a troca de faixa é determinado a partir de que a velocidade permitida do caminho seja maior em uma porcentagem à velocidade estimada do objeto em movimento, assim como que a estrada o permita por meio de um estado g_{global} calculado de acordo à informação dos pontos globais na longitude de arco Δs_f e a curvatura nesse intervalo. Os estados projetados são cinco: estrada

mão única, estrada mão dupla, interseção no horizonte, rotatória, virar a direita e virar a esquerda. Os dois últimos são obtidos se a integração da curvatura k_b no arco da distância Δs_f é maior que um valor limite θ_{turn} . Note-se que se a distância avaliada é muito grande pela velocidade do veículo, os estados virar a esquerda ou a direita devem ser avaliados com o primeiro valor máximo absoluto que cumpra a condição, pois a estrada pode estar em uma condição de dupla curva. O algoritmo é apresentado no algoritmo 2.

```

for  $i \leftarrow 1$  to  $N$  do
  if  $v_{path} < k_o.v_o$  &  $g_{global}$  é permitido &  $\Delta s_{ff} > \frac{v_{path}^2 - v_x^2}{2 \cdot acc_{long}}$  then
     $C_m \leftarrow 1 - C_m$ ;
    descartar caminhos sem obstaculos
  else
    descartar caminhos com obstaculos
  end
end

```

Algoritmo 2: Risco de seguridade ante objetos em movimento

A distância mínima para quando o veículo está seguindo um objeto em movimento é calculada achando que o objeto em movimento vai a máxima velocidade permitida do caminho (v_{path}) e para abruptamente, portanto o veículo autônomo desacelera longitudinalmente Dec_{long} . O anterior é definido pela equação (4.19).

$$d_{path} < \frac{v_{path}^2}{2 \cdot Dec_{long}} \quad (4.19)$$

4.5 Perfil de velocidade

Para se alcançar alto desempenho de controle e uma condução segura, a análise da velocidade do veículo é imprescindível. A velocidade desempenha um importante papel na tarefa de seguir trajetórias, porque o carro não pode sempre transladar na velocidade máxima permitida v_{path} que seria o desejado para chegar ao destino o mais rápido permitido. O perfil de velocidade é gerado considerando-se a curvatura do caminho, as acelerações e desacelerações longitudinais possíveis do veículo e um fator de segurança que avalia a proximidade do obstáculos e possíveis condições de emergência.

4.5.1 Condições de risco intrínseco

Condições de capotamento, derrapagem e também conforto dos passageiros estão correlacionados intimamente com a aceleração lateral desenvolvida pelo carro (KAMNIK *et al.*, 2003). A aceleração lateral pode ser aproximada pela aceleração centrípeta do modelo cinemático da bicicleta adotado e pode ser calculada considerando-se velocidade constante a partir de:

$$acc_{lateral} = \kappa v_{car}^2$$

Assim, a velocidade que o veículo poderá desenvolver ao longo de um caminho não só será limitada pela velocidade máxima permitida da estrada mas também pela aceleração lateral máxima desejada (CHU *et al.*; LI *et al.*, 2012; 2014).

$$v_{\kappa} = \sqrt{\frac{|Acc_{lateral}|_{max}}{max(\kappa(s))}}$$

4.5.2 Condições de risco extrínsecos por obstáculos

Quando o veículo atravessa passagens estreitas um motorista real instintivamente reduziria a velocidade do veículo. Assim mesmo precisa acontecer com o veículo autônomo quando está atravessando por passagens estreitas. Uma medida do risco é determinada pela função de custo de risco intrínseco por objetos estáticos, C_s para o caminho selecionado.

$$v_{safe} = (1 - k_{safe} C_s^2) v_k$$

na qual k_{safe} é o ganho de segurança para o ajuste da velocidade. Ele é um parâmetro heurístico que deve ser experimentalmente escolhido de acordo com a estilo de direção desejado.

Note-se que se C_m é zero a velocidade selecionada deve ser o mínimo entre v_o e v_{safe} dado que o carro vai em modo comboio com o objeto em movimento a frente. No caso em que C_m não seja zero e $\frac{k_o}{\Delta s_f} \cdot v_o$ seja maior que v_{safe} o caminho selecionado deve voltar a ser selecionado fazendo $v_{path} = v_{safe}$.

4.5.3 Acelerações longitudinais

As mudanças no perfil de velocidade serão limitadas pelas aceleração longitudinal Acc_{long} e desaceleração longitudinal Dec_{long} possíveis de serem desenvolvidas pelos atuadores e pela sensação de conforto dos passageiros. Considerando aceleração constante a partir do modelo adotado pode-se determinar a velocidade sobre o arco da distância pela equação (4.20) (LI *et al.*, 2014).

$$v_{min} < v(s_p) \leq \sqrt{v^2(s_0) + 2Acc_{long}s_p} \quad (4.20)$$

na qual,

$$v_{min} = \begin{cases} \sqrt{v^2(s_0) + 2Dec_{long}s_p} & , v^2(s_0) + 2Dec_{long}s_p \\ 0, & casocontrario \end{cases}$$

$$s_p = sQ(s)$$

$v(s_0)$ é a velocidade atual do veículo e $v(s_p)$ estará limitada a v_{safe} .

4.6 Simulação do planejamento de trajetórias

Para validar o algoritmo de trajetórias implementado foi desenvolvida uma simulação em Matlab, onde o ambiente é um espaço bidimensional o qual é detectado através de um sensor proprioceptivo de tipo LIDAR e sensores da posição e orientação do veículo ideais. O ambiente simulado é uma estrada de dupla curva, tendo cada curva um raio horizontal de $50m$, que concorda com o raio mínimo permitido em uma estrada CLASSE III de regiões onduladas no qual pode-se ter uma velocidade máxima de até $40km/h$ (DOS TRANSPORTES, 1973). Na figura 4.6 é apresentado uma estrada de $315m$ que vai ser utilizada na simulação. Na figura 4.7b é apresentada a curvatura de referência em preto onde idealmente são estradas compostas por círculos e linhas, de onde é obtida uma série de pontos de referência globais, com os quais é conformado o conjunto de *splines* que definem a trajetória global. Note-se que o espaço curvilíneo para poder ser contínuo em sua segunda derivada apresenta algumas diferenças com a estrada ideal como pode ser visto na figura 4.7a. Nesta mesma figura é possível ver o mapa que o LIDAR consegue

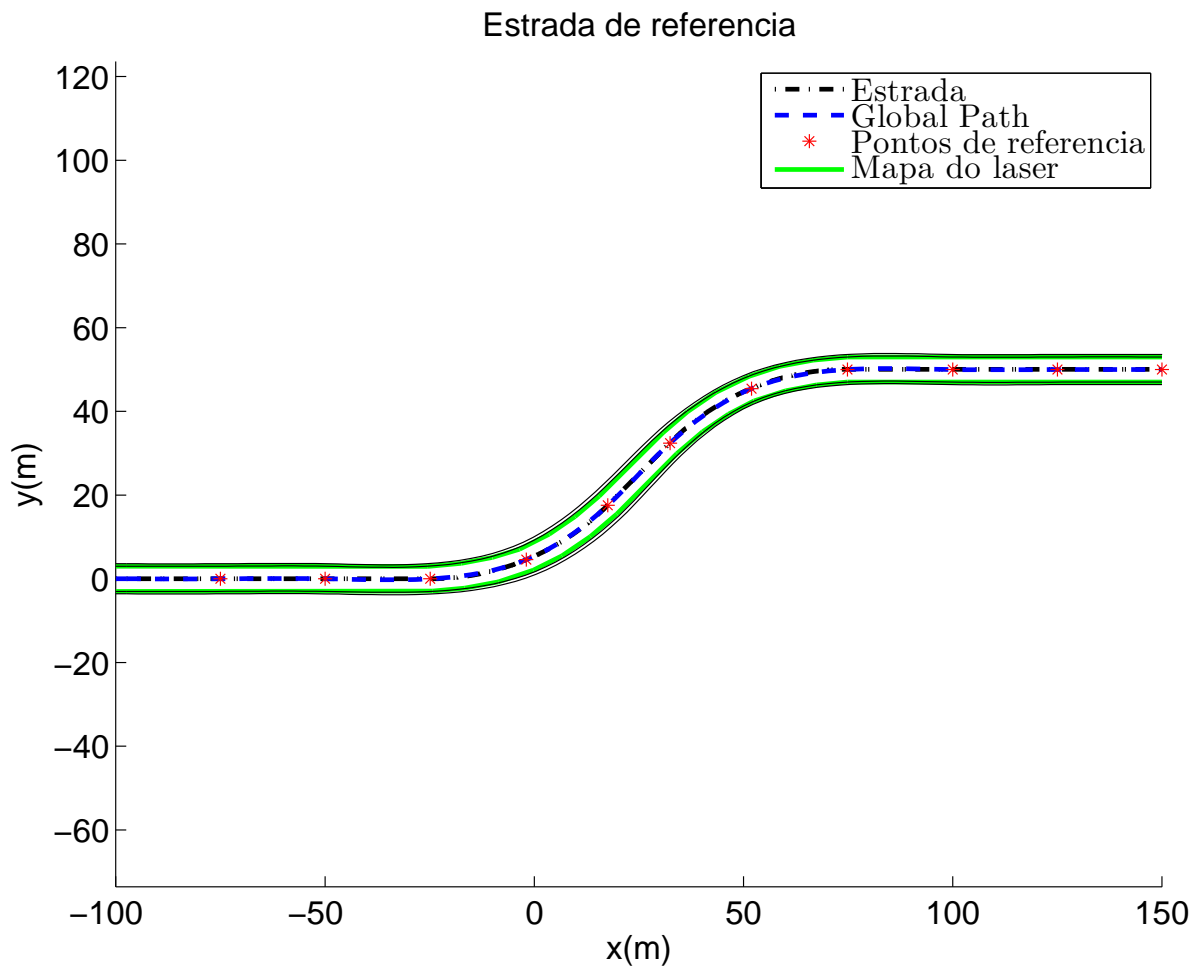


Figura 4.6: Estrada de referencia para a simulação

detetar, dado que o algoritmo só utiliza linhas retas como mapa.

A seguir serão apresentados os resultados da simulação divididos em quatro seções: Simulação do LIDAR, detecção de obstáculos fixos e em movimento, geração de trajetórias e seleção de trajetórias.

4.6.1 Simulação do LIDAR

Para a simulação do laser foi feito um mapa composto por linhas que representam os extremos das estradas e os obstáculos. Igualmente o sensor laser simulado é um SICK LMS291 que tem um ângulo de abertura de 100° e uma resolução de $0,5^\circ$ e um alcance máximo de $70m$. O funcionamento do sensor consiste em encontrar a distância (l_i) entre o ponto $P(x,y)$ da linha k do mapa e o laser para o ângulo ϕ_i .

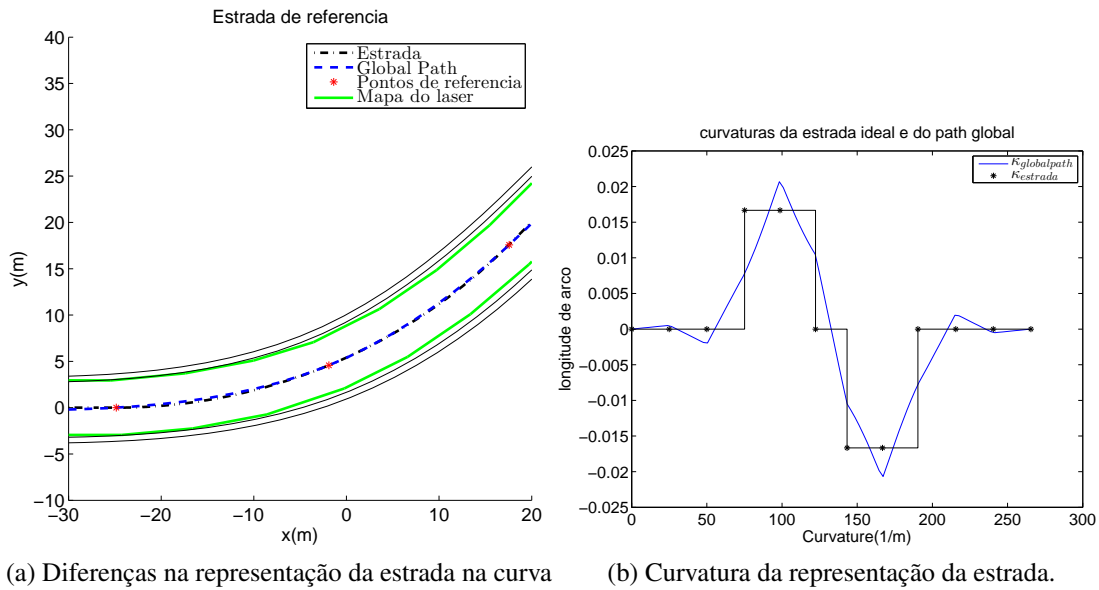


Figura 4.7: Estrada de referência para os testes

Portanto, se encontramos o ponto de interseção entre a linha do laser i no ângulo ϕ_i e a linha do mapa k temos as equações (4.21) e (4.22) as quais formam um sistema de duas equações e duas incógnitas, na quais l_{ik} é o resultado do laser e $l_2 < r$ e r é a magnitude da linha k . Note-se que a posição real do veículo é conhecida e a posição do laser com respeito a ele também.

$$P_x = x_k + l_2 \cos(\beta) = x_{laser} + l_{ik} \cos(\phi_i) \quad (4.21)$$

$$P_y = y_k + l_2 \sin(\beta) = y_{laser} + l_{ik} \sin(\phi_i) \quad (4.22)$$

Finalmente o menor valor de l_{ik} para o mesmo ângulo ϕ_i é selecionado, $l_i = l_{ik}^{mínimo}$. Depois de recorrer todas as linhas se l_i é menor que L_{max} então é uma medida válida. Finalmente para simular os objetos em movimento a geometria deles são inseridas através de linhas no mapa de acordo a sua posição real no instante de tempo antes de sensoriamento do mapa. Na figura 4.9 é apresentado o resultado no espaço bidimensional obtido pelo laser (linhas brancas pontilhadas). Na figura 4.10 é apresentado o resultado do laser no sistema de coordenadas local, que significa que para cada ângulo de saída vai gerar uma distância ao obstáculo.

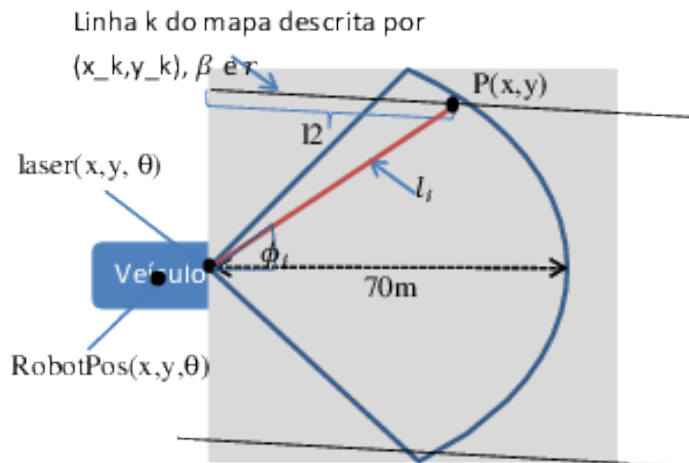


Figura 4.8: Simulação do sensor LIDAR através de um mapa de linhas

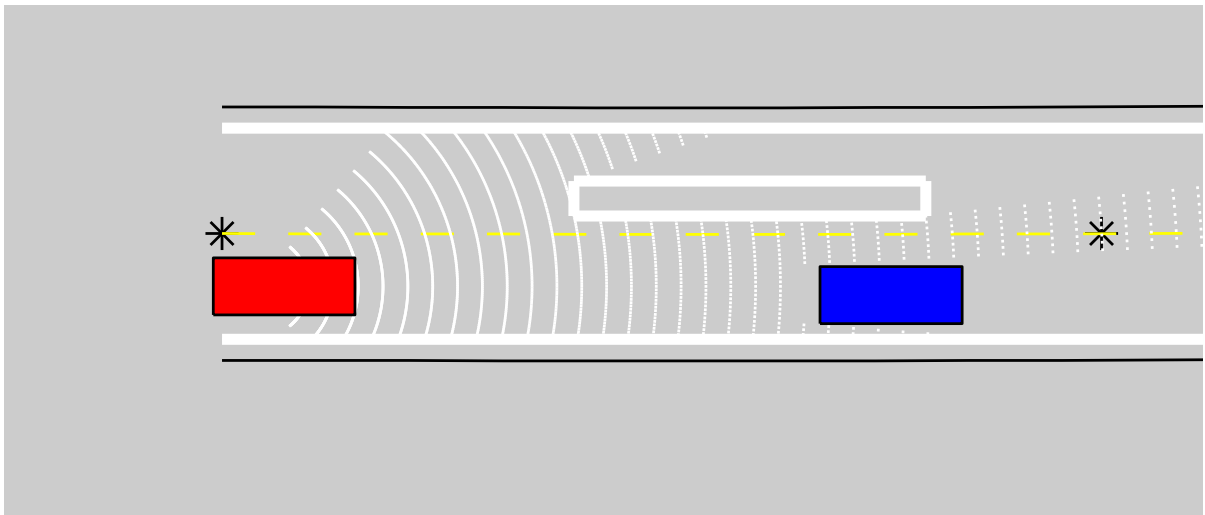


Figura 4.9: Resultado de uma amostra do LIDAR no ambiente de simulação

4.6.2 Detecção de objetos

Para a detecção de objetos é realizado o processo contrário da simulação do laser, são procuradas linhas com o resultado do sensor. A figura 4.11 apresenta o resultado durante quatro iterações, onde em vermelho estão as linhas achadas e em preto a linha identificada como objeto tipo carro durante varias iterações. O algoritmo para a detecção das linhas é descrito pelo algoritmo 3, no qual é assumido que a posição real do veículo é conhecida, no entanto ela vai ser uma aproximação gerada por um algoritmo de localização global do veículo.

Para a detecção do objeto tipo carro nas linhas é procurado que a longitude da linha seja similar as dimensões de um veículo e que a orientação β seja perpendicular com uma tolerância

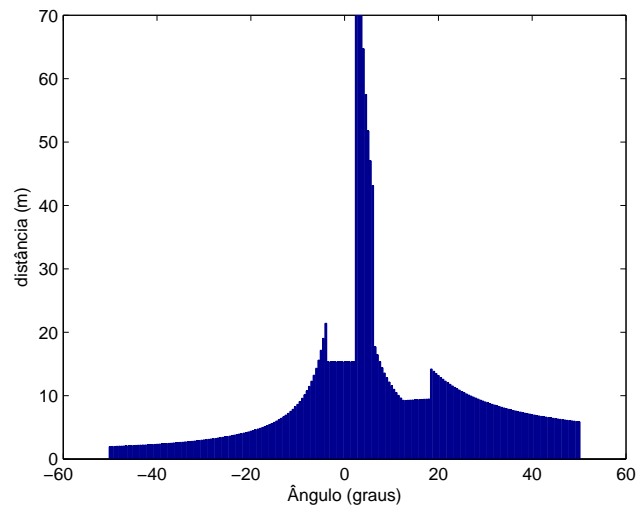


Figura 4.10: Resultado de uma amostra do LIDAR com respeito ao ângulo de saída ϕ_i

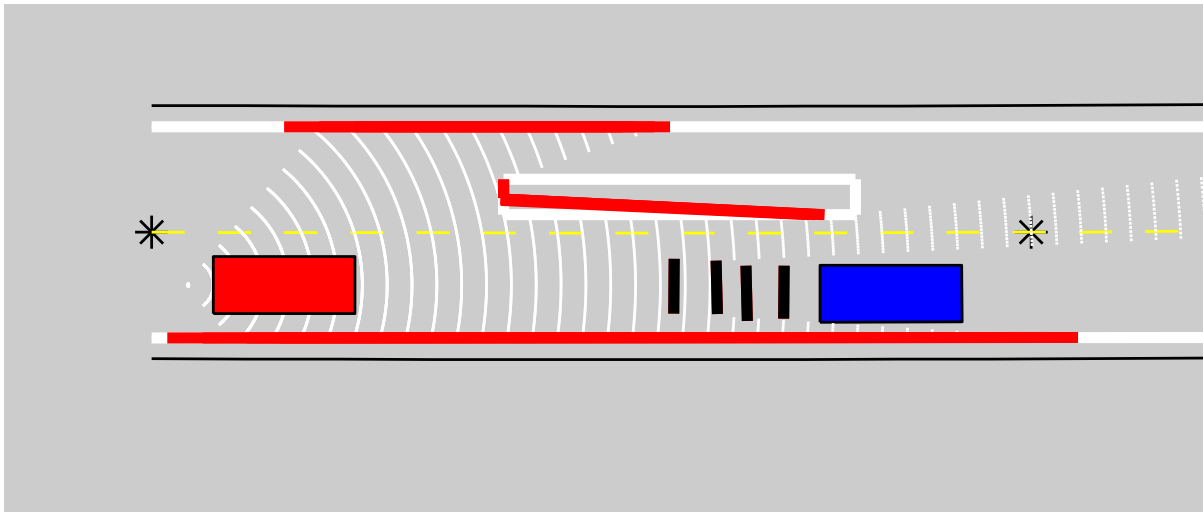


Figura 4.11: Resultado da detecção de linhas do LIDAR no ambiente de simulação

ao ângulo da estrada ou no caso de já ser detectado o objeto em iterações anteriores com o ângulo $\theta_{car2}(k-1)$, e a distância entre a estimação da posição do objeto e o centro da linha, portanto $\|pos_{car2} + v_{car2}T_s - pos_{linha}\| < r$, na qual r é um valor em metros.

Portanto uma vez obtida a linha que representa o objeto em movimento (preta na figura 4.11), seu ponto médio e o ângulo perpendicular, obtém-se a localização do objeto em movimento. Calculados estes valores, assume-se que os objetos detectados são estáticos, no entanto para atribuir uma velocidade aos objetos em movimento candidatos, uma estimação cinemática por um filtro de Kalman é realizada. Após algumas amostragens alguns objetos serão estimados em movimento. O modelo cinemático do objeto em movimento baseia-se nas equações de estado (G.9), tendo em conta que será limitado a veículos, no entanto a entrada de esterçamento

```

Result: lines
 $x_0 \leftarrow x_{carro} + l[1] \cos(\phi[1]);$ 
 $y_0 \leftarrow y_{carro} + l[1] \sin(\phi[1]);$ 
 $\beta_0 \leftarrow atan2(y_f - y_0, x_f - x_0);$ 
for  $i \leftarrow 2$  to  $N$  do
     $x_i \leftarrow x_{carro} + l[i] \cos(\phi[i]);$ 
     $y_i \leftarrow y_{carro} + l[i] \sin(\phi[i]);$ 
     $\beta_i \leftarrow atan2(y_i - y_0, x_i - x_0);$ 
    if  $\beta_i - \beta_0 > \delta_\perp$  then
         $lines.append(x_0, y_0, x_{i-1}, y_{i-1}, \beta_0);$ 
         $x_0 \leftarrow x;$ 
         $y_0 \leftarrow y;$ 
         $\beta_0 \leftarrow \beta_i;$ 
         $k \leftarrow 0$ 
    else
         $k ++;$ 
         $\beta_0 \leftarrow mean(\beta_{i-k:i})$ 
    end
end

```

Algoritmo 3: Detecção de linhas pelo LIDAR

será substituída pela curvatura de acordo ao modelo cinemático do veículo e a velocidade de entrada será considerada uma variável de estado. Portanto, o modelo de estado é descrito pela equação (4.23).

$$\dot{\mathbf{x}}_{car2} = \begin{bmatrix} \dot{x}_{car2} \\ \dot{y}_{car2} \\ \dot{\theta}_{car2} \\ \dot{v}_{car2} \end{bmatrix} = \begin{bmatrix} v_{car2} \cos \theta_{car2} \\ v_{car2} \sin \theta_{car2} \\ v(\kappa_v + w_\kappa) \\ w_v \end{bmatrix} \quad (4.23)$$

na qual \mathbf{x}_{car2} são as variáveis de estado, w_κ é a variação da curvatura de distribuição normal com média zero e variância σ_κ e w_v é a aceleração longitudinal aleatória de distribuição normal com média zero e variância σ_v . (4.23) é discretizada usando Euler é aplicado o Filtro Estendido de Kalman (ver seção 3.3) com as medições encontras pelo sensor LIDAR descritas pela equação (4.24).

$$\mathbf{z}_k = \begin{bmatrix} x_{car2} \\ y_{car2} \\ \theta_{car2} \end{bmatrix} = H\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{x} \quad (4.24)$$

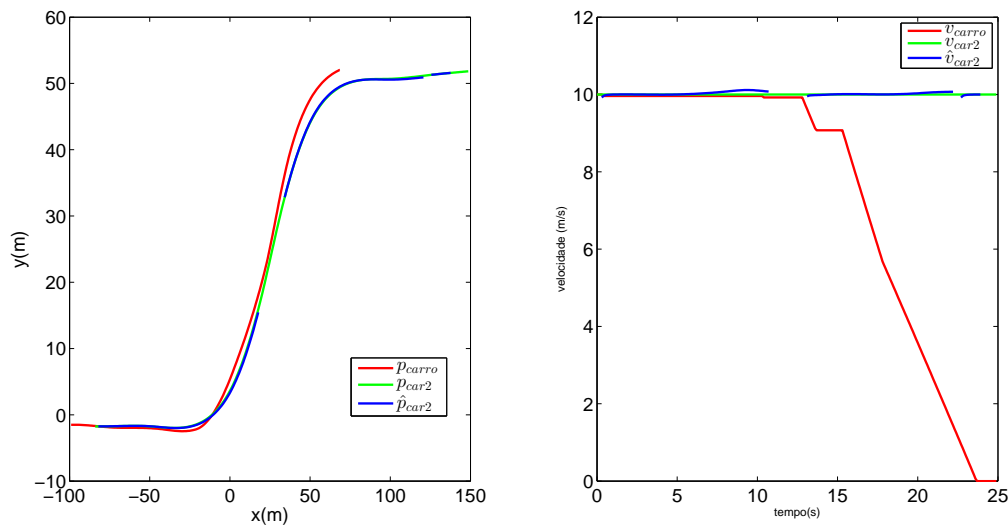


Figura 4.12: Resultado da detecção das variáveis de estado do objeto em movimento

Na figura 4.12 é mostrado a detecção do objeto em movimento durante a simulação, com respeito a posição e a velocidade do carro, para um objeto em movimento a velocidade constante que segue a curvatura da estrada. Note-se que dada a orientação do veículo as vezes não é possível detetar o objeto em movimento (na figura não é mostrado o valor estimado). Nesse caso o estimador continua funcionando até que o número de iterações sem detecção seja superior a um valor limite estimado heurísticamente.

4.6.3 Geração de caminhos candidatos

O gerador de caminhos sempre leva o veículo na direção do espaço global, dado que as condições finais do caminho é seguir com valor constante q_f a trajetória global, no entanto a seleção de seus parâmetros e das condições iniciais do veículo geram trajetórias com diferentes características. A figura 4.13a apresenta como os caminhos gerados seguem a curva do caminho global só com a equação polinomial e analítica de $q(s)$ o qual gera vantagens sobre outros métodos testados (LI *et al.*, 2014) (GARCIA *et al.*, 2014). A figura 4.13b mostra como se comportam os caminhos gerados quando o veículo tem um ângulo de orientação diferente da trajetória, fazendo com que ele alcance o ângulo da trajetória de uma forma suave. As figuras 4.13c e 4.13d apresentam a forma geométrica da curva para diferentes valores de Δs_f , onde percebe-se porque o parâmetro deve depender da velocidade do veículo dado que a velocidades altas deve ser muito mais suave o deslocamento lateral. Note-se que diferentes valores de q_i trocam o ponto inicial e portanto geram caminhos candidatos diferentes.

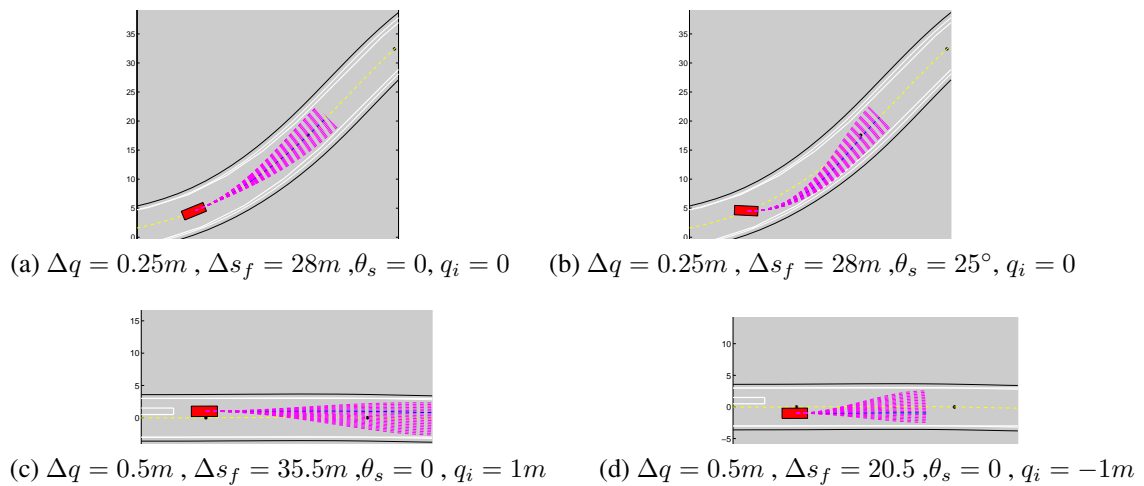


Figura 4.13: Caminhos candidatos para diferentes condições iniciais

4.6.4 Seleção de trajetórias

Na seleção de trajetórias quatro funções de custo são calculadas para determinar o melhor caminho. Em um primeira simulação, quando não temos objetos em movimento, na figura 4.14 as trajetórias realizadas com uma só função de custo são apresentadas. Nestas simulações o veículo inicia em uma posição $s = 0$, $q = -1$ e $\theta_s = 5^\circ$ com uma velocidade $v = 10m/s$ e o planejamento local feito a $2hz$. Quando só a consistência do caminho é calculada, é assumido que o carro inicialmente quer seguir o caminho global, portanto como pode-se ver na figura 4.16a em uma primeira instância os custos são altos mas depois como ele consegue seguir o caminho global, o custo é zero. Quando só a curvatura é utilizada como critério de seleção, a função de custos vai conseguir ser zero nas retas, mas nas seções curvas da estrada ela vai ser o menor valor possível como é mostrado na figura 4.16b, como pode-se ver na figura 4.14 onde para minimizar a curvatura o veículo se desloca nos extremos das estradas nas curvas.

Finalmente se só a função de obstáculos é utilizada como é mostrado na figura 4.16c, quando existem objetos na estrada a função de custos aumenta e como pode-se ver na figura 4.15, na primeira iteração o veículo projeta ficar mais longe do obstáculo através de sua trajetória. Como nenhuma outra restrição existe quando não existem objetos na rua ele vai ficar no centro dado que sair da rua é considerado um obstáculo. Quando todas as funções de custo são calculadas e ponderadas, o valor da função minimizada é apresentada na figura 4.16d, na qual $w_s = 5$, $w_\kappa = 0.2$ e $w_c = 1$. Note-se que quando se está na curva e existe um objeto, existe risco intrínseco e extrínseco o qual gera uma posição de perigo para o veículo.

O segundo cenário consiste em simular o que acontece com as funções de custo quando

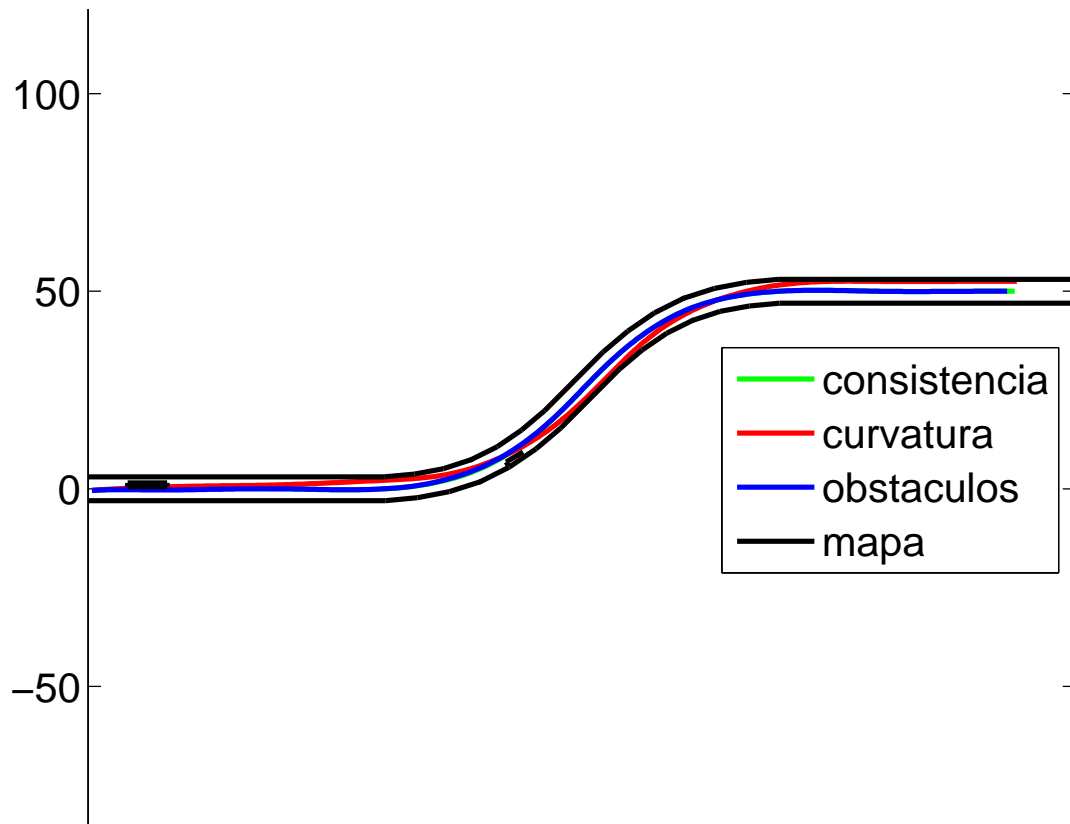


Figura 4.14: Seleção de trajetória para diferentes funções de custo

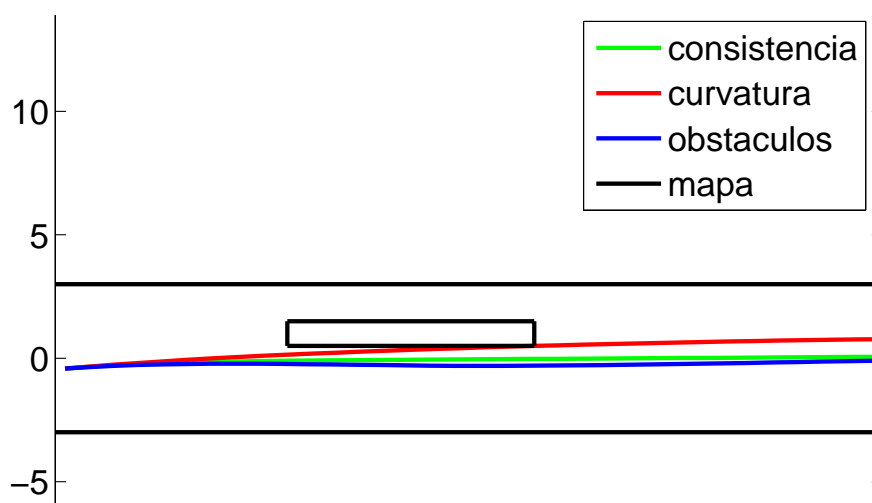
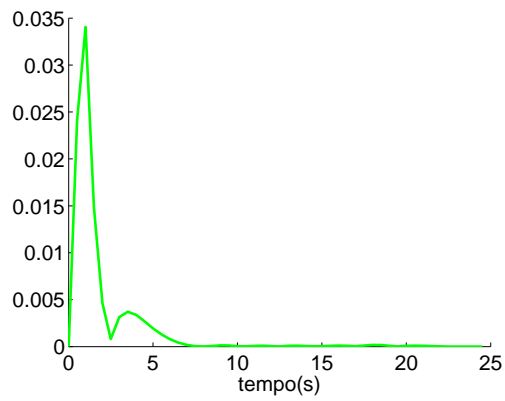
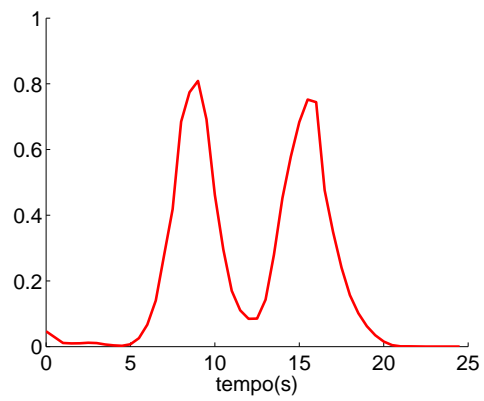


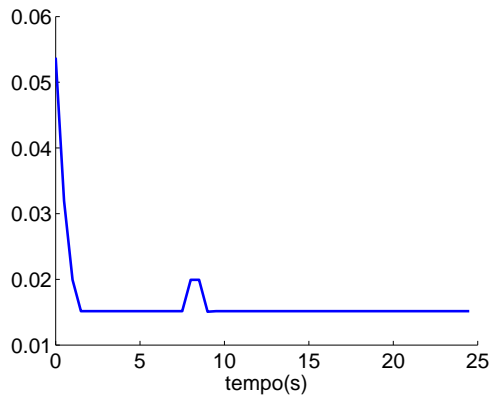
Figura 4.15: Seleção de trajetória ante diferentes funções de custo na parte inicial da simulação



(a) Função de custo de consistência



(b) Função de custo da curvatura



(c) Função de custo de obstáculos estáticos

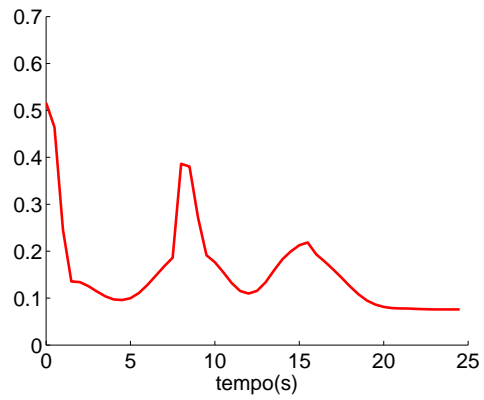
(d) Função de custo combinada $w_s = 5$, $w_\kappa = 0.2$, $w_c = 1$

Figura 4.16: Função de custo para diferentes critérios

existe um objeto em movimento na frente do veículo. Neste caso será um carro (*car2*) com uma posição e uma velocidade estimada. Na figura 4.17 são mostradas as trajetórias que seguiram os veículos durante o caminho global de referência. A simulação é realizada segundo as seguintes condições iniciais:

- Variáveis de velocidade. $v_x = v_{path} = v_{car2} = 10m/s$, $acc_{long} = 1m/s^2$
- Variáveis do planejamento. $\Delta s_f = 28m$, $\Delta q = 0.25m$, $\sigma_m = 0.3$, $\sigma_s = 0.8$, $w_s = 5$, $w_\kappa = 0.2$, $w_c = 1$ e $w_m = 2$, $k_{safe} = 0.5$
- Parâmetros do estimador do objeto em movimento. Dos sensores $\sigma_{angulo} = 5^\circ$ e $\sigma_x = \sigma_y = 0.1m$ e das entradas $\sigma_\kappa = 0.005$ e $\sigma_v = 0.5m/s$. Note-se que a matriz de covariâncias é iniciada com uma grande incerteza na velocidade do objeto em movimento. Os valores de tolerância na detecção da linha que representa o objeto são: $\beta = 30^\circ$, $r = 0.2m$
- Posição inicial do veículo autônomo no espaço curvilíneo global. $q(s_i) = -0,5$, $s_i = 0$, $\dot{q}(s_i) = 0$
- Posição do objeto em movimento no espaço curvilíneo global. $q(s_{car2}) = -0,75$, $s_{car2} = 15$, $\dot{q}(s_{car2}) = 0$.
- O período de amostragem da estimação da posição do objeto em movimento é de $0,1s$, o do planejamento local a cada $2,5s$ e das entradas da simulação cinemática dos veículos é de $0,01s$, dado que os modelos cinemáticos dos veículos são feita em tempo contínuo com ODE45.

A figura 4.18 apresenta em linhas azuis as distâncias do laser aos extremos do objeto em movimento, em magenta os possíveis caminhos que pode realizar o veículo autônomo e em verde o caminho estimado do objeto em movimento considerando que o veículo vai a velocidade constante. Anteriormente à seleção destes possíveis caminhos, as funções de custo das trajetórias são descritas pela figura 4.19. As barras azuis representam que esses caminhos tem objetos estáticos portanto devem ser descartados, assim como a linha contínua vermelha representa em que caminhos se vai cruzar com o objeto em movimento. Adicionalmente podemos ver que como o carro está em $q = -0,5$ o custo de consistência nesse caminho é zero e que não ocorre o mesmo com a curvatura que tem seu valor mínimo perto de $q = 0$, dado que o caminho global não é exatamente uma reta.

A figura 4.20 apresenta como fica o custo ponderado, quando o veículo autônomo não consegue ultrapassar o objeto em movimento. Note-se que a curva de custo do objeto em movimento foi invertida, porque se quer maximizar o caminho que segue o veículo.

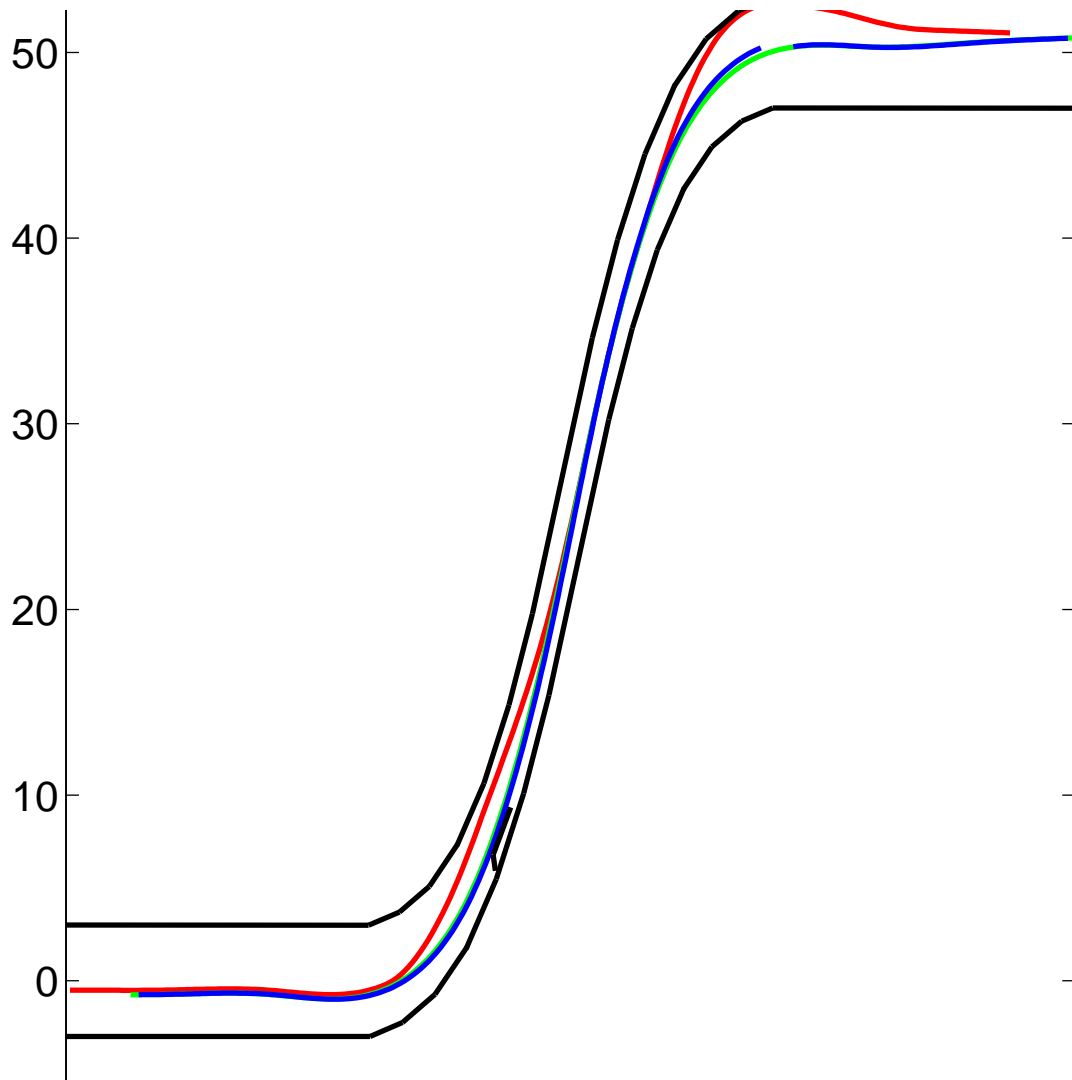


Figura 4.17: Trajetória ante um objeto em movimento na mesma velocidade

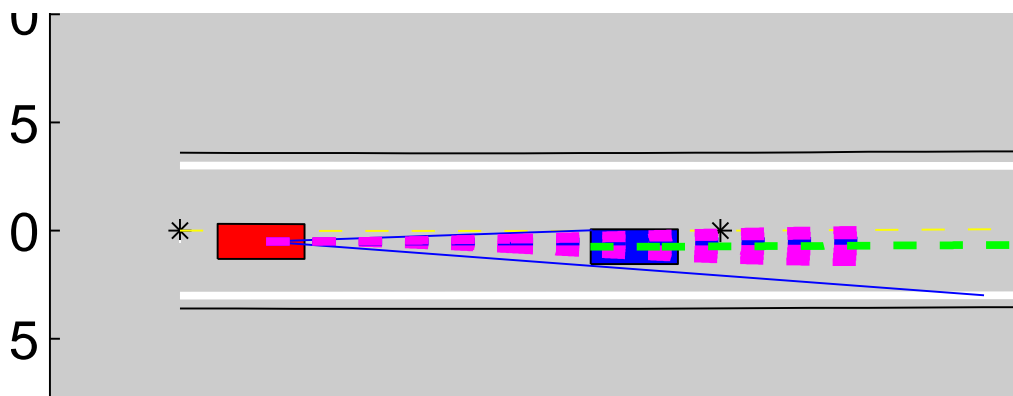


Figura 4.18: Rotas possíveis para seguir o objeto em movimento

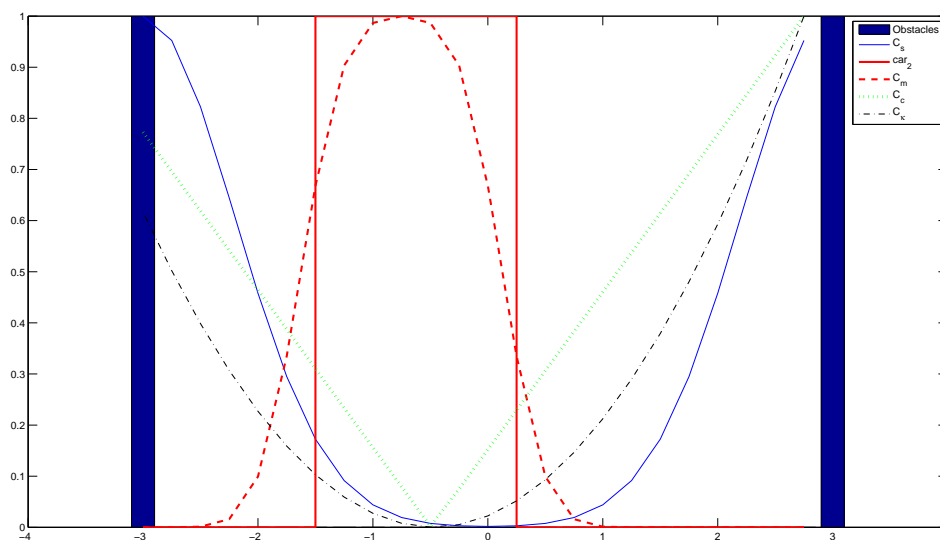


Figura 4.19: Valores do custo de cada um dos caminhos com um objeto em movimento na frente

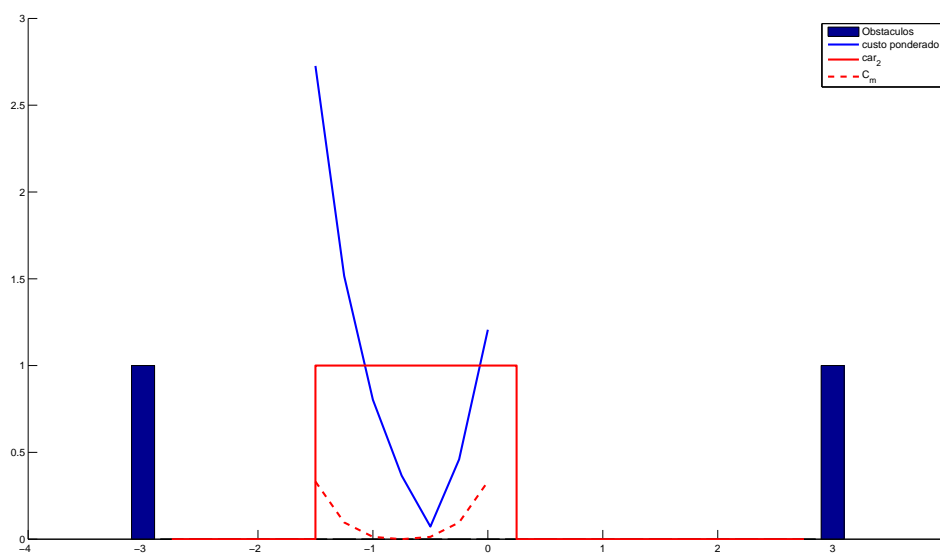


Figura 4.20: Soma ponderada do custo dos caminhos possíveis para seguir um objeto em movimento

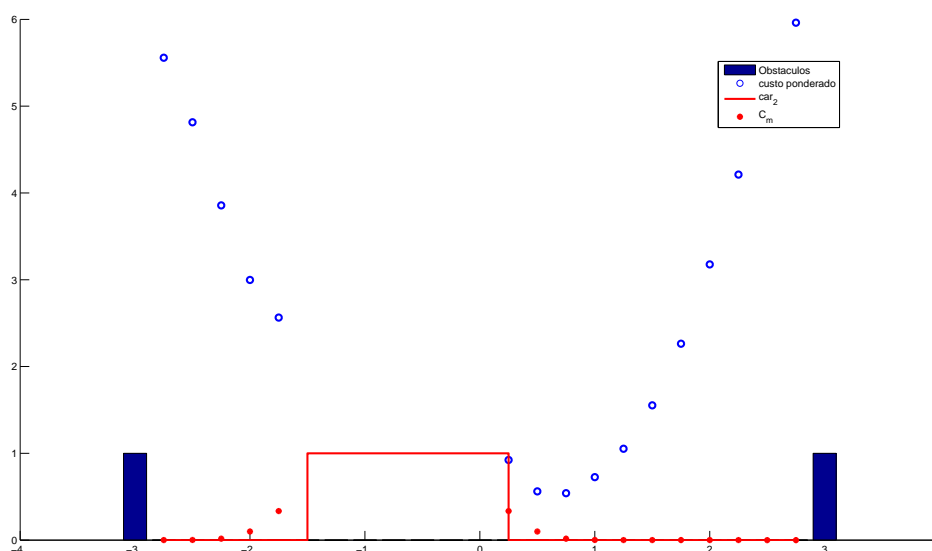


Figura 4.21: Soma ponderada do custo do caminhos possíveis para seguir um objeto em movimento

No caso contrário, em que a velocidade do objeto em movimento é menor e que no caminho estimado os dois objetos não vão bater, a função de custo ponderada é mostrado na figura 4.21, o caminho ótimo do ponto de vista do objeto em movimento seria ficar longe do objeto, mas ficar longe significa acrescentar o custo por curvatura, consistência e de objetos estáticos, portanto a ponderação dos elementos é muito importante e talvez sua normalização prévia não seja a melhor opção, principalmente no caso da curvatura onde podemos selecionar um caminho sub ótimo, por uma curvatura insignificante.

O custo do caminho ótimo nas seguintes iterações é apresentada na figura 4.22. A aproximadamente 10s vemos que existe o máximo custo, isso ocorre porque ali o caminho tem uma curva e também tem um objeto fixo, como é mostrado na figura 4.23, neste caso dado que existe um objeto estático (linhas brancas) o qual o veículo na frente passa muito próximo (com risco), o carro autônomo tenta minimizar evitando o obstaculo estático reduzindo o risco. Note-se que no seguinte período de amostragem do planejamento local o veiculo autônomo volta a seguir o caminho do veículo em movimento.

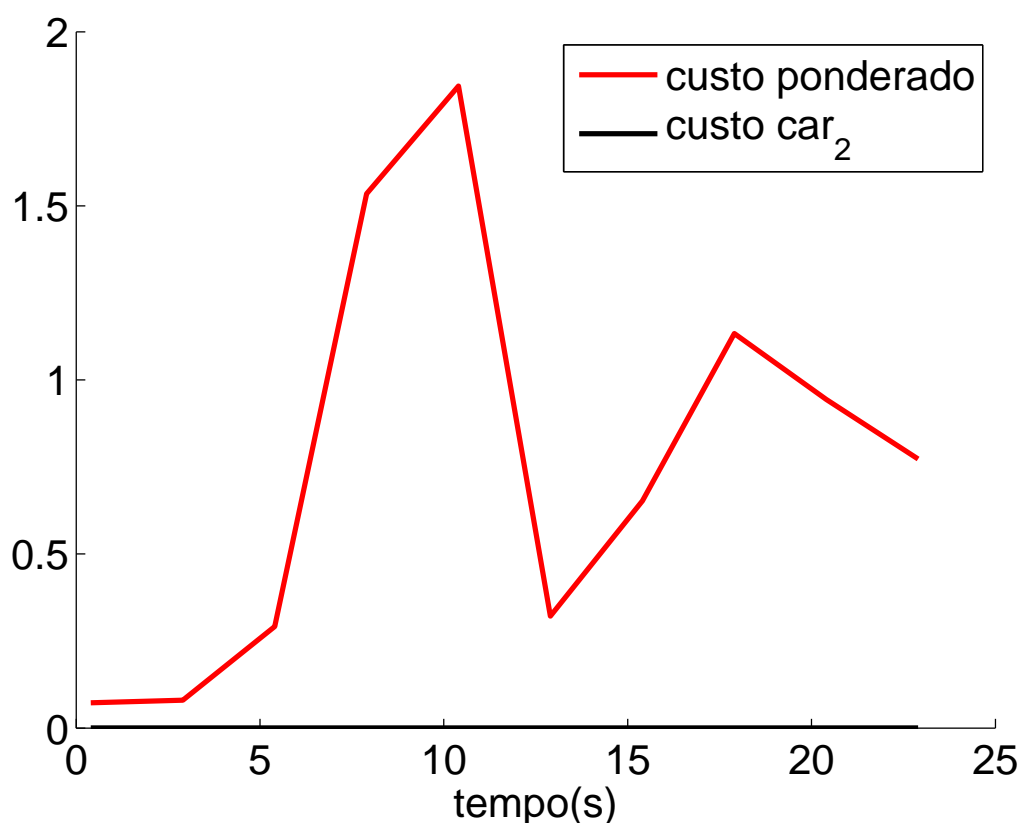


Figura 4.22: Soma ponderada do custo seguindo um objeto em movimento

4.7 Considerações finais

O presente capítulo apresentou a proposta de planejamento de trajetórias do veículo autônomo através da geração de caminhos candidatos em um espaço curvilíneo dependente do caminho global assumido conhecido. A técnica selecionada (CHU *et al.*, 2012) permitiu utilizar funções de risco intrínsecos e extrínsecos do veículo para selecionar o caminho ótimo. Simulações baseadas no modelo cinemático do veículo são feitas para validar o algoritmo proposto permitindo testar a extensão do método para objetos em movimentos. No capítulo a seguir é apresentado como pode ser usado os indicadores de risco calculados para o condicionamento do controle e do planejamento cooperativo.

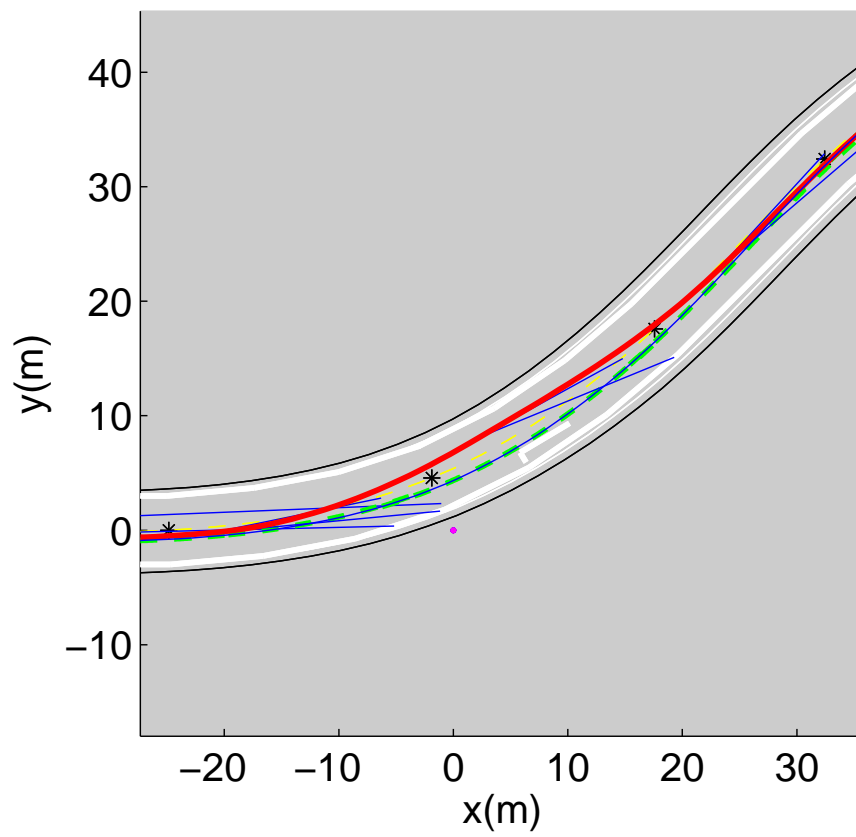


Figura 4.23: Ponto crítico do caminho onde existe um objeto estático em uma curva seguindo um objeto em movimento

5 Interação do condutor

Para o controle cooperativo e a transferência de controle é necessário primeiro conhecer a interação do condutor com o sistema de atuação do veículo tais como: Acelerador, Freio e Direção. O sensor de interação com o condutor deve ser tal que permita converter em um sinal elétrico a ação que o motorista normalmente executa e a proposta é usar os sensores que os veículos de gama média e alta já tem. O sensor do acelerador vai ser o sensor de deslocamento do pedal do acelerador, no freio vai ser um sensor de pressão do pedal do freio e no caso da direção um sensor de torque. Note-se que os dois primeiros já estão incorporados no VILMA, mas o último, que já existe comercialmente em direções assistidas eletricamente e sistemas *Steer-by-Wire*, não existe no VILMA. Como será explicado na seção 6.1.2 foi instalado um sensor de torque no eixo do volante da direção.

No controle da direção a proposta consiste em determinar como os sensores de interação serão utilizados na etapa de controle e planejamento de trajetórias para determinar o risco de uma transferência de controle dado que o caminho desejado pelo condutor seja diferente do planejado pelo sistema automático. Na figura 5.1 é apresentada a proposta de condicionamento do controle e planejamento cooperativo através da predição do risco. A primeira etapa que consiste no horizonte dinâmico utiliza os cálculos do controle preditivo para determinar no horizonte de predição do sistema do controlador o que vai acontecer se o sinal de atuação do sistema automático for desligado neste horizonte. A segunda etapa que consiste no horizonte cinemático usa a informação do ambiente para determinar as condições finais do que deseja fazer o condutor para determinar uma função analítica da curvatura do que o usuário vai realizar neste horizonte de predição. Com o resultado da simulação da trajetória os valores calculados podem ser convertidos ao espaço curvilíneo do planejamento para determinar o risco intrínseco e extrínseco a partir das funções de custo utilizadas na seleção da trajetória de referência.

Para entender a proposta de condicionamento da cooperação, primeiro será mostrado a simulação do controle de trajetória cooperativo ante trajetórias diferentes entre o condutor e o veículo autônomo. Em seguida será explicada a formulação matemática proposta para os horizontes dinâmicos e cinemáticos e finalmente é apresentada como é encontrada a divergências de trajetórias planejadas.

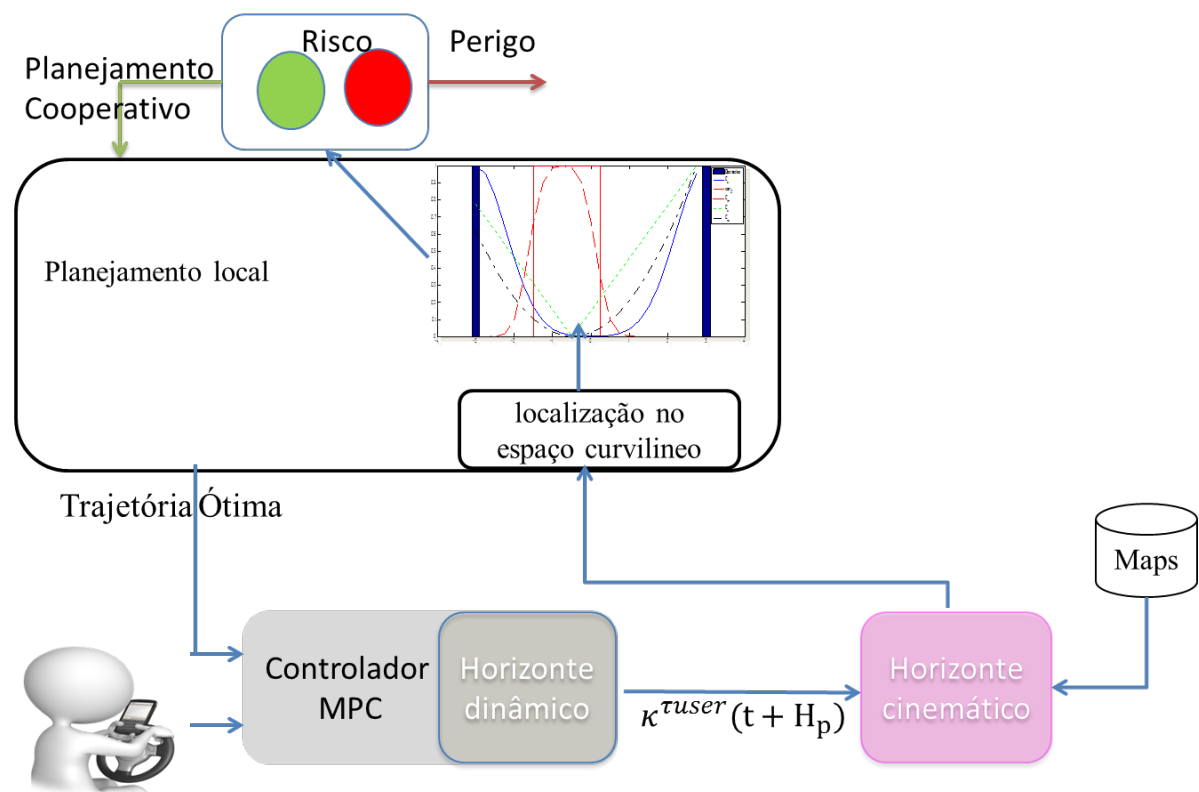


Figura 5.1: Diagrama de blocos da proposta de implementação do Risco para o condicionamento da cooperação

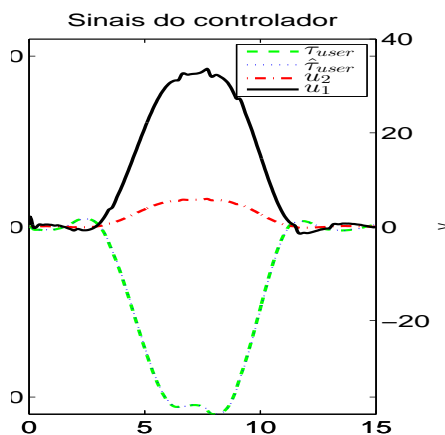


Figura 5.2: Sinais do controlador MPC e do controlador do condutor para referências diferentes $y_{mpc} = 3,5$ $y_{condutor} = 0$

5.1 Controle de trajetórias cooperativo para referências diferentes

Partindo das simulações apresentadas na seção 3.5, foi avaliado o desempenho do sistema se os sinais de referência do sistema automático de controle e do controlador do condutor desejam ir em trajetórias diferentes, para o qual será utilizado como exemplo o seguinte cenário: O condutor quer ir reto e o sistema automático quer ultrapassar um objeto, portanto fará o mesmo trajeto descrito na seção 3.4 variando o valor em y , sem modificar os pontos de início e final da trajetória em x . A configuração dos controladores será a mesma explicada 3.5 e igual nos dois sistemas de controle com $\beta = 5$, com retroalimentação das variáveis estimadas e com o controle automático medindo o torque do condutor.

A figura 5.2 mostra que ao tentar seguir a trajetória do controle automático gera condições de torque impossíveis fisicamente de realizar para qualquer dos dois sistemas de controle.

Na figura 5.3 é possível ver em cada instante de tempo k como se comportam os sinais a serem controlados (equação (3.19)) no último instante do horizonte de predição, $k + H_p$, de forma independente para cada um dos sinais de controle (v_m e τ_{user}). Dada a dinâmica do carro, que está relacionada com o tempo de atraso do ângulo de guinada, no erro lateral e de orientação, não é possível detectar visualmente a diferença do sinal de referência, mas tanto no ângulo da roda como na velocidade do erro lateral é perceptível a diferença nas saídas de controle do sistema automático.

Finalmente, para poder analisar as saídas de predição pelo controle no planejamento de trajetórias globais e locais, o ângulo das rodas $\delta_f(k + H_p)$ e o modelo cinemático do veículo foi

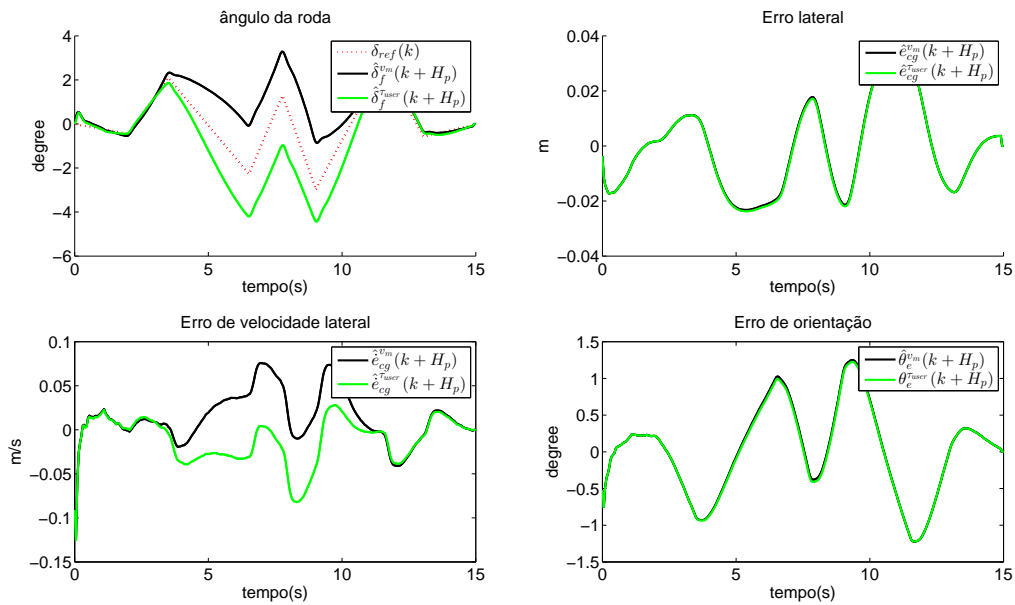


Figura 5.3: Predição de sinais de controle para referências diferentes $y_{mpc} = 3,5$ $y_{condutor} = 0$

convertido para curvatura através da equação $\kappa(k + H_p) = \tan(\delta_f(k + H_p))/L$ (BARFOOT E CLARK, 2004). Na figura (5.4) pode-se ver o comportamento da curvatura desejada pelo planejamento de trajetória do sistema automático ($\kappa(k)$) e as curvaturas no horizonte de predição dos sinais de controle ($\kappa^{vm}(k + H_p)$ e $\kappa^{tuser}(k + H_p)$). Este gráfico possibilitou formular a proposta dos horizontes dinâmico e cinemático que serão descritos a continuação.

5.2 Horizonte dinâmico

A estimação dos sinais de controle futuros do condutor e do sistema automático, obtidas a partir da equação (3.6), é denominado horizonte dinâmico. Para o caso do controle automático temos a equação (5.1),

$$\mathcal{Y}(k)^{vm} = \Psi \mathbf{x}(k) + \Phi \mathcal{U}_2(k) + \Upsilon \begin{bmatrix} v_m(k-1) \\ \hat{u}_2(k-1) \end{bmatrix} + \Theta \Delta \mathcal{U}(k) \quad (5.1)$$

na qual, $\Delta \mathcal{U}(k)$ é o sinal de controle obtida da otimização pela equação (3.15) onde $\mathbf{u}(k-1) = \begin{bmatrix} v_m & \hat{\tau}_{user} \end{bmatrix}^T$ e \hat{u}_2 é uma variável que o programa vai armazenando de quanto deveria ser o torque do condutor de acordo ao controle cooperativo. No caso da estimação do usuário temos a equação (5.2),

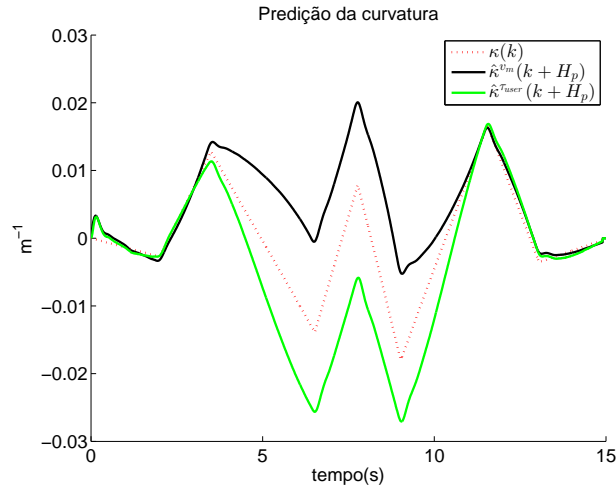


Figura 5.4: Predição da Curvatura para os diferentes esforços de controle

$$\mathcal{Y}(k)^{\tau_{user}} = \Psi \mathbf{x}(k) + \Phi \mathcal{U}_2(k) + \Upsilon \begin{bmatrix} 0 \\ \hat{\tau}_{user}(k-1) \end{bmatrix} + \Theta \begin{bmatrix} \begin{bmatrix} 0 \\ \Delta \hat{\tau}_{user} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} 0 \\ \Delta \hat{\tau}_{user} \end{bmatrix} \end{bmatrix} \quad (5.2)$$

na qual, $\Delta \hat{\tau}_{user} = \hat{\tau}_{user}(k-1) - \hat{\tau}_{user}(k-2)$ é a variação do sinal de torque do condutor no horizonte de predição, assumido constante e igual à tendência atual. Note-se que os primeiros dois termos são comuns nas duas predições e que já foram calculados previamente no cálculo do sinal de controle, portanto o custo computacional da predição é mínimo. Note-se também que o tempo do horizonte dinâmico aqui adotado é corresponde ao tempo do horizonte de predição, ou seja $k + H_p$, utilizado na técnica de controle preditivo.

5.3 Horizonte cinemático

O problema de realizar um horizonte dinâmico maior consiste na suposição de que o torque do condutor e ou variação do mesmo vai ser constante durante todo o horizonte, portanto a proposta consiste em usar um modelo cinemático do veículo da bicicleta, a partir da equação (G.9). A partir do cenário e do mapa conhecido as seguintes suposições são realizadas:

- O perfil de velocidade é conhecido no horizonte de predição cinemático, portanto as curvaturas $\kappa^{\tau_{user}}(k)$ até $\kappa^{\tau_{user}}(k + H_p)$ podem ser transformadas ao espaço de longitude da

curva ou comprimento de arco pela equação (5.3).

$$s^{\tau_{user}}(m) = \sum_{i=0}^m v_{car}(k-i) T_c \quad (5.3)$$

- Num comprimento de arco, $s_{max}^{\tau_{user}}$, o ângulo de guinada vai ser um valor constante, portanto $\theta^{\tau_{user}}(s_{max}^{\tau_{user}}) = \theta_{final} - \hat{\theta}_{car}(k)$
- Num comprimento de arco, $s_{zero}^{\tau_{user}}$, a curvatura aplicada no carro vai ser o valor da estrada até $s_{max}^{\tau_{user}}$, portanto $\kappa^{\tau_{user}}(s_{zero}^{\tau_{user}} : s_{max}^{\tau_{user}}) = \kappa_b$
- A função matemática que descreve a curvatura realizada pelo usuário vai ser do tipo polinomial, descrita pela equação (5.4),

$$\begin{aligned} \kappa^{\tau_{user}}(s^{\tau_{user}}) &= k_0 + k_1 s^{\tau_{user}} + k_2 (s^{\tau_{user}})^2 \dots + k_n (s^{\tau_{user}})^n \\ &= \begin{bmatrix} 1 & s^{\tau_{user}} & (s^{\tau_{user}})^2 & \dots & (s^{\tau_{user}})^n \end{bmatrix} \mathbf{k}_u \end{aligned} \quad (5.4)$$

- O modelo cinemático do carro será o apresentado na seção (G.1). Portanto o ângulo de guinada é definido pela integral com respeito a $s^{\tau_{user}}$, como é apresentado em (LI *et al.*, 2014) pode ser descrito pela equação (5.5),

$$\hat{\theta}^{\tau_{user}}(s^{\tau_{user}}) = \int_0^{s_{max}^{\tau_{user}}} \kappa^{\tau_{user}}(s^{\tau_{user}}) ds^{\tau_{user}} \quad (5.5)$$

Encontrar a função polinomial da curvatura é um problema de ajuste de curvas dos coeficientes da função que descreve a curvatura, o qual pode ser resolvido analiticamente porque a equação (5.5), gerará também um polinômio e permitirá formar um sistema de equações lineares, descritos pela equação (5.6), que pode ser resolvido como um problema de otimização realizando a mesma metodologia da seção 3.2.2.

$$\begin{aligned}
\begin{bmatrix} \kappa^{\tau_{user}}(k+0) \\ \vdots \\ \kappa^{\tau_{user}}(k+H_p) \\ \kappa^{\tau_{user}}(s_{zero}^{\tau_{user}}) \\ \vdots \\ \kappa^{\tau_{user}}(s_{max}^{\tau_{user}}) \\ \theta^{\tau_{user}}(s_{max}^{\tau_{user}}) \end{bmatrix} &= \begin{bmatrix} k_0 \\ \vdots \\ k_0 + k_1 s^{\tau_{user}}(H_p) \dots + k_n (s^{\tau_{user}}(H_p))^n \\ k_0 + k_1 s_{zero}^{\tau_{user}} \dots + k_n (s_{zero}^{\tau_{user}})^n \\ \vdots \\ k_0 + k_1 s_{max}^{\tau_{user}} \dots + k_n (s_{max}^{\tau_{user}})^n \\ k_0 s_{max}^{\tau_{user}} + \frac{k_1}{2} (s_{max}^{\tau_{user}})^2 \dots + \frac{k_n}{n+1} (s_{max}^{\tau_{user}})^{n+1} \end{bmatrix} \\
\mathbf{y}_u(k) &= \Theta_u \mathbf{k}_u(k) \\
&= \begin{bmatrix} 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & s^{\tau_{user}}(H_p) & \dots & (s^{\tau_{user}}(H_p))^n \\ 1 & k_1 s_{zero}^{\tau_{user}} & \dots & (s_{zero}^{\tau_{user}})^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & s_{max}^{\tau_{user}} & \dots & (s_{max}^{\tau_{user}})^n \\ s_{max}^{\tau_{user}} & \frac{1}{2} (s_{max}^{\tau_{user}})^2 & \dots & \frac{1}{n+1} (s_{max}^{\tau_{user}})^{n+1} \end{bmatrix} \begin{bmatrix} k_0 \\ k_1 \\ \vdots \\ k_n \end{bmatrix}
\end{aligned} \tag{5.6}$$

A função objetivo é descrita pela equação (5.7)

$$V(k) = \|\Theta_u \mathbf{k}_u(k) - \mathbf{y}_u(k)\|_{Q_u}^2 \tag{5.7}$$

na qual a matriz Q_u é uma matriz de ponderação de dimensão do número de equações lineares. Resolvendo a norma quadrática e derivando-a obtêm-se \mathbf{k}_u que é descrito pela equação (5.8). Note-se que a matriz \mathcal{M}_u só vai ser alterada se o perfil de velocidades mudar porque em qualquer outro caso ela vai ser constante o que faz com que a predição da função de curvatura possa ser implementada em tempo real facilmente.

$$\begin{aligned}
\mathbf{k}_u &= (\Theta_u^T Q_u \Theta_u)^{-1} \Theta_u^T Q_u \mathbf{y}_u(k) \\
&= \mathcal{M}_u \mathbf{y}_u(k)
\end{aligned} \tag{5.8}$$

Uma vez encontrada a função de curvatura estimada da manobra do condutor no instante k , o sistema poderá estimar a posição final $\hat{x}_{user}(s_{max}^{\tau_{user}})$ e $\hat{y}_{user}(s_{max}^{\tau_{user}})$ para que o planejador

consiga de forma prematura referenciar a posição futura do veículo e estimar o risco intrínseco e extrínseco do que deseja o condutor. Isso é realizado pela integração numérica do sistema de equações (5.9),

$$\begin{aligned}
 \hat{x}(s^{\tau_{user}}) &= \int_0^{s_{max}^{\tau_{user}}} \cos \hat{\theta}^{\tau_{user}}(s^{\tau_{user}}) ds^{\tau_{user}} \\
 \hat{y}(s^{\tau_{user}}) &= \int_0^{s_{max}^{\tau_{user}}} \sin \hat{\theta}^{\tau_{user}}(s^{\tau_{user}}) ds^{\tau_{user}} \\
 \hat{\theta}^{\tau_{user}}(s^{\tau_{user}}) &= \int_0^{s_{max}^{\tau_{user}}} \kappa^{\tau_{user}}(s^{\tau_{user}}) ds^{\tau_{user}}
 \end{aligned} \tag{5.9}$$

Para o caso quando existe diferença de trajetória, $y_{mpc} = 0.3$ e $y_{condutor} = 0$, temos os sinais de controle cooperativo da figura (5.5a) que nos permite observar visualmente a diferença de trajetórias desejadas. Na figura 5.5b é observado as curvaturas estimadas no final do horizonte dinâmico (verde e preta), a referência (vermelha) e as estimadas no horizonte cinemático (azul) que são calculadas a uma frequência $f_{path} = 10hz$ durante os momentos em que a diferença das predições é significativa. Note-se que cada curvatura usada para estimar a função de curvatura ($\kappa^{\tau_{user}}(s^{\tau_{user}})$) do instante k obtida através da equação (5.6) é uma predição futura no final do horizonte dinâmico ($l + H_p$) no instante $l = k - i$, onde i varia entre 1 até H_p .

Na figura 5.6 é apresentada a posição atual estimada do veículo ($\hat{x}_{car}(k), \hat{y}_{car}(k)$) em pontos vermelhos e as trajetórias estimadas do que deseja fazer o condutor em um horizonte de predição de $25m$ em cor azul. Os parâmetros para a simulação foram $\theta_{final} = 0$, $v_{car}(k + i) = v_{car}(k) = 10m/s$, $s_{zero}^{\tau_{user}} = 24,98m$, $s_{max}^{\tau_{user}} = 25m$ e $n = 5$.

5.4 Detecção de trajetória diferente

A ideia do algoritmo de detecção de divergência da trajetória é saber o mais rápido possível a intenção do condutor em fazer uma outra trajetória. Porém, na figura 5.5a é fácil comparar visualmente os torques e perceber que quando as derivadas dos sinais estão em direções contrárias o sistema quer ir para lados opostos, entretanto o ruído da medição do torque do usuário não permite que este seja um método válido na etapa inicial da mudança de trajetória. Portanto,

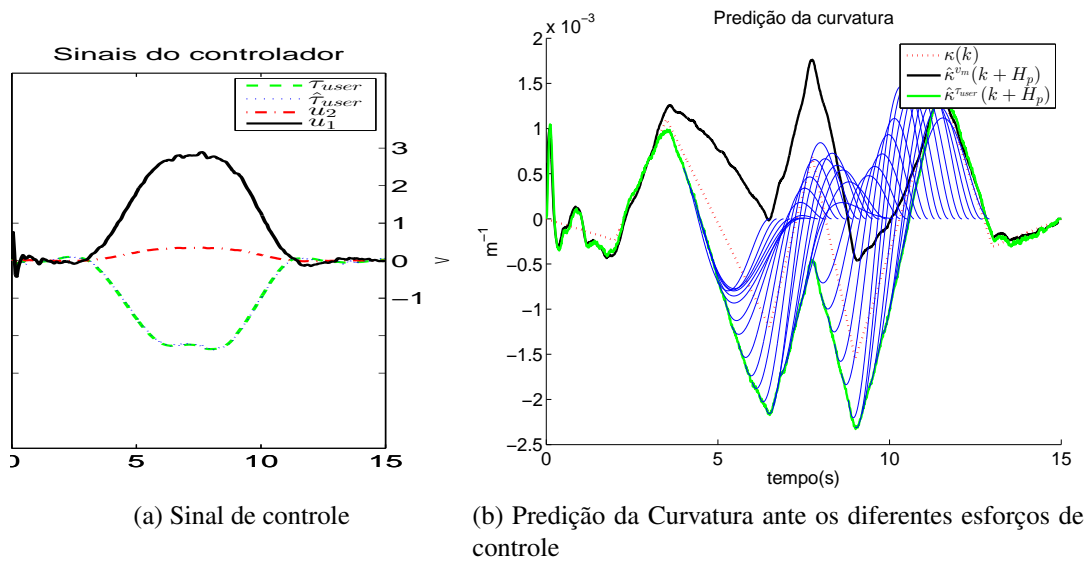


Figura 5.5: Sinais do controlador MPC e do controlador do condutor para referências diferentes $y_{mpc} = 0.3$ $y_{condutor} = 0$

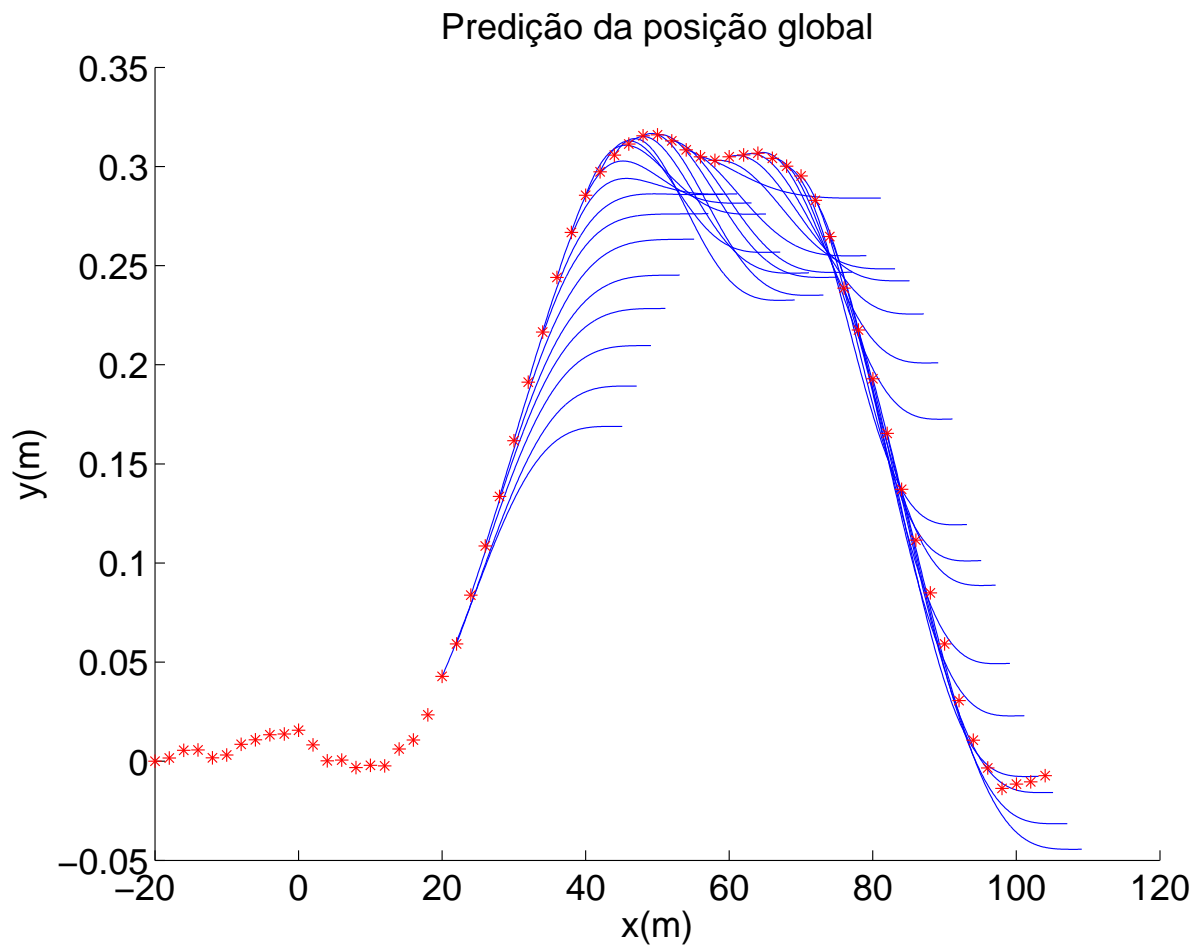


Figura 5.6: Posição atual estimada do veículo $(\hat{x}_{car}(k), \hat{y}_{car}(k))$ em pontos vermelhos e as trajetórias estimadas (azul) do que deseja fazer o condutor em um horizonte cinemático de $25m$.

a ideia é comparar os sinais dos valores dos ângulos da roda $\delta_f^{t_{user}}$ e $\delta_f^{v_m}$ obtidos pela equações (5.1) e (5.2) respectivamente. Note-se que dependendo dos parâmetros do controle cooperativo os sinais podem ser contrários e não existir falha, portanto a diferença entre eles no final do horizonte dinâmico deve ser maior a um parâmetro calculado heurísticamente. As condições lógicas que detetam a diferença são descritas pela equação (5.10).

$$sign\left(\overline{\Delta\delta_f^{t_{user}}(t:t+H_p)}\right)! = sign\left(\overline{\Delta\delta_f^{v_m}(t:t+H_p)}\right) \text{ AND } |\delta_f^{t_{user}}(t+H_p) - \delta_f^{v_m}(t+H_p)| > \delta_{min} \quad (5.10)$$

Note-se que este cálculo é realizado no programa de controle que é realizado a uma frequência de amostragem maior, gerando duas vantagens: A primeira é poder ter diferentes estimativas de detecção de falha durante um período de amostragem do planejamento de trajetórias, e A segundo, é reduzir a quantidade de informação que deve ser enviada do módulo de controle ao módulo de planejamento, pois uma vez feita a detecção da falha não é necessário enviar o vetor $\delta_f^{v_m}(t:t+H_p)$.

5.5 Simulação

Nesta simulação o espaço global é descrito pelo mesmo caminho descrito na seção (3.4). A figura 5.7 apresenta as curvaturas desejadas pelo sistema automático (vermelho), a desejada pelo condutor (Amarelo) e curvaturas estimadas no horizonte $\delta_f^{t_{user}}(t+H_p)$ e $\delta_f^{v_m}(t+H_p)$ em verde e preto respectivamente. Finalmente, na mesma figura é mostrado em marrom a curvatura predita pelo sistema, assumindo que $\kappa^{t_{user}}(s_{zero}^{t_{user}} : s_{max}^{t_{user}})$ e $\theta^{t_{user}}(s_{max}^{t_{user}})$ serão as mesmas do caminho global depois de 30m.

Neste caso para a estimação da curvatura primeiro foram avaliadas duas alternativas, a primeira utilizando na estimação do horizonte cinemático $\delta_f^{t_{user}}(k:k+H_p)$ e a segunda utilizando $\delta_f^{t_{user}}(l+H_p)$ com $l = k - i$, onde i varia entre 1 até H_p . A figura 5.8 apresenta em vermelho o realizado pelo veículo, em verde a desejada pelo condutor, em azul a estimada com o primeiro método e em marrom a estimada pelo segundo método. Note-se que como busca-se uma proposta de detetar a divergência de trajetórias o mais rápido possível o desejado pelo condutor será adotado o primeiro método, entretanto se o desejado fosse prever o que o condutor estaria fazendo em modo manual o mais indicado seria utilizar o segundo método porque ele contém mais informações sendo utilizadas na estimação.

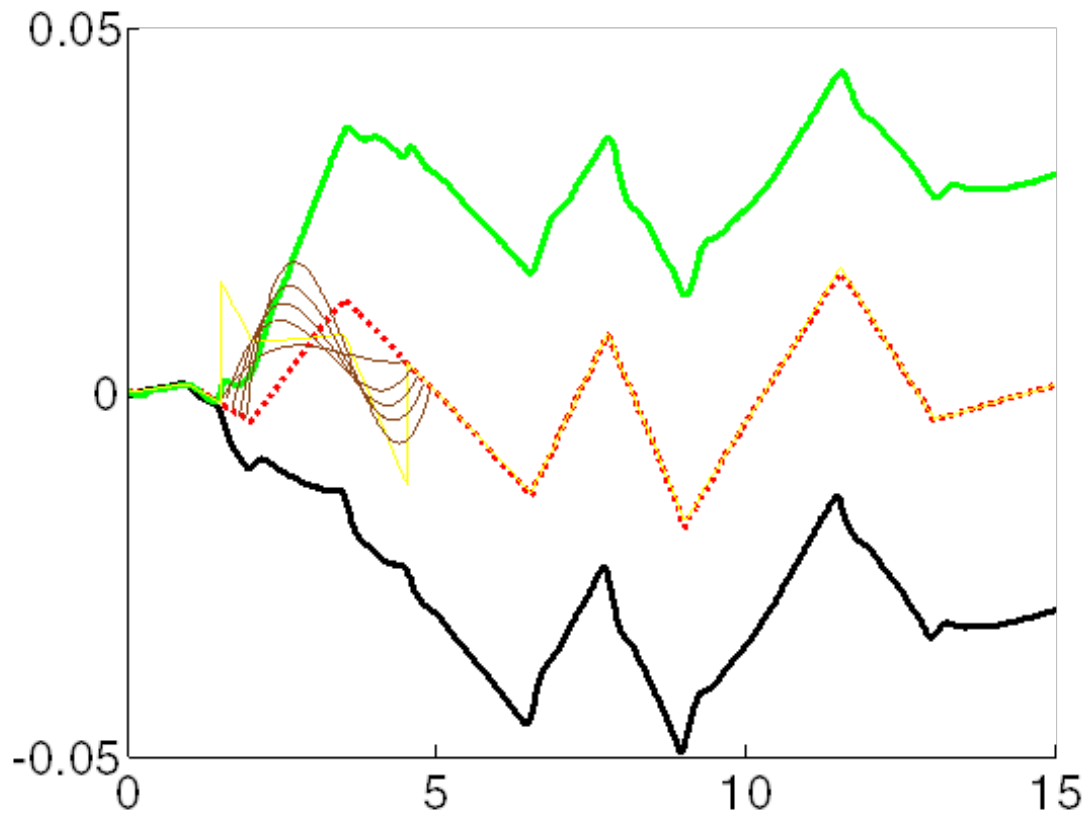


Figura 5.7: Curvatura na simulação de trajetórias diferentes

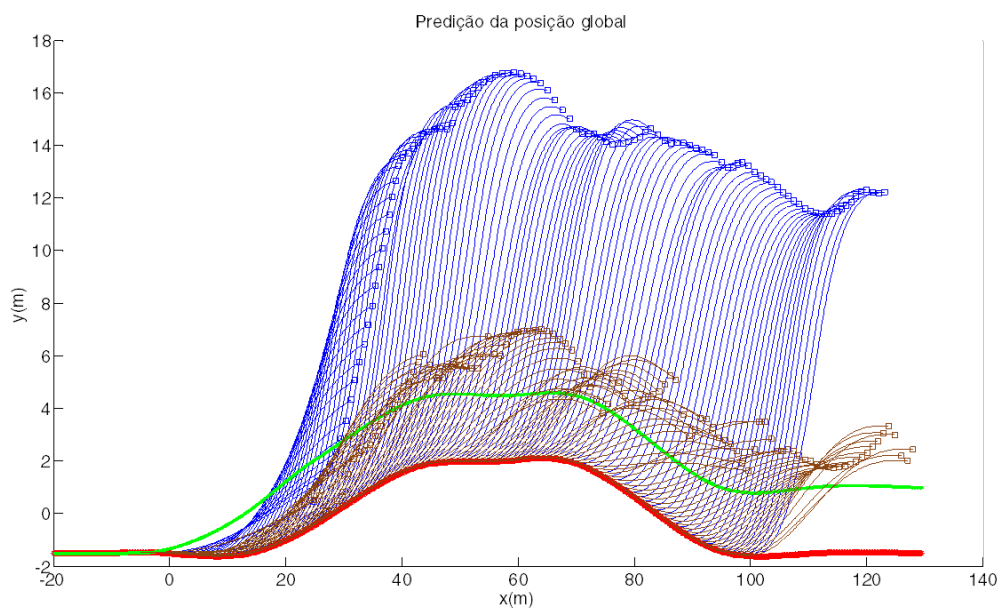


Figura 5.8: Predição da trajetória desejada pelo condutor

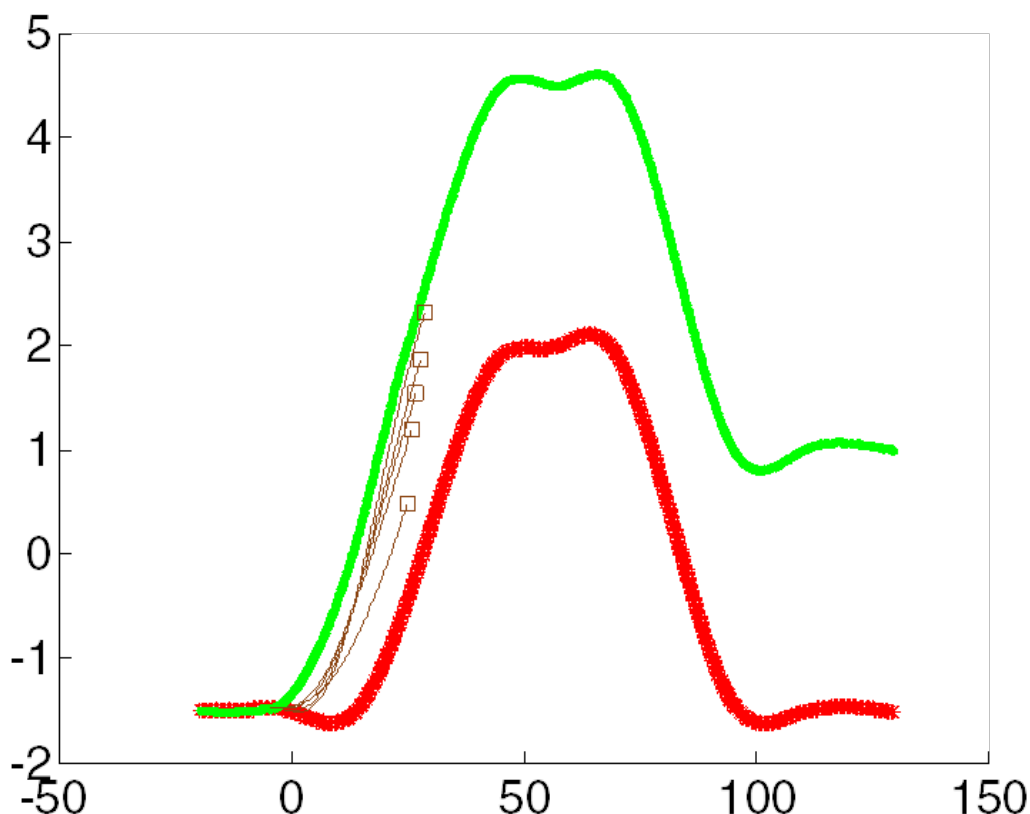


Figura 5.9: Predição da trajetória desejada do condutor no horizonte cinemático

Portanto utilizando o primeiro método, na figura 5.9 é apresentado a predição da trajetória estimada do condutor (marrom) onde é possível ver que desde o início da mudança é possível detectar para onde vai o condutor.

Finalmente, para encontrar as funções de risco das trajetórias, o planejamento deve receber a curvatura ($\kappa^{T_{user}}$) e os valores q e s , portanto, o processo de localização no espaço curvilíneo deve ser realizado com o resultado da equação (5.9). Uma vez feito o processo de localização no espaço curvilíneo o valor de q obtido no final do horizonte cinemático é procurado nas funções de risco da iteração atual (exemplo figura 4.19). Se o risco é baixo (Verde no semáforo) o planejamento de trajetórias deve replanejar o caminho de acordo ao desejado pelo condutor, gerando um planejamento cooperativo, e caso contrário se o risco for alto a estratégia vai ser continuar com a trajetória atual e poderá gerar sinais de alarme. O procedimento pode ser ressumido pela figura 5.10, onde é apresentado o procedimento realizado a cada ponto do planejamento de trajetórias onde o controle detectou uma divergência no torque com o condutor.

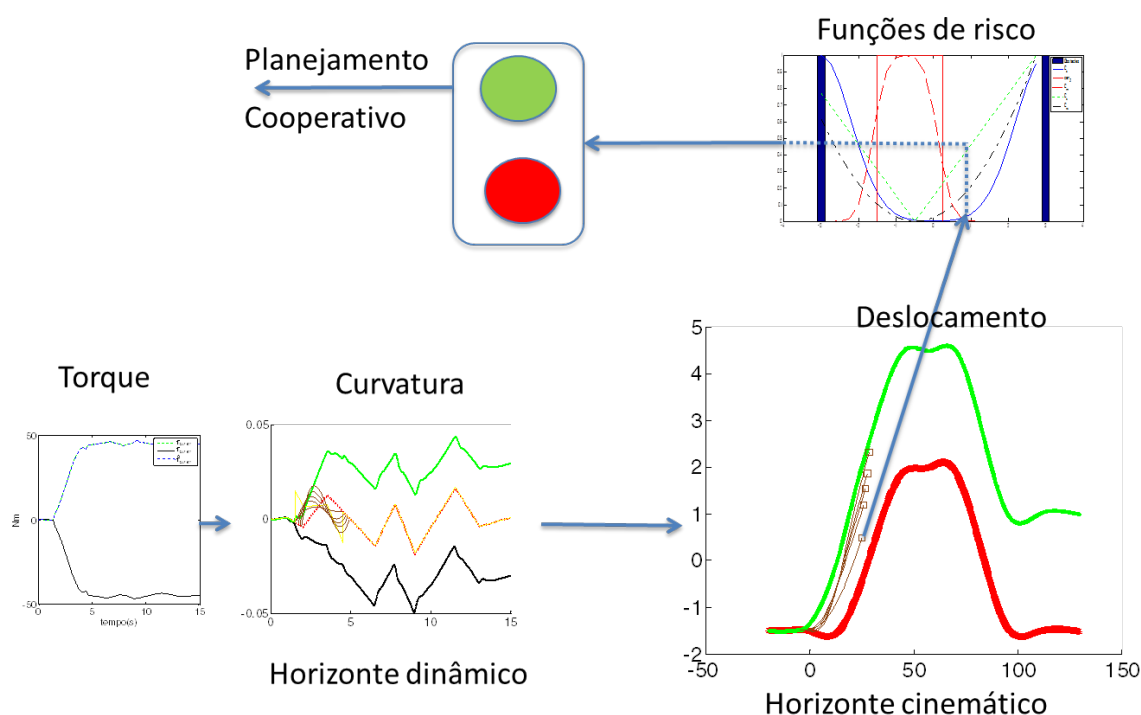


Figura 5.10: Simulação da medição do risco condicionamento da cooperação entre o condutor e o sistema autônomo.

5.6 Considerações finais

O presente capítulo apresentou a proposta de contribuição da tese. Ela consiste em condicionar o controle cooperativo e o planejamento cooperativo a partir de uma predição da trajetória desejada do condutor através de modelos dinâmicos e cinemáticos do veículo. Entre as vantagens encontradas no método proposto se encontra a utilização dos mesmos cálculos realizados na etapa de controle e seleção da trajetória ótima no planejamento. No capítulo a seguir são apresentados os resultados experimentais realizados na plataforma projetada (VILMA01), que consistiram em analisar a instrumentação do veículo em testes para validar os conceitos de controle e planejamento cooperativo.

6 Resultados Experimentais

Depois de implementada a automação do veículo foram feitos testes funcionais de cada um dos elementos do hardware e do software do sistema embarcado os quais são descritos no apêndice D. Estes testes são apresentados em tabelas descrevendo a funcionalidade testada, o procedimento realizado e as conclusões do teste. Depois destes testes foram realizados experimentos de controle do veículo através de um gamepad permitindo que o carro pudesse ser totalmente dirigido do banco do copiloto. Finalmente foram realizados experimentos para analisar a instrumentação do veículo e testes da percepção de VILMA01 através de uma câmera na frente do veículo.

A seguir serão apresentados os resultados da análises da instrumentação do VILMA01 para a localização espacial do veículo. Depois procurando testar os conceitos de condicionamento de controle e planejamento cooperativo propostos nesta tese será apresentado o desenvolvimento simplificado de um nó de controle cooperativo baseado em imagens implementado em ROS e os resultados do VILMA01 em modo cooperativo.

6.1 Resultados da instrumentação do VILMA01

Com a finalidade de encontrar as capacidades do veículo automatizado e determinar a viabilidade de implementar a técnica de controle cooperativo no VILMA01 foram necessários realizar uma serie de testes para analisar a informação da instrumentação do veículo, a qual é descrita a seguir.

6.1.1 Modelo estático do motor de direção

O modelo do torque gerado pelo motor da direção ao sistema foi verificado a partir da superposição dos gráficos da equação (G.25) com as figuras do fabricante do motor como é mostrado na figura 6.1, onde em linhas pontilhadas é apresentado como varia o torque gerado e a velocidade do motor de acordo com a voltagem (cor do ponto) de entrada na equação.

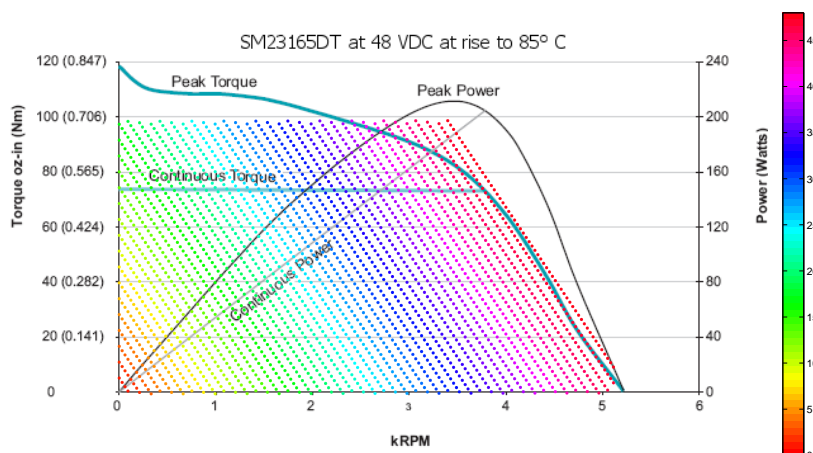


Figura 6.1: Caracterização estática do motor da direção.

6.1.2 Calibração do sensor de torque da direção

O sensor de torque são extensômetros ligados em ponte de Wheatstone completo configurado para medir torção em aço sem compensação de temperatura, cuja referência é 1-VY41-3/350 do fabricante HBM. O sensor está conectado a um sistema de instrumentação com sensor de temperatura incluso e com comunicação sem fio da empresa Microstrain. O sensor foi projetado para ser instalado na barra de união do sistema de direção, porque ela é a que tem menor parede no tubo o que acrescentaria a sensibilidade da medida, mas por limitações de espaço, o sensor teve que ser instalado atrás do volante na barra com menor sensibilidade como é mostrado na figura 6.2. É importante salientar que por não serem conhecidas as características do material e o diâmetro interno do eixo não foi possível obter um modelo do torque medido, e portanto foi necessário fazer uma calibração de $\mu m/m$ para Nm .

Para a obtenção de variação de resistência para $\mu m/m$, foram usados os dados do fabricante dos extensômetros e o software assistente da instrumentação. Depois disso foram adquiridos dados da relação torque versus $\mu m/m$, através de um torquímetro digital ao qual era aplicado uma carga constante através de uma massa pendurada em sua extremidade. A figura 6.3 apresenta as medições realizadas pela instrumentação do veículo durante a etapa de calibração. Note-se que a temperatura durante o experimento não foi constante. Outras calibrações não foram feitas por baixa disponibilidade do torquímetro.



Figura 6.2: Localização do extensômetro no eixo do volante

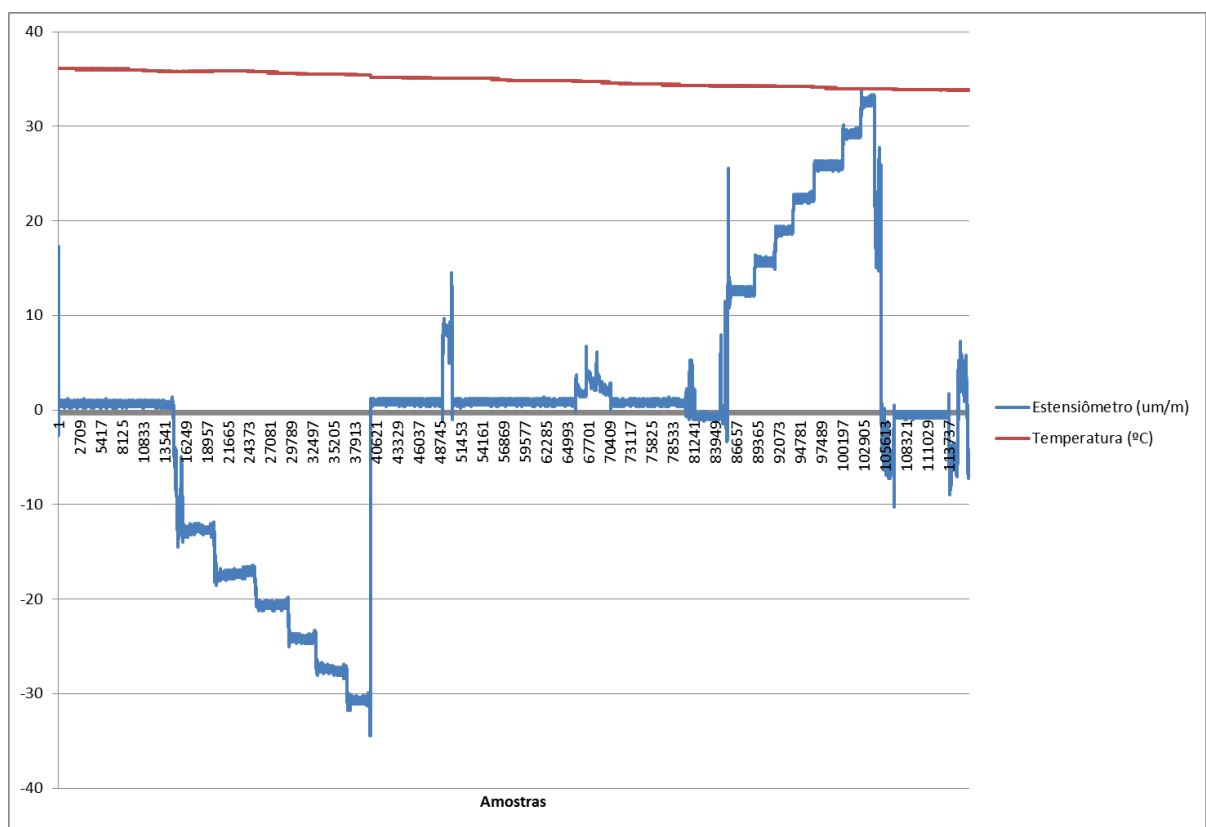


Figura 6.3: Medidas do extensômetro durante o processo de calibração

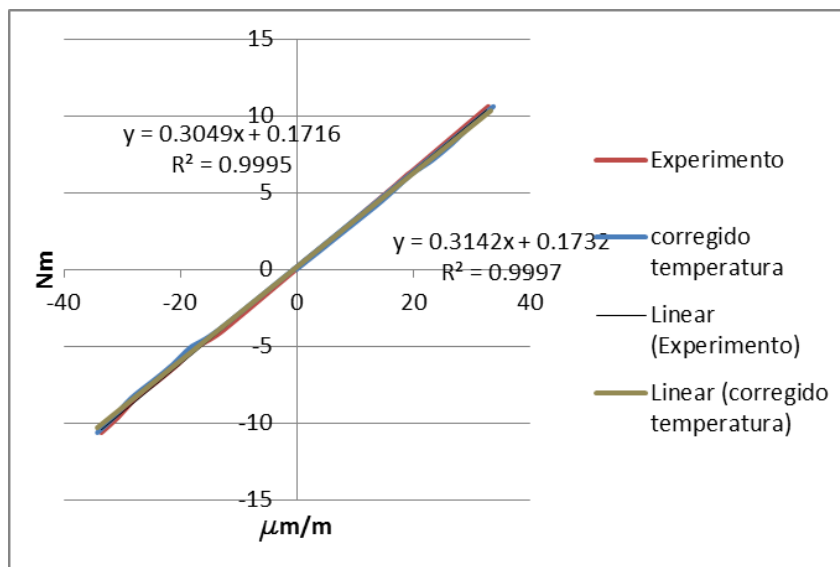


Figura 6.4: Relação entre Nm e $\mu m/m$ do extensômetro.

O torquímetro digital utilizado na calibração não apresenta nenhuma saída, portanto as medidas foram tomadas de forma visual com uma resolução de $\pm 0.1 Nm$. A figura 6.4 apresenta os resultados da medição. Na figura são apresentados dois resultados, o primeiro o resultado medido pela instrumentação, e o segundo a medida corrigida de acordo com as curvas do fabricante do extensômetro para a temperatura promédio do experimento. Como pode ser visto o índice R-quadrado da regressão linear das duas retas é perto de um, portanto pode-se aproximar a uma reta.

A forma como o dado do sensor chega ao programa do MicroAutobox é descrito pela figura 6.5. Em primeira instância a instrumentação amplifica o sinal de voltagem diferencial, depois o converte para um sinal digital e transforma seu valor em $\mu m/m$. Este valor é enviado por uma sinal sem fio a um período de amostragem de 128hz ao receptor que converte o número de ponto flutuante em volts, configurando a tensão máxima ($3V$) para $60\mu m/m$ e o mínimo ($0V$) para $-60\mu m/m$. Este valor enviado é convertido novamente pelo sistema embarcado do MicroAutobox com um conversor analógico digital em um valor de ponto flutuante de 32 bits definido como x , sendo 0 para $0V$ e 1 para $5V$. Uma vez no MicroAutobox o valor deve ser convertido de novo a $\mu m/m$ pela equação (6.1).

$$X = \frac{(5x - 1.5)120\mu m/m}{1.5V} \quad (6.1)$$

Note-se, que a resolução do conversor analógico para digital do embarcado (14 bits) é maior que a resolução do conversor digital para analogico portanto não existe perda de infor-

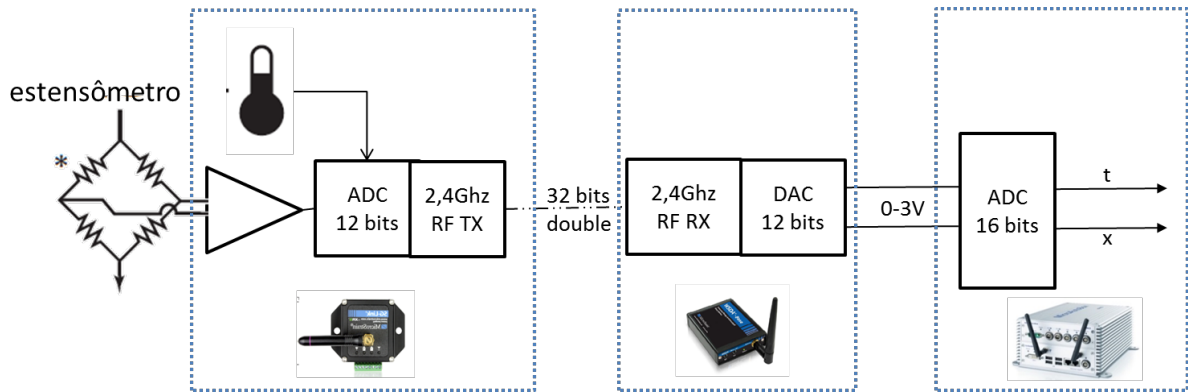


Figura 6.5: Diagrama de blocos da instrumentação do sensor de torque

mação.

O segundo passo é realizar a compensação por temperatura, tendo em conta que o sinal de temperatura tem um caminho similar ao sinal do extensômetro, a temperatura deve ser convertida pela equação $T = 5000.t/6,25 - 67,84$. Portanto, o valor corrigido que chamaremos ϵ é descrito pela equação (6.2).

$$\epsilon = X + (-15,48 + 1,72T - 0,055T^2 + 0,000232T^3) \quad (6.2)$$

Finalmente, o valor é convertido a torque (Nm) com a equação (6.3)

$$\tau_{user} = 0.3049\epsilon - 2.69 \quad (6.3)$$

Aproximando a uma distribuição normal o erro esperado é aproximadamente de $\pm 0.2Nm$. Assim, assumindo um intervalo de confiança de 99,6%, o equivalente a 3 desvios padrões, resulta em um desvio padrão de $\sigma_{\tau_{user}} \approx 0.07Nm$. Note-se que a suposição de erro com distribuição gaussiana é a representação dos seguintes erros possíveis: serrilhamento (*aliasing*) nos sinais dado os diferentes períodos de amostragem, a temperatura, o erro do sensor de torque de referência, erros de quantização, entre outros.

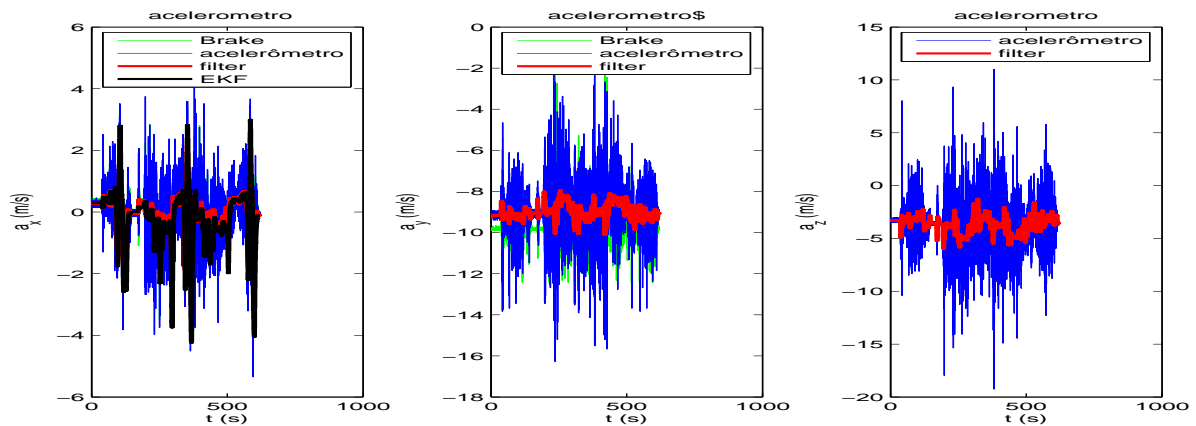


Figura 6.6: Acelerômetros durante percurso um filtrados e sem filtrar.

6.1.3 Acelerômetros e barômetro

O sistema embarcado tem acelerômetros nos três eixos e um barômetro, adicionalmente o freio gera a informação do acelerômetros no eixo lateral e vertical do veículo, mas a localização do MicroAutobox foi instalado na parte traseira do encosto da cadeira dos passageiros portanto as medições tem uma diferença estática causada principalmente pelo ângulo de inclinação do encosto. A figura 6.6 apresenta os valores medidos durante o percurso 1 (figura 6.9), assim como os valores filtrados com um passa baixos e o resultado do valor estático calculado pelo filtro de Kalman apresentado na seção 6.1.7.

A partir dos sensores do MicroAutobox tratados com filtros passa baixo aplicando as fórmulas descritas em (PEDLEY, 2013) para os acelerômetros e a fórmula barométrica (KREYSZIG, 2010) para obter a altitude a nível do mar, foram obtidos os ângulos de rolagem e inclinação e a altitude da rodovia como é apresentado na figura 6.7. A modo de verificação dos dados a altitude é comparada com o valor do GPS e o ângulo de inclinação é comparado com o ângulo estimado a partir da equação (6.4),

$$pitch_{barometro}(k) \approx \arcsin \left(\frac{h(k) - h(k-1)}{\hat{v}_x(k)Ts} \right) \quad (6.4)$$

na qual $h(k)$ é a altitude estimada, \hat{v}_x e a velocidade estimada do veículo e Ts é o período de amostragem.

As figuras 6.8a, 6.8b e 6.8c apresentam a altitude da estrada, o ângulo de rolagem e o ângulo de inclinação sobre o percurso um (figura 6.9). Note-se que o percurso foi realizado mais de duas vezes e os dados são muito similares. No ângulo de inclinação existem muitas

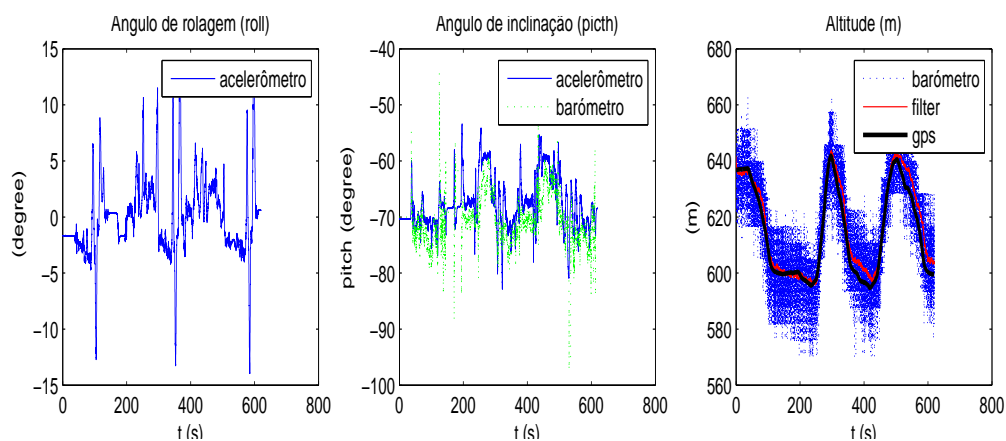
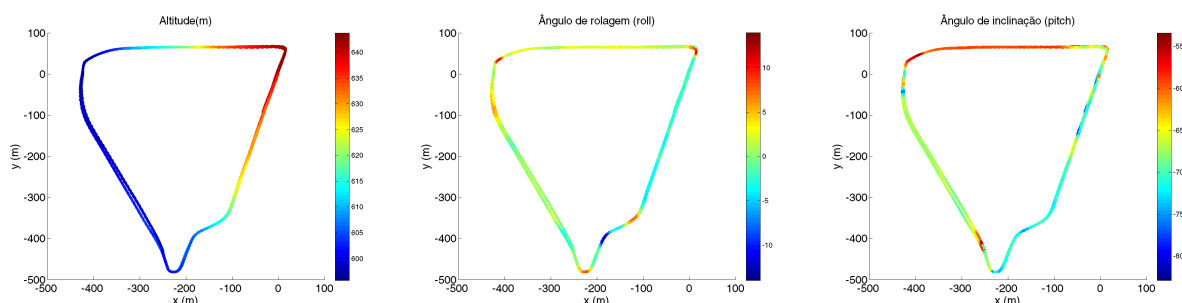


Figura 6.7: altitude, ângulos de rolagem e inclinação estimadas durante o percurso um



(a) Gráfico de altitude estimada com o (b) Ângulo rolagem do acelerômetro (c) Ângulo inclinação do acelerômetro barômetro

Figura 6.8: Altitude, ângulos de rolagem e inclinação estimadas da estrada para o percurso um perturbações sendo as mais relevantes as que ocorrem nas lombadas do percurso.

6.1.4 Sensor GPS

O sensor GPS utilizado é um u-blox referência EVK-6V de baixo custo projetado para veículos. Na figura 6.9 e 6.10 são apresentados dois percursos realizados durante os testes. Os valores medidos pelo GPS são comparados de forma visual com o planejado pelo Google Maps através do site. O GPS adicionalmente gera uma estimativa do vetor velocidade a qual será usada para comparar à velocidade do veículo e a partir de um limite de velocidade é usado para estimar o ângulo de guinada do veículo.



Figura 6.9: Comparação entre o gps e rota planejada pelo Google Maps no recorrido um.

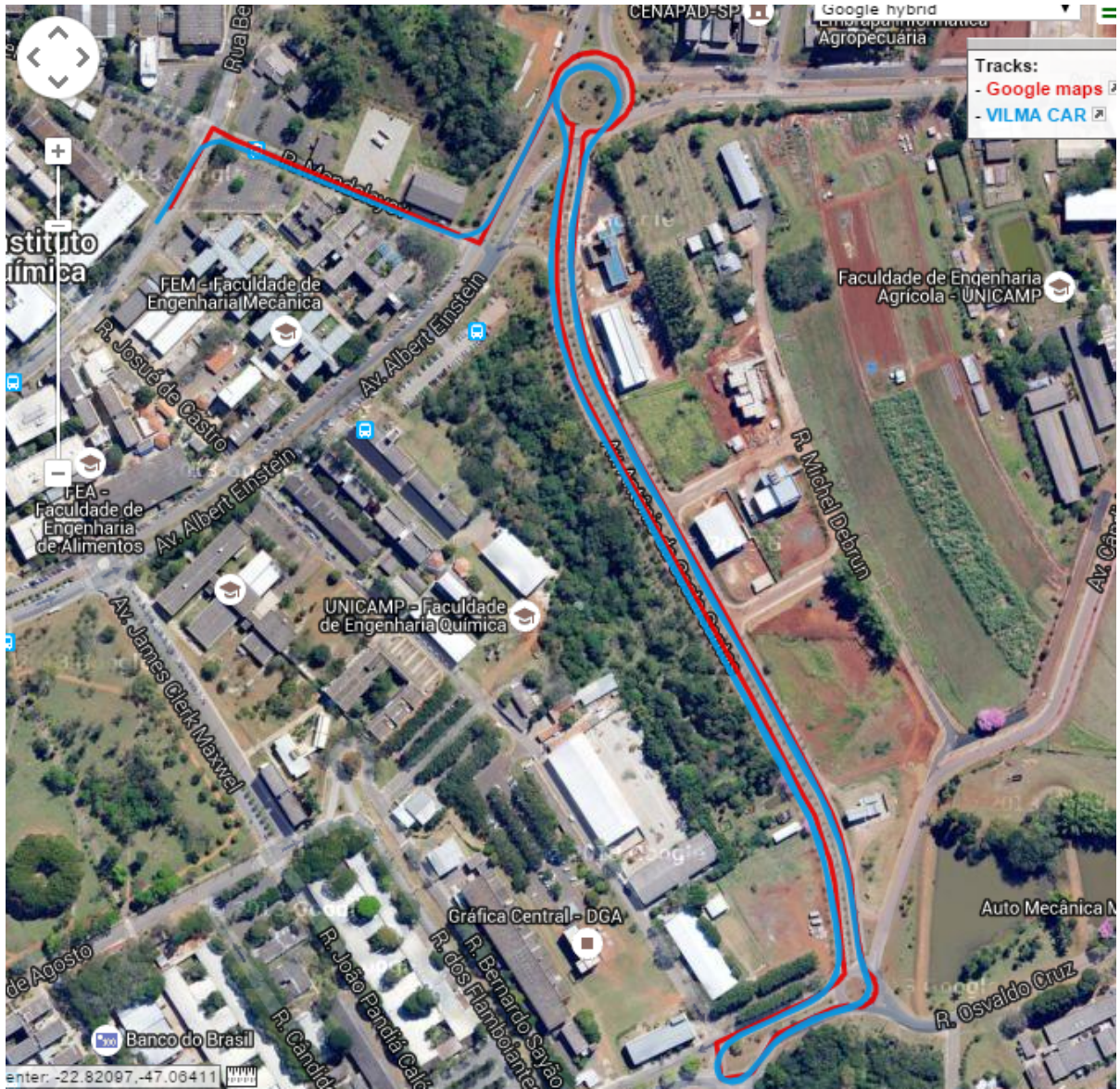


Figura 6.10: Comparação entre o gps e rota planejada pelo Google Maps no recorrido dois.

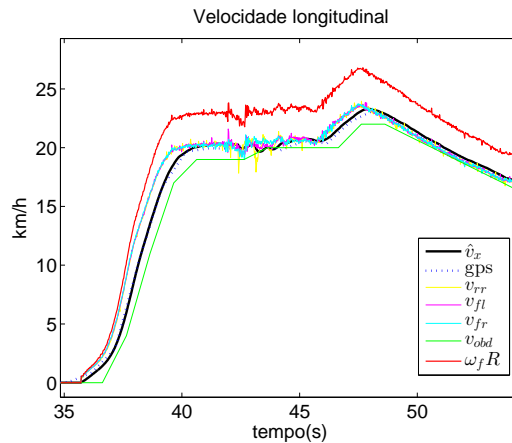


Figura 6.11: Sensores de velocidade do VILMA

6.1.5 Estimação da velocidade longitudinal do veículo

Para a estimação da velocidade longitudinal do veículo tem os sensores de velocidade de cada uma das rodas enviada pelo sistema de freio pela rede CAN, a velocidade reportada pela porta OBD(on-board protocol) do veículo e a velocidade estimada pelo GPS. Na figura 6.11 é apresentado um intervalo de medição dos sensores. Nesta figura é possível ver que a média da velocidade angular das rodas frontais (enviada pelo sistema do freio) multiplicado pelo raio teórico da roda (referencia *195/60 R15 88H*) tem uma relação de 0.93 com a velocidade do veículo o que pode ser causado pelo raio efetivo (DOUMIATI, 2009) do pneu. Adicionalmente foi possível encontrar que existe uma defasagem entre as velocidades das rodas e as reportadas pelo GPS e pelo sensor OBD a qual pode ser explicado pela inércia do veículo, pela dinâmica longitudinal do veículo que gera um deslizamento, pelo sistema de suspensão do veículo que gera um tempo de resposta entre a força gerada na iteração da roda com o solo e a força percebida pelo chassis do veículo.

Para a estimação da velocidade longitudinal do veículo foram utilizadas a inversa das equação (6.5) (DOUMIATI *et al.*, 2010) através da pseudoinversa da matriz D .

$$\begin{bmatrix} v_{rl} \\ v_{rr} \\ v_{fl} \\ v_{fr} \end{bmatrix} = \begin{bmatrix} 1 & \frac{-E}{2} & 0 \\ 1 & \frac{E}{2} & 0 \\ 1 & \frac{-E}{2} \cos(\delta_f) + l_f \sin(\delta_f) & \sin(\delta_f) \\ 1 & \frac{E}{2} \cos(\delta_f) + l_f \sin(\delta_f) & \sin(\delta_f) \end{bmatrix} \begin{bmatrix} v_x \\ \gamma \\ v_y \end{bmatrix} = D \begin{bmatrix} v_x \\ \gamma \\ v_y \end{bmatrix} \quad (6.5)$$

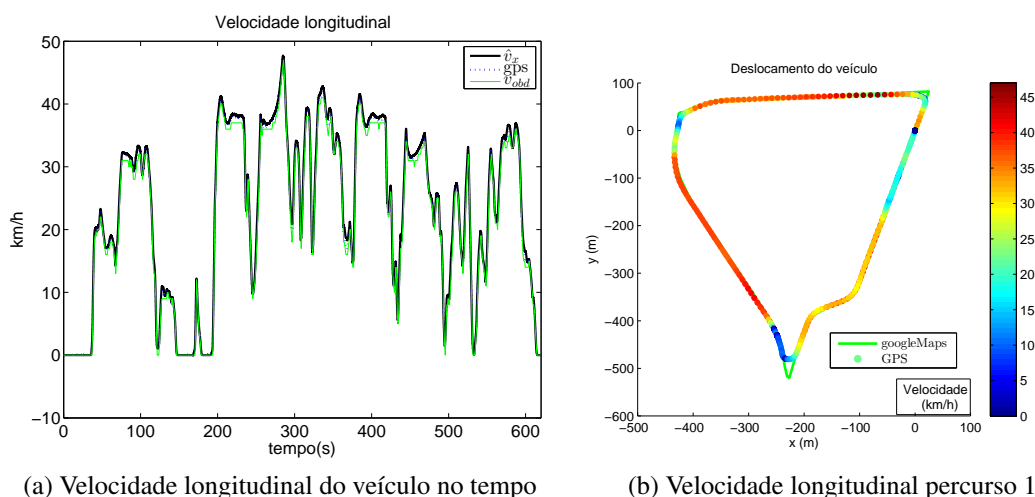


Figura 6.12: Velocidade longitudinal estimada percurso 1

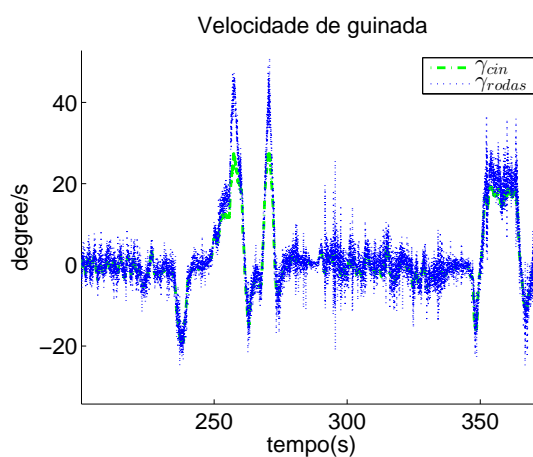
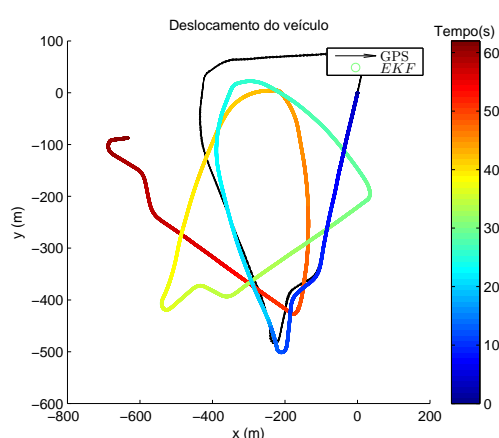


Figura 6.13: Velocidade angular estimada com a velocidade das rodas

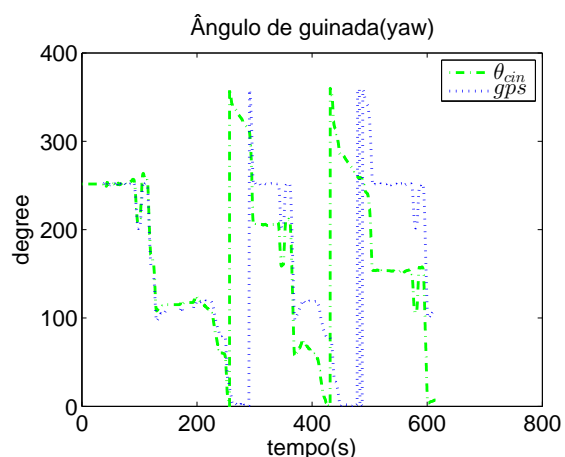
A velocidade longitudinal estimada é filtrada com um passa baixo com $\tau = 0.4s$ procurando modelar a defasagem. Nas figuras 6.12a e 6.12b é apresentada a velocidade longitudinal estimada no tempo e a velocidade longitudinal calculada pelo GPS durante o percurso 1. Finalmente na figura 6.13 é apresentada a velocidade angular do veículo estimada a partir da velocidades das rodas comparada com a calculada pelo modelo cinemático do veículo.

6.1.6 Modelo cinemático do veículo

A partir do modelo cinemático descrito no apêndice pelo sistema de equações (G.9) foi feito a integração pelo método de Runge-Kutta obtendo como resultado as figuras 6.14a e 6.14b. Na primeira figura é possível observar que o maior erro, que se acumula porque não tem nenhuma correção é gerado em uma pequena curva na reta perto do laboratório. Este erro pode ser

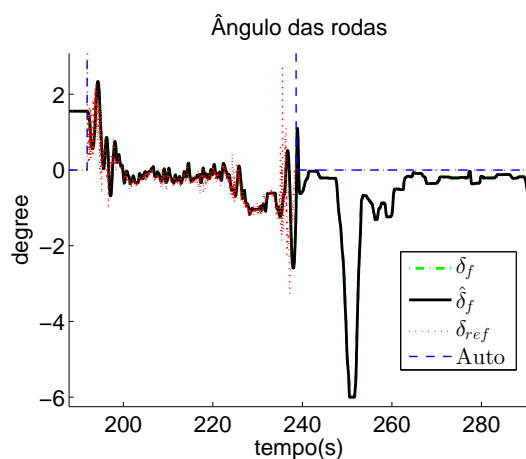


(a) Modelo cinemático do veículo

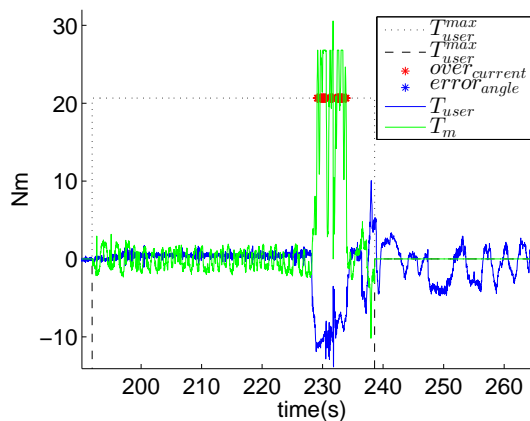


(b) Ângulo de guinada do veículo estimado pelo modelo cinemático

Figura 6.14: Modelo cinemático para o percurso 1



(a) Ângulo das rodas estimado



(b) Torque do motor de direção

Figura 6.15: Problema no ângulo das rodas por controle cooperativo

explicado porque o ângulo do volante (medido) com respeito ao ângulo da roda(estimado) tem uma zona morta ao redor de zero que não está sendo modelada e porque nesta etapa do percurso ocorreu uma sobre corrente no motor (pelo controle cooperativo) que por falta de uma chaveta no motor de direção e a zona morta da caixa de engrenagem pela torção gerou um deslizamento entre o ângulo medido (do motor) e o ângulo estimado (do volante). A falha descrita pode ser observada nas figuras 6.15a e 6.15b. Note-se que este erro tem um agravante dado que a velocidade angular depende da velocidade longitudinal a qual nesta zona do percurso (figura 6.12b) é de aproximadamente 40kmh .

6.1.7 Localização com o filtro de Kalman Estendido (EKF)

Com o objetivo de fusionar os sensores proprioceptivo (giroscópio, acelerômetros, velocidades das rodas) com o sensor exteroceptivo (GPS) para estimar a localização do veículo foi aplicado um filtro EKF de acordo com a metodologia explicada na seção 3.3 com o modelo dinâmico do veículo, mas sem o modelo dinâmico da direção pelo problema descrito na seção anterior. Com a equação (G.17) e $\dot{a}_0 = 0$ foi criado o modelo de estado desejado para o filtro EKF, utilizando como estados $\hat{\mathbf{x}}_{\mathbf{k}} = \left[\hat{v}_y \quad \hat{\gamma} \quad x \quad y \quad \theta \quad a_0 \right]^T$, vetor de entrada $\mathbf{u}_{\mathbf{k}} = [\delta_f]^T$ e vetor de medidas $\mathbf{z} = \left[a_y + a_o \quad \gamma_{rodas} \quad x_{gps} \quad y_{gps} \quad \theta_{gps} \right]^T$. Note-se que a_0 é a componente de baixa frequência do sensor de aceleração lateral que tem como resultado o apresentado na figura 6.6a que modela o componente da gravidade no acelerômetro. Como parâmetros do modelo dos pneus são utilizados os apresentados no anexo de (YIH E GERDES, 2005) e os cálculos da massa e da inércia do veículo, cujos valores foram estimados por outro estudante de doutorado do departamento no LabeDin. Os ruídos do modelo e dos sensores são descritos na equação (6.6) e (6.7)

$$\begin{aligned} \sigma_{\mathbf{vk}} &= \left[\sigma_{a_y} \quad \sigma_{\gamma} \quad \sigma_{x_{gps}} \quad \sigma_{y_{gps}} \quad \sigma_{\theta_{gps}} \right]^T \\ &= \left[(0.2m/s)^2 \quad (4^\circ/s)^2 \quad (1m)^2 \quad (1m)^2 \quad (1^\circ)^2 \right]^T \end{aligned} \quad (6.6)$$

$$\begin{aligned} \sigma_{\mathbf{wk}} &= \left[\sigma_{f_r} \quad \sigma_{f_y} \quad \sigma_{\dot{x}} \quad \sigma_{\dot{y}} \quad \sigma_{\dot{\theta}} \quad \sigma_{\dot{a}_0} \right]^T \\ &= \left[(50N)^2 \quad (50N)^2 \quad (0.5m/s^2)^2 \quad (0.5m/s)^2 \quad (0.3rad/s)^2 \quad (0.3m/s^3)^2 \right]^T \end{aligned} \quad (6.7)$$

Note-se que os valores de $\sigma_{\mathbf{vk}}$ modelam o desvio padrão dos sensores obtidos a partir da observação dos sinais adquiridos. Por outro lado como o modelo simplificado do veículo não tem sido identificado corretamente as covariâncias $\sigma_{\mathbf{wk}}$ das entradas estocásticas foram determinadas heurísticamente procurando ter valores congruentes com as unidades de cada uma.

Na figura 6.16 é apresentado o ângulo das rodas estimado a partir do raio de giro mínimo do veículo versus o ângulo do volante assumindo uma relação linear e o modelo geométrico do veículo (SNIDER, 2009). Nesta figura também são apresentados adicionalmente os períodos nos quais o controle de seguimento das linhas no percurso um está ativado. O resultado da estimação do percurso um é apresentado na figura 6.17 na qual pode-se apreciar o mesmo erro do modelo cinemático ao redor dos 220s (cor *cyan*) o qual é corrigido na segunda volta (cor

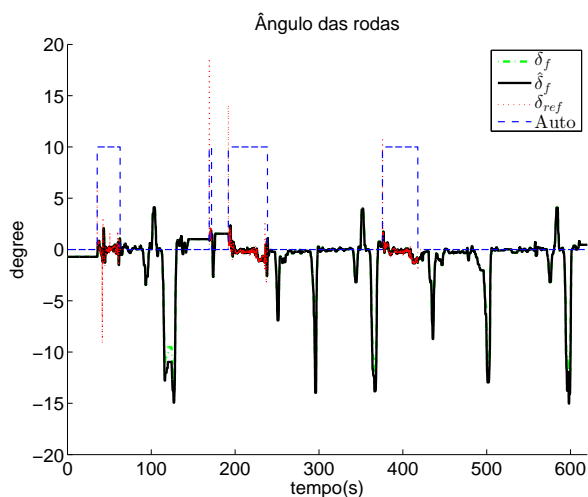


Figura 6.16: Ângulo das rodas estimada do percurso um

amarelo) devido a correção do estimador de estados.

Finalmente na figura 6.18 são apresentadas as variáveis de estado do ângulo e da velocidade de guinada, assim como da velocidade lateral estimadas. Note-se que o sensor de orientação a partir da velocidade do GPS e o sensor de aceleração lateral somente são utilizados quando a velocidade do veículo é maior a 1.5m/s. Na figura 6.18d é amplificado a dupla curva na zona de descida do veículo para enfatizar que o modelo tem um problema que quando o veículo gira à direita sub estima a velocidade de guinada, mas quando gira à esquerda existe uma sobre estimação da velocidade de guinada, o qual pode ser explicado por um erro nos parâmetros das forças das rodas (maior peso a esquerda que a direita), necessidade de incluir o eixo de inclinação no modelo ou da estimação do ângulo das rodas (note-se na figura 6.13 que um erro similar ocorre com o modelo cinemático). Tentando solucionar o problema foi usada a fórmula de Ackerman (CORDEIRO *et al.*, 2012) para calcular a força lateral de cada uma das rodas dianteiras, mas isso não solucionou o problema.

6.1.8 Recomendações

Na instrumentação do veículo é necessário incluir um sensor do ângulo da direção do veículo, assim como ter um modelo da dinâmica longitudinal e testar o modelo da dinâmica da direção. Para melhorar a estimação dos ângulos de rolagem, guinada e inclinação do veículo é recomendado ter uma central inercial que permita incluir as velocidades angulares em todos os eixos, e um processo de identificação de parâmetros dos modelos do veículo. Os ângulos de rolagem e inclinação aferidos devem ser incluídos no modelo de dinâmica planar para melhorar

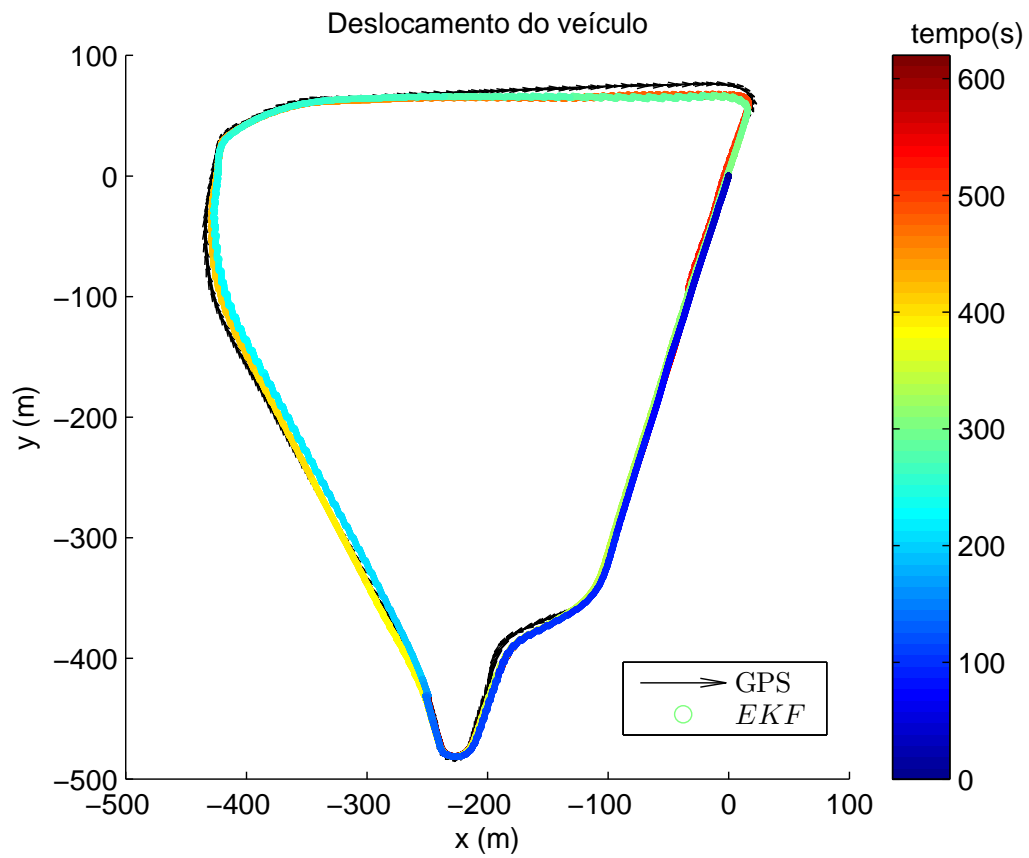


Figura 6.17: Estimação da posição do EKF com o modelo dinâmico

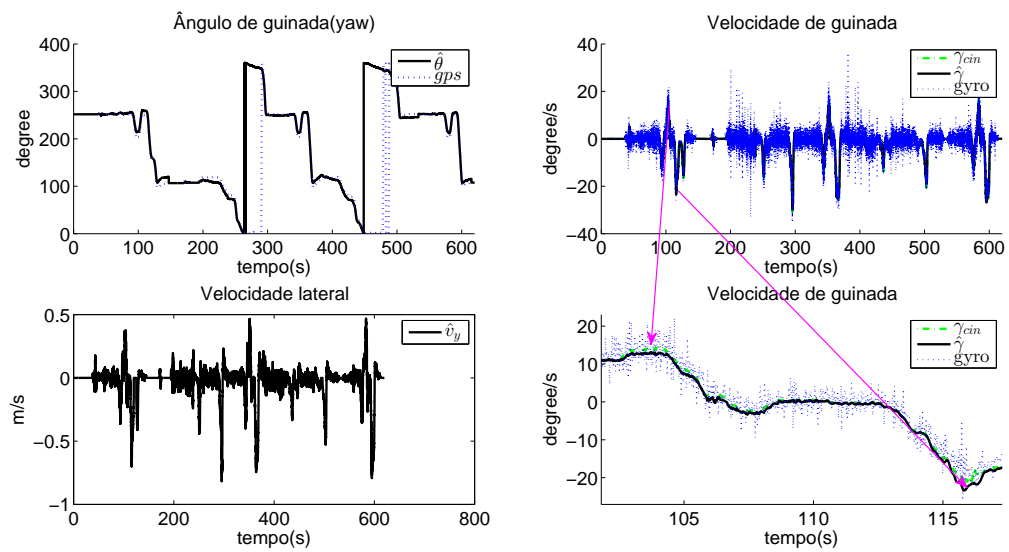


Figura 6.18: Variáveis de estado estimadas do percurso 1

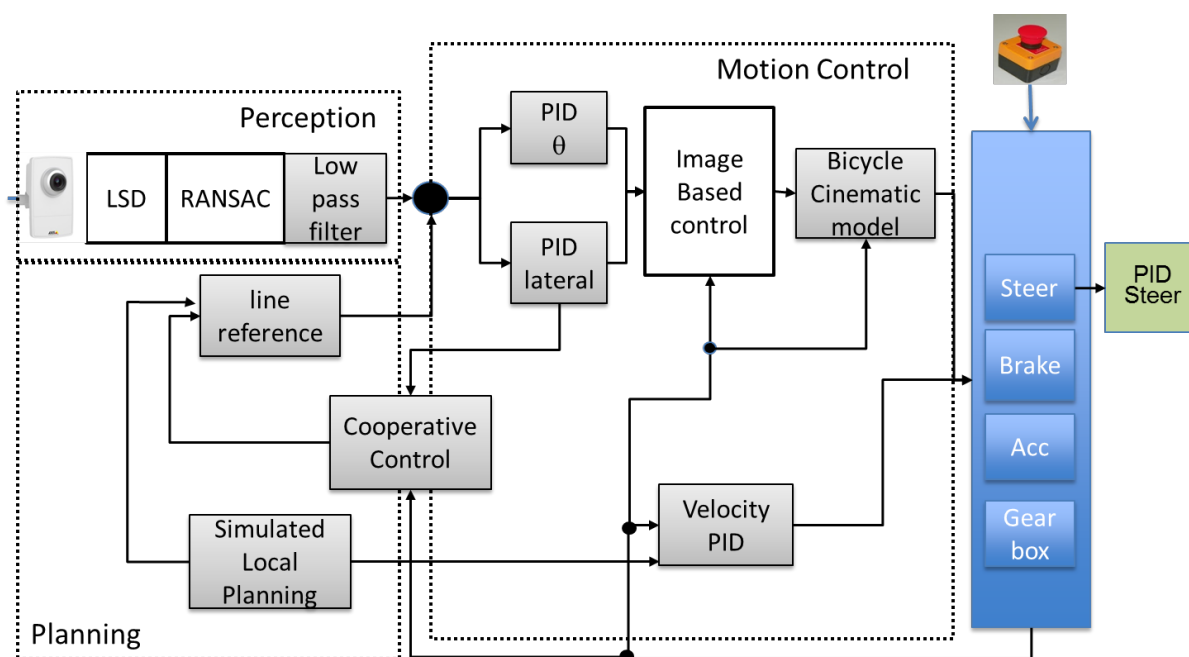


Figura 6.19: Diagrama de bloco do controle de seguimento de linhas da estradas

a estimação da posição global através do vetor de aceleração v_{xw} . No momento a transformação de latitude/longitude em metros da medida do GPS é feito pelo sistema de modelo de referência global WGS84(RODRIGUEZ RUIZ, 2014) com o pacote *Geodesy* do ROS, mas é necessário revisar os parâmetros do mesmo para serem adaptados à localização brasileira.

6.2 Controle de trajetórias do VILMA01 baseado em imagens

Nesta seção será apresentado a implementação do controle de trajetórias para seguir as linhas da estrada implementado como um nó de ROS realizado com a finalidade de testar os conceitos de condicionamento de controle e planejamento cooperativo. Na figura 6.19 é apresentado um diagrama de blocos do controle de trajetórias desenvolvido e implementado.

A seguir será descrita a implementação de cada uma dos blocos dividido nas camadas: Percepção, controle, planejamento e controle cooperativo. Finalmente serão apresentados alguns resultados experimentais.

6.2.1 Percepção

Para a detecção da linha da estrada, primeiro foi feito um algoritmo de detecção de linha na camada de percepção, utilizando a técnica de detecção de linhas segmentadas (GROMPONE VON GIOI *et al.*, 2012) o qual gera umas retas que são classificadas e fusionadas para obter uma linha só, através de RANSAC (*RANdom SAmple Consensus*)(SIEGWART *et al.*, 2011a)(FISCHLER E BOLLES, 1981) usando a biblioteca PCL (RUSU E COUSINS, 2011). A linha resultante é descrita pela equação (6.8), dado que m^{image} é a pendente utilizada para encontrar o erro de orientação e b^{image} utilizado para encontrar o erro lateral em um valor fixo de y^{ref} (no sistemas de coordenadas da imagem) determinado heurísticamente para ter um compromisso entre a estabilidade e a suavidade da resposta (SNIDER, 2009).

$$x^{image} = m_k^{image} y^{ref} + b_k^{image} \quad (6.8)$$

Os parâmetros da reta são filtrados com um filtro passa baixa de primeira ordem descrito pela equação (6.9).

$$\begin{aligned} \dot{m}^{image} &= \frac{1}{T_{lpm}} (-m^{image} + m^{RANSAC}) \\ \dot{b}^{image} &= \frac{1}{T_{lpb}} (-b^{image} + b^{RANSAC}) \end{aligned} \quad (6.9)$$

O filtro discretizado por Euler é implementado pela equação (6.10)

$$\begin{aligned} m_k^{image} &= m_{k-1}^{image} (1 - \alpha_m) + \alpha_m m^{RANSAC} \\ b_k^{image} &= b_{k-1}^{image} (1 - \alpha_b) + \alpha_b b^{RANSAC} \end{aligned} \quad (6.10)$$

dado que $\alpha_m = T_i/T_{lpm}$, $\alpha_b = T_i/T_{lpb}$ e T_i é o tempo de ciclo entre cada imagem.

6.2.2 Planejamento

O planejamento neste experimento foi simulado através de um *gamepad*. O planejamento da velocidade foi feito através de velocidades constantes que são modificadas por um sinal analógico gerado pelo *gamepad*, limitados com aceleração/desaceleração constante. A referência do controle de trajetórias é descrita por uma linha criada diretamente na imagem através da equação (6.11).

$$x^{ref} = m_k^{ref} y^{ref} + b_k^{ref} \quad (6.11)$$

Com a finalidade de testar o planejamento cooperativo foram usados dois botões do *gamepad* para simular o risco de variar referência lateral à direita ou à esquerda. O motorista poderá variar o sinal de referência através do sinal do volante do veículo somente se o planejamento (os botões) estiver ativo, variação esta que será governada pela equação (6.12).

$$b_k^{ref} = b_{k-1}^{ref} + \beta_1 \tau^{user} + \beta_2 \quad (6.12)$$

dado que β_1 e β_2 são parâmetros heurísticos em *pixels/Nm* e *pixels* respectivamente encontrados heurísticamente e que k varia com o período de amostragem do sensor do torque. Note-se que quando o condutor libera a direção, a referência volta a ser a predeterminada inicialmente ($b_k^{ref} = b_0^{ref}$) através de uma rampa para suavizar o esforço de controle, ou seja para suavizar a curvatura feita pelo veículo com a finalidade de aumentar a segurança intrínseca do mesmo.

6.2.3 Controle de movimento

A figura 6.20 apresenta as linhas na imagem que são necessárias para o controle, a amarela é a linha detetada pelo processamento da imagem (seção 6.2.1), a roxa é a linha detetada filtrada no eixo temporal pelo filtro da equação 6.10 e a branca é a linha de referência do controle da equação 6.11. O objetivo final do controle é que as linhas brancas e roxas sejam coincidentes em orientação e o ponto de referencia y_{ref} do erro lateral seja zero ($x_{ref} = y_{ref}$).

Para atingir esta estratégia de controle, na camada de controle foi implementado um controlador baseado em imagem (CHERUBINI *et al.*, 2011). Ele se chama desta forma porque traba-

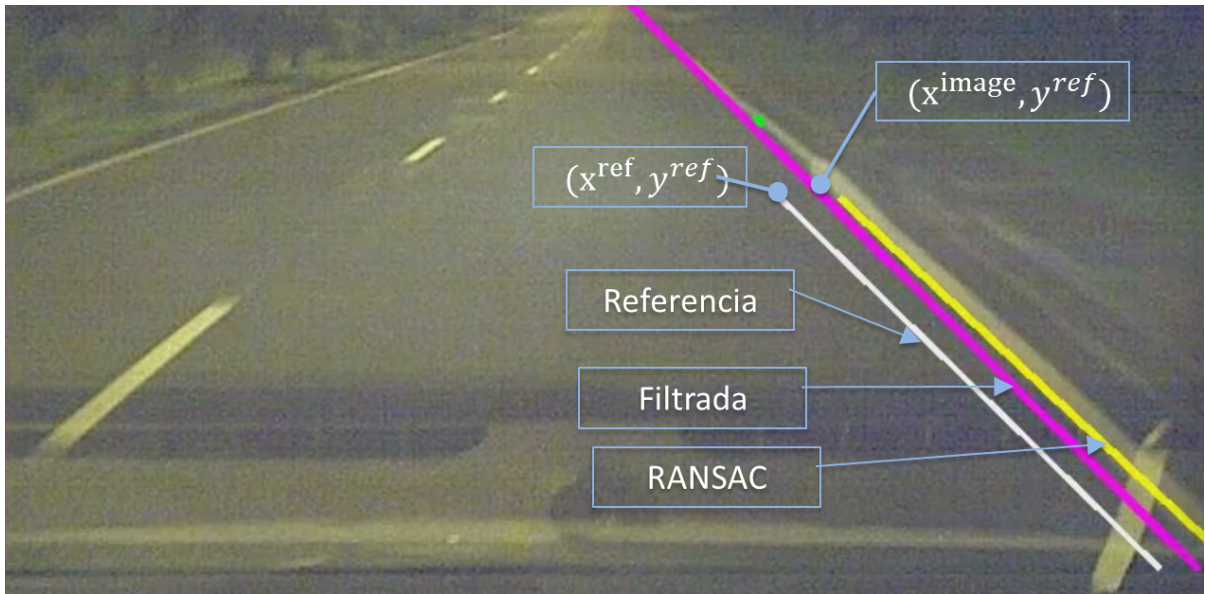


Figura 6.20: Controle Baseado em imagem

lha com erros no sistema de coordenadas da imagem e gera a velocidade angular (γ) necessária para um veículo não holonômico através de um modelo de câmera generalizada. A velocidade angular calculada é transformada em um ângulo de direção através do modelo cinemático do veículo (seção G.1). Para que os erros da trajetória e da velocidade converjam a zero, três controladores PID foram implementados, um para o erro de orientação (e_θ), outro para o erro lateral (e_l) e um para a velocidade do veículo (e_{v_x}). Os dois primeiros sinais de controle são somados a partir das equações apresentadas em (CHERUBINI *et al.*, 2011) para comandar o ângulo da direção e o terceiro controlará o freio e o acelerador do veículo.

Os PID implementados são descritos pela equação (6.13) em contínuo e pela equação (6.14) em forma discreta através de Euler.

$$u^{\{e\}} = k_p e + k_i \int e \cdot dt + k_d \dot{e} \quad (6.13)$$

$$u_k^{\{e\}} = k_p e_k + \frac{k_d}{T_i} (e_k - e_{k-1}) + k_p T_i \sum_{i=0}^k e_{k-i} \quad (6.14)$$

nas quais e é erro respectivo $e_\theta = \text{atan}(m^{\text{ref}}) - \text{atan}(m_k^{\text{image}})$, $e_l = x_k^{\text{ref}} - x_k^{\text{image}}$ ou $e_{v_x} =$

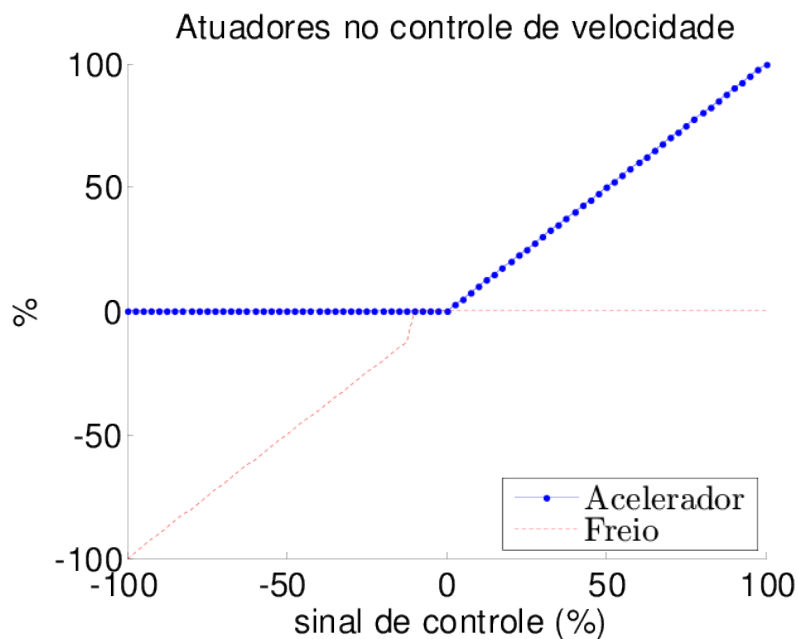


Figura 6.21: Zona morta no controle de velocidade do veículo

$v^{ref} - v_x$ e $u^{\{e\}}$ é o valor do sinal de controle. Os controladores apresentam não linearidades, uma é que em todos os controladores o integrador é saturado a um determinado valor, outra é que o controlador de velocidade tem uma zona morta na saída (ver figura 6.21) para que a frenagem seja suave e só seja utilizado quando realmente for necessário.

6.2.4 Controle Cooperativo

A primeiro teste de controle cooperativo foi desenvolvido como medida de segurança e consiste que em quando o condutor pressione o pedal do freio o controle de velocidade seja desativado, mas o controle de trajetória continua funcionando normalmente a menos que o condutor pressione o botão de emergência, momento no qual todos os controles serão desativados. Para a cooperação no controle de trajetórias, definimos controle cooperativo em que o condutor ajude a convergir o erro lateral a zero, ou em outras palavras ajude a seguir a linha da estrada.

Durante as análises e os testes em simulação feitos em uma arquitetura de controle de trajetórias de duas camadas (GARCIA *et al.*, 2014), com o PID utilizado no firmware do motor, foi possível perceber que qualquer torque gerado pelo condutor vai ser tratado como uma perturbação para o PID da direção, não interessando se ele está na direção que diminui o erro lateral. O mesmo comportamento foi encontrado nos testes experimentais do controlador baseado em imagem. Portanto a estratégia adotada foi variar o sinal de referência b_k^{ref} com uma velocidade

dependente do torque do usuário se e somente se o torque aplicado pelo condutor tiver a mesma direção do erro e os dois estiverem fora de uma zona morta. O sinal de referência será governado pela equação (6.12).

Portanto o controle cooperativo realmente troca de forma temporal a rota planejada superpondo os comandos do planejamento nos casos em que vai ajudar a diminuir o erro lateral. O algoritmo 4 descreve a implementação da proposta.

```

if (left button or  $\text{sgn}(\tau^{user}) = \text{sgn}(e_l)$ ) and  $|\tau^{user}| > \tau_{min}$  then
  |  $b_k^{ref} \leftarrow b_{k-1}^{ref} + \beta_1 \text{sgn}(\tau^{user}) + \beta_2$ 
else if (right button or  $\text{sgn}(\tau^{user}) = \text{sgn}(e_l)$ ) and  $|\tau^{user}| > \tau_{min}$  then
  |  $b_k^{ref} \leftarrow b_{k-1}^{ref} - \beta_1 \text{sgn}(\tau^{user}) + \beta_2$ 
else
  | if  $\text{abs}(b_k^{ref} - b_0^{ref}) > \Delta_b$  then
  | |  $b_k^{ref} \leftarrow b_{k-1}^{ref} - \text{sign}(\tau^{user})\Delta_b$ 
  | else
  | |  $b_k^{ref} \leftarrow b_0^{ref}$ 

```

Algoritmo 4: Algoritmo de planejamento cooperativo no controle baseado em imagens

6.3 Resultados do VILMA01 com condicionamento de controle cooperativo

Devido a que a marcação das linhas da estradas estão em perfeito estado na maioria do percurso dois (figura 6.10) este foi o selecionado para realizar os testes de controle cooperativo. Na figura 6.22a são mostrados os intervalos do percurso onde o veículo tinha ativado a função de modo autônomo e a figura 6.22b apresenta o perfil de velocidades no mesmo percurso.

Na figura 6.23 é apresentado o comportamento do sinal de controle (o ângulo do motor da direção) no percurso dois. Dela é possível corroborar que o controle do motor de direção em modo autônomo funciona corretamente. Um dos problemas encontrados nesta camada de controle é que o motor está configurado em faixas de velocidades de rotação angulares baixas de acordo as desejadas pelo sinal de controle, porém as velocidades da direção são similares as utilizadas pelo condutor em situações de condução normal, portanto é necessário na camada de controle de trajetórias realizar um melhor processo de filtragem do ruído. Outro problema esta relacionado ao discutido na seção anterior que o zero da direção não é conhecido com precisão e a dinâmica do volante não está sendo incluída.

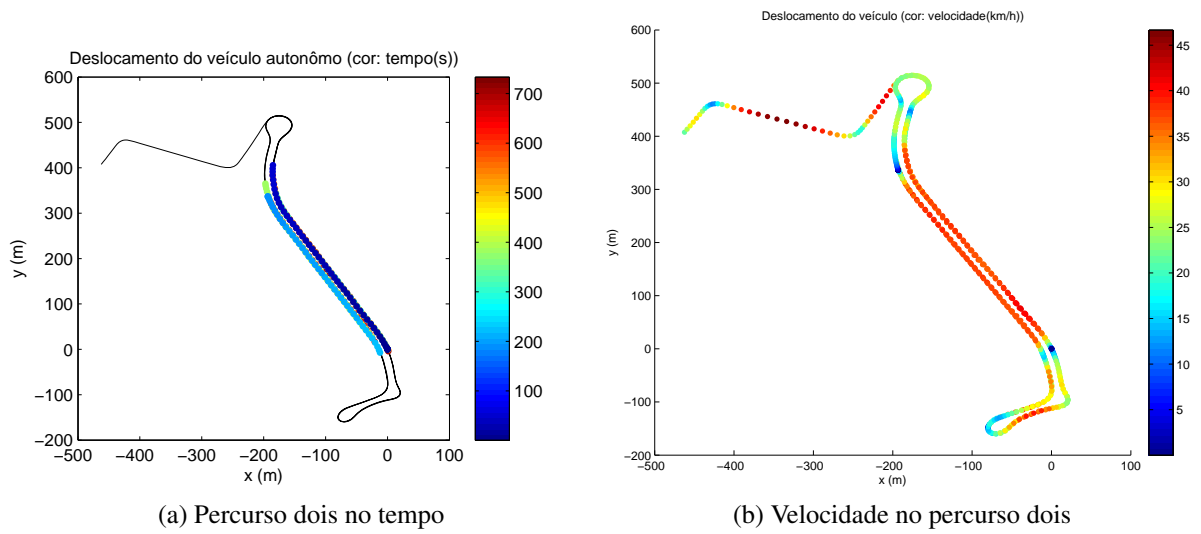


Figura 6.22: Tempo e velocidade do percurso dois

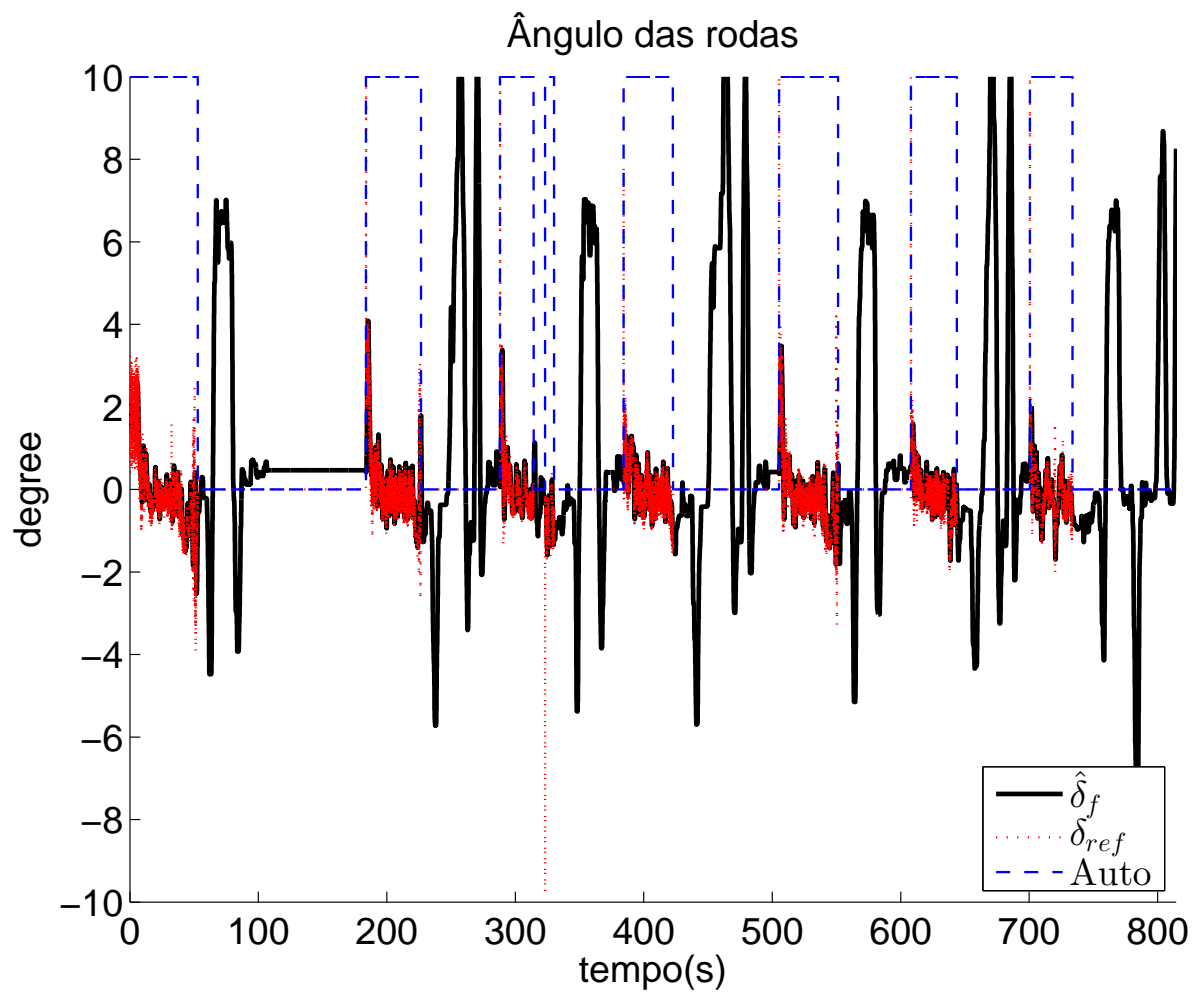


Figura 6.23: Ângulo estimado das rodas no percurso dois

6.3.1 Teste do Controle de Velocidade longitudinal

Na Figura 6.24a é apresentada a velocidade do percurso dois, a qual é comandada por uma velocidade de referência de 40km/h que é obtida com uma aceleração de 3m/s^2 integrada por Euler. O controle de velocidade só inicia quando é liberado o pedal do freio do condutor (linha preta da figura 6.24b). Neste tempo (5-6s) pode-se ver que existe um atraso entre o sinal de referência e a velocidade longitudinal (estimada através da velocidade das rodas). Este atraso pode ser proveniente de diferentes fontes: do sensor de velocidade que só inicia a gerar valores a partir de aproximadamente 2km/h ; da dinâmica longitudinal do veículo que não está sendo considerada; do tempo de resposta do motor do veículo que também não foi considerado; e da existência de uma zona morta do sinal do pedal do acelerador. Note-se que a zona morta do sinal do pedal do acelerador acontece principalmente pela diferença que existe entre o sinal gerado pelo sistema automático, que modela a curva ideal dos potenciômetros do pedal do acelerador (ver apêndice 7), e a curva real dos potenciômetros do pedal do acelerador (ver acc_{user} depois dos 50s).

Sobre a estratégia de segurança implementada, pode-se observar ao redor de 50s que quando o pedal do freio é pressionado pelo condutor o controle de velocidade longitudinal é desligado (acc é levado a zero) até quando o pedal é liberado. Neste momento uma nova curva de referência é gerada para o veículo voltar à velocidade planejada no sistema automático. No controle de velocidade longitudinal foi percebido que é necessário nos trabalhos futuros compensar a força da gravidade gerada pelo ângulo de inclinação do veículo através de um controle *feedforward*, dado que as acelerações não conseguem ser atingidas com o mesmos parâmetros do controlador nas descidas e subidas.

6.3.2 Teste de condicionamento de controle cooperativo

Na figura 6.25a é apresentado o comportamento do erro lateral no início do percurso 2. Nela é possível ver como depois dos 6 segundos, que o veículo inicia o seu percurso, o erro lateral se aproxima de zero e fica oscilando ao redor de zero, uma vez que existem erros de detecção na imagem causados principalmente pela distorção radial da câmera. Entre os 15 e 20 segundos na figura 6.25c é possível ver como o condutor aplica dois torques no volante. O primeiro torque (15 – 16s) no sentido contrario do erro lateral faz com que o motor tente evitar o desejado pelo condutor, como pode ser visto na figura o torque do motor é máximo e em sentido contrario do sinal de torque do condutor. O segundo torque no sentido do erro

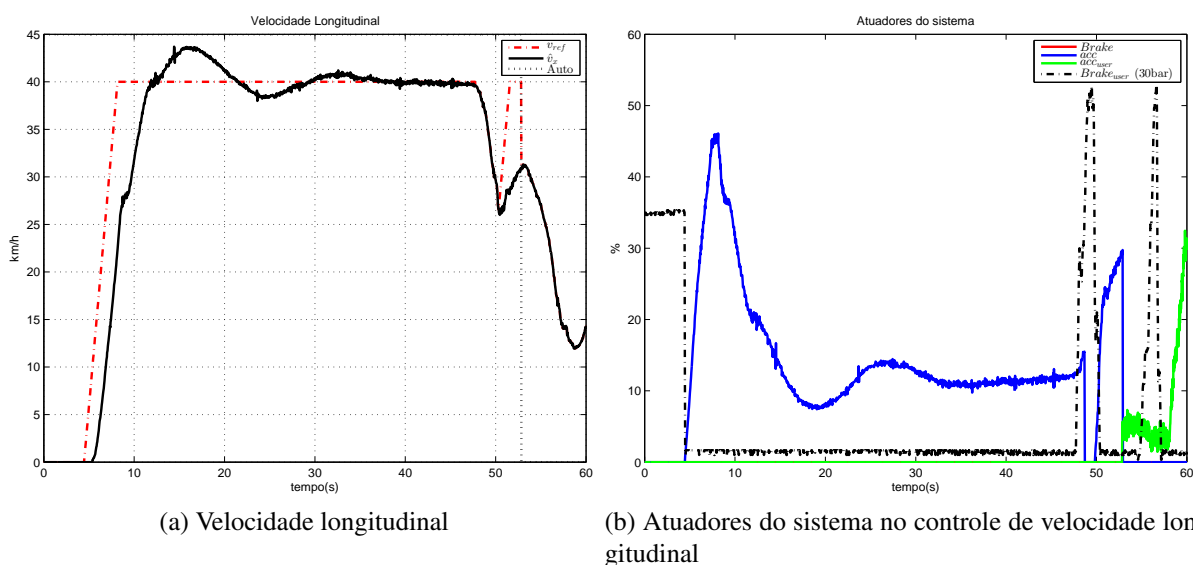


Figura 6.24: Controle de velocidade longitudinal

lateral faz com que o torque do motor continue baixo, para que isso seja possível o erro lateral do controlador é modificado (linha azul figura 6.25a) fazendo com que b_k^{ref} seja acrescentado levemente para variar a velocidade angular requerida no veículo. Note-se na figura 6.25b que no erro de orientação o comportamento cooperativo não gera trocas perceptíveis. Entre os 40 e 50s pode-se observar outro teste de controle cooperativo que ocorre num momento onde dada a velocidade do veículo (perto de 40km/h de acordo a figura 6.22b) e as condições do algoritmo de detecção da estrada (ele deteta linhas e não curvas) faz com que o controle autônomo sozinho não consiga seguir a estrada (erro lateral grande), porém com a ajuda do condutor foi possível fazer a curva com erro lateral mínimo.

6.3.3 Teste de condicionamento de planejamento cooperativo

Na figura 6.26a é apresentado o erro lateral num teste de planejamento cooperativo onde o condutor quer se afastar lateralmente da faixa da estrada. Neste caso do ponto de vista do controle cooperativo não seria permitido, mas dado que o planejamento estimou que o risco é mínimo (pressionamento dos botões do *gamepad*), o sistema automático modifica o erro lateral para fazer com que o veículo fique mais longe da faixa da estrada. Nesta estratégia de acordo com a figura 6.26c não foi possível obter torque baixos no motor em toda a etapa cooperativa, causado por duas causas principais. A primeira porque o sinal de referência não está sendo trocado de forma correta através de uma estimação da trajetória desejada do condutor pelas limitações na estimação da mesma. A segunda causa ocorre pela estratégia de controle imple-

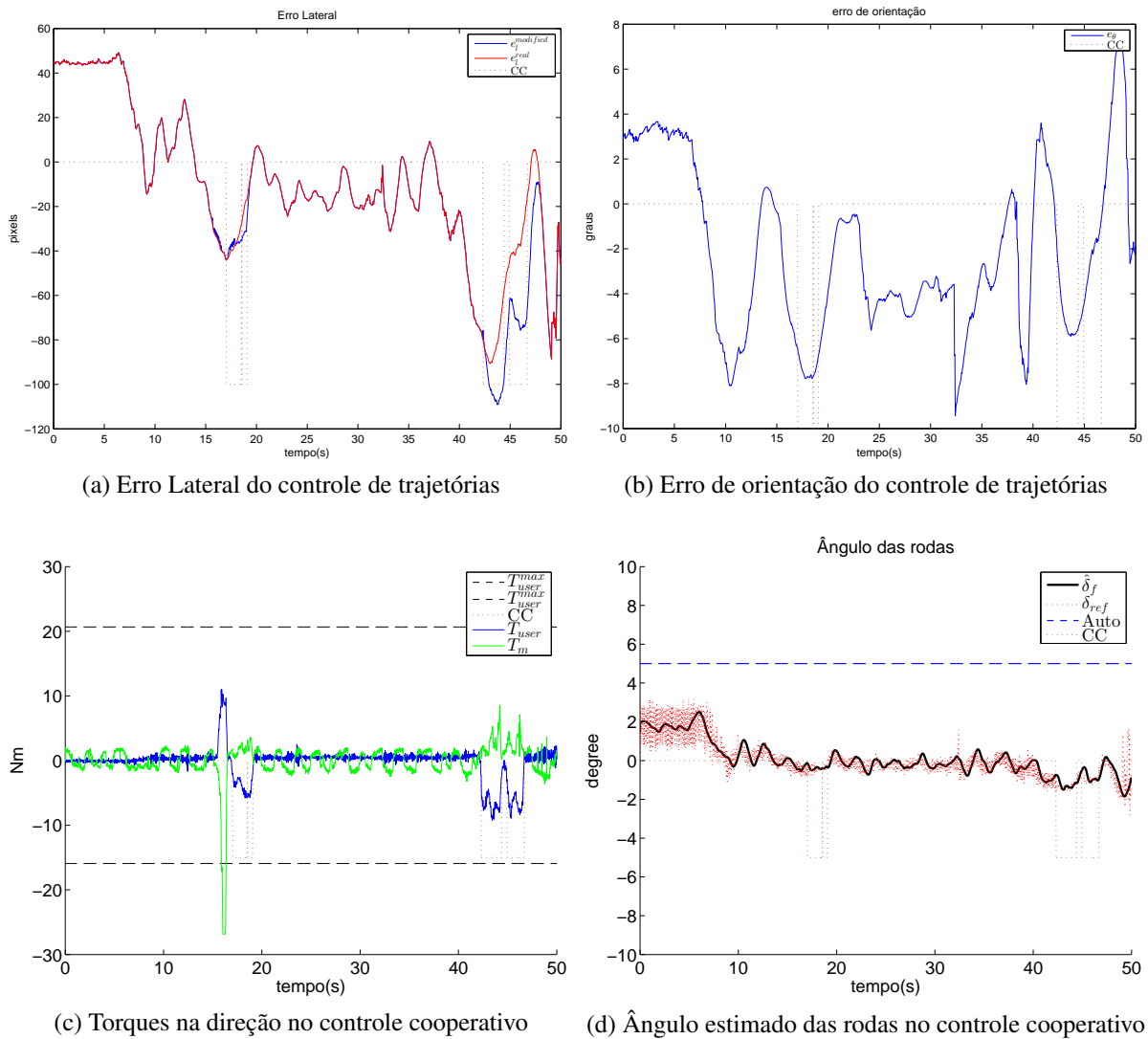


Figura 6.25: Controle de trajetória cooperativo no percurso 2

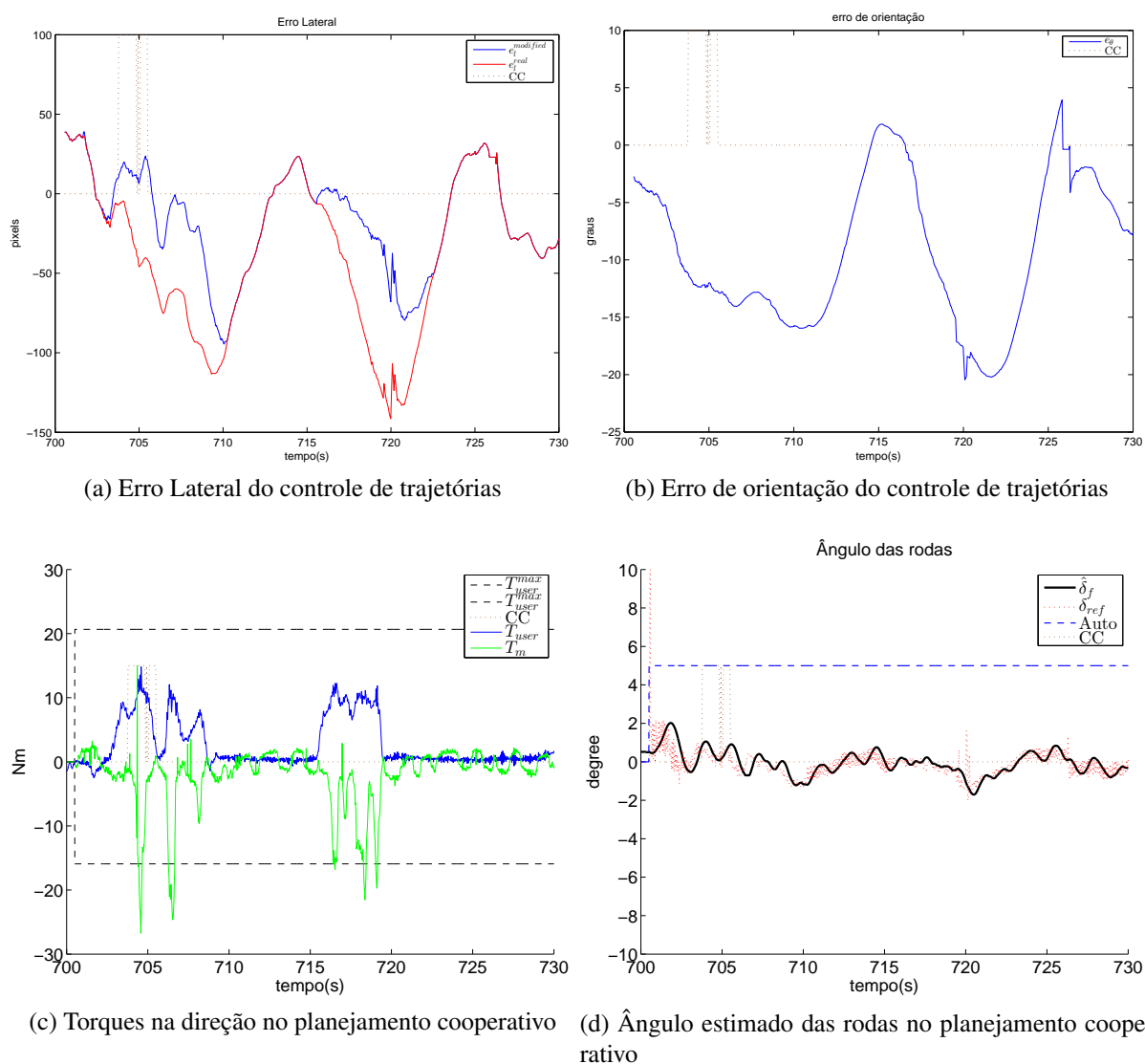


Figura 6.26: Controle de trajetória cooperativo no percurso 2

mentada, dado que como pode ser observado na figura 6.26a que embora o erro lateral esteja sendo modificado este continua sendo contrario a sinal de torque do condutor, o que gera que a derivada do ângulo da direção (figura 6.26d) que afeta o torque do motor seja contrário ao torque do condutor nesse instante de tempo.

Na figura 6.26b pode-se ver que o erro de orientação fica longe zero nestos casos, isso acontece porque ao trocar o erro lateral na direção contraria ao ângulo de orientação de referencia também deve ser modificado.

7 Conclusão

Foi mostrado que é possível fazer o condicionamento do controle cooperativo com análises de risco utilizando a arquitetura de veículo autônomo desenvolvida com mínimos custos computacionais. As análises de risco permitem prever possíveis acidentes do condutor antes de realizar a transferência de controle. O método proposto integra as camadas de planejamento e controle de trajetórias do veículo autônomo. A camada de controle foi implementada com a técnica de controle preditivo baseado num modelo dinâmico do veículo e da direção. Nesta camada foi incluído o torque do condutor para achar um sinal de controle cooperativo ótimo e ainda com os mesmos cálculos prever a tendência do condutor num horizonte futuro, chamado horizonte dinâmico. Na camada de planejamento, através de um espaço curvilíneo que modela o mapa ou cenário onde se encontra o veículo são calculados alguns caminhos candidatos com o modelo cinemático do veículo sobre os quais é selecionado o caminho ótimo com respeito aos riscos intrínsecos e extrínsecos do veículo. Nesta etapa a predição da tendência do condutor, chamada horizonte cinemático, gera uma trajetória prevista utilizando o horizonte dinâmico, o modelo cinemático do veículo e o modelo do cenário (espaço curvilíneo). Esta trajetória é relacionada com os caminhos candidatos e com as análises de risco da camada de planejamento para determinar o condicionamento do controle ou do planejamento cooperativo.

A presente proposta de controle propõe uma metodologia na qual integra o controle do sistema de direção com o controle de trajetórias possibilitando prever as trajetórias desejadas do condutor, sem tratá-lo como uma entrada exógena, o que no estado da arte revisado é feito em camadas ou malhas de controle diferentes. Esta estratégia é análoga à apresentada por (ENACHE *et al.*, 2009) que detecta situações de emergência para a desatenção do condutor de forma binária o que leva a validar suas estratégias através de controle comutado.

O controle preditivo utilizado na camada de controle permite projetar a cooperação entre o sistema autônomo e o condutor através de um sistema MIMO integrando os benefícios do controle *feedback* e do controle *feedforward* porque são levadas em conta entradas exógenas e endógenas. O controle preditivo utiliza a informação presente e futuro da trajetória planejada, e os sinais de controle anteriores o que ajuda a diminuir o risco de consistência (consumo de energia) e o risco intrínseco (suavidade do caminho) porque fazem parte da otimização dos sinais de controle.

A técnica de planejamento de trajetórias selecionada permite utilizar os conceitos de funções de custo de suavidade, consistência e margem de segurança frente a obstáculos os quais foram relacionados com os conceitos de risco intrínsecos e extrínsecos do veículo. Uma das

vantagens encontradas neste método é que ele avalia a melhor trajetória independente da velocidade e sua facilidade para ser estendido com outras funções de risco. Uma das propostas de contribuição deste trabalho incorpora nesta técnica de planejamento uma nova função de risco para analisar objetos em movimento. Adicionalmente este método permitiu analisar os riscos dos caminhos estimados que o condutor deseja fazer quando o sistema está em controle cooperativo.

A transformação de um veículo comercial em um veículo autônomo, VILMA01 (Primeiro Veículo Inteligente do Laboratório de Mobilidade Autônoma), foi bem sucedida. Foi desenvolvida a través de uma metodologia baseada nos conceitos da mecatrônica porque analisou o problema de forma holística para projetar as partes do sistema e a interação entre as mesmas. Na conversão foram testados os elementos individuais da automação e a integração dos mesmos. A arquitetura de hardware e software implementada permitiu a integração da confiabilidade de um sistema embarcado de tipo automotivo (MicroAutobox Dspace) com a flexibilidade e capacidade de sistemas distribuídos de computação tipo PC baseado em Linux e ROS. Esta arquitetura permitiu um rápido desenvolvimento, um aproveitamento de códigos já existentes e realizar fáceis procedimentos de depuração.

A partir das análises dos dados da instrumentação atual do VILMA01 foram encontrados que para se ter uma localização confiável é necessário ter sensores, como uma IMU, que permitam estimar o ângulo de guinada a frequências próximas da frequência de controle de movimento. Adicionalmente foi encontrado que é requerido ter um sensor adicional para medir o ângulo da direção devido que o encoder do motor de direção apresenta uma zona morta para esta medida quando esta ativado o controle cooperativo.

Estes elementos faltantes na instrumentação para a localização robusta do veículo impediu implementar o controle cooperativo baseado na técnica de controle preditivo no VILMA01, portanto foi implementado uma técnica de controle baseado em imagens para validar os conceitos de condicionamento de controle e planejamento cooperativo.

Foi implementado com sucesso um controlador baseado em imagens, para seguir as linhas da estrada, com uma câmera de baixo custo. Testes com velocidades abaixo de 60km/h foram bem sucedidos em diferentes condições do ambiente. A técnica de controle proposta implementada que incluiu estratégias de controle e planejamento cooperativo permitiu ao condutor assistir ao controle autônomo para seguir a linha da estrada, bem como variar a referência lateral do planejamento através do volante da direção, permitindo validar os conceitos de condicionamento propostos nesta tese.

Como trabalhos futuros se propõe a implementação no VILMA01 da proposta de condicionamento do controle cooperativo com análises de risco através da técnica de controle preditivo com a finalidade de validar experimentalmente a predição dos caminhos desejados pelo condutor.

Para a implementação, o primeiro passo que deve ser feito é adicionar a instrumentação necessária e uma identificação dos parâmetros do modelo da direção e do veículo para se ter uma localização confiável. O seguinte passo seria testar com dados reais os algoritmos de localização e controle de trajetórias que já foram testados no sistema embarcado com *software in the loop*.

Finalmente, outras propostas futuras seriam a de levar a proposta de condicionamento de controle cooperativo por análises de risco no controle longitudinal através da dinâmica longitudinal do veículo, assim como a de incluir análises de cenários na etapa de planejamento que permita correlacionar os parâmetros das funções de custo de risco extrínseco e intrínseco com as estatísticas de acidentes, de forma que de acordo aos riscos determinados pelas estatísticas, o veículo autônomo possa ter um planejamento mais seguro.

Referências

Nhtsa light vehicle antilock brake system research program task 5.2/5.3: Test track examination of drivers collision avoidance behavior using conventional and antilock brakes. Relatório Técnico DOT HS 809 561, National Highway Traffic Safety Administration (NHTSA), March 2003.

Fatos e estatísticas de aciedntes de trânsito em são paulo. Relatório técnico, CET, Companhia de Engenharia de Trafego do São Paulo, 2011.

1000 relatorios de investigacao de acidente municipio de são paulo. Relatório técnico, CET, Companhia de Engenharia de Tráfego Sao Paulo, 2012.

AHN, C.; PENG, H. e TSENG, H.E. Estimation of road friction for enhanced active safety systems: Dynamic approach. In **American Control Conference, 2009. ACC'09.**, pp. 1110–1115. IEEE, 2009.

ANIMATICS. **USER'S GUIDE , Class 5 SmartMotor™ Technology with combitronic.** MOOG Animatics, rev. b ed., 2014.

ARRUDA, T.A. Arquitetura de hardware e software para supervisão e controle de um carro autônomo. In **Congresso Brasileiro de Automática, Campina Grande.** 2012.

ARTHUR DE MIRANDA, N. **Sistema de percepção visual embarcado aplicado à navegação segura de veículos.** 2011. Tese (Doutorado). Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica.

AVAK, B. **Modeling and Control of a Superimposed Steering System.** 2004. Dissertação (Mestrado). School of Electrical and Computer Engineering Georgia Institute of Technology, USA.

BAHADORIMONFARED, A.; SOORI, H.; MEHRABI, Y.; DELPISHEH, A.; ESMAILI, A.; SALEHI, M. e BAKHTIYARI, M. Trends of fatal road traffic injuries in iran (2004–2011). **PloS one**, v. 8, n. 5, e65198, 2013.

BAJCINCA, N.; NUTHONG, C. e SVARICEK, F. Road feedback estimation for steer-by-wire control. In **Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE**, pp. 1288–1293. oct. 2006.

BALEEIRO, M.F. Sistema de acionamento elétrico para frenagem de um veículo autônomo. Relatório técnico, Universidade Federal de Minas Gerais, 2009.

BARFOOT, T. e CLARK, C. Motion planning for formations of mobile robots. **Robotics and Autonomous Systems**, v. 46, n. 2, 65–78, 2004.

URL: <http://www.sciencedirect.com/science/article/pii/S0921889003001854>

BERNARDES VITOR, G. **Urban environment perception and navigation using robotic vision: conception and implementation applied to automous vehicle**. 2014. Tese (Doutorado). Compiègne.

BERTOZZI, M.; BROGGI, A.; COATI, A. e FEDRIGA, R.I. A 13,000 km Intercontinental Trip with Driverless Vehicles: The VIAC Experiment. **IEEE Intelligent Transportation System Magazine**, v. 5, n. 1, 28–41, 2013. ISSN: 1939-1390.

BRUYNINCKX, H. Open robot control software: the orocos project. In **Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on**, v. 3, pp. 2523–2528. IEEE, 2001.

CAMPBELL, M.; GARCIA, E.; HUTTENLOCHER, D.; MILLER, I.; MORAN, P.; NATHAN, A.; SCHIMPF, B.; ZYCH, N.; CATLIN, J.; CHELARESCU, F. *et al.* Team cornell: technical review of the darpa urban challenge vehicle. **DARPA Urban Chall. Tech. Pap**, 2007.

CETIN, A.; ADLI, M.; BARKANA, D. e KUCUK, H. Compliant control of steer-by-wire systems. In **Advanced Intelligent Mechatronics, 2009. AIM 2009. IEEE/ASME International Conference on**, pp. 636–643. 2009.

CETIN, A.E.; ADLI, M.A.; BARKANA, D.E. e KUCUK, H. Adaptive on-line parameter identification of a steer-by-wire system. **Mechatronics**, v. 22, n. 2, 152 – 166, 2012.

URL: <http://www.sciencedirect.com/science/article/pii/S0957415812000037>

CHEN, H.; YANG, Y. e ZHANG, R. Study on electric power steering system based on {ADAMS}. **Procedia Engineering**, v. 15, n. 0, 474 – 478, 2011. <ce:title>CEIS 2011</ce:title>.

URL: <http://www.sciencedirect.com/science/article/pii/S1877705811015918>

CHEN, Y.L.; SUNDARESWARAN, V.; ANDERSON, C.; BROGGI, A.; GRISLERI, P.; PORTA, P.P.; ZANI, P. e BECK, J. Terramax: Team oshkosh urban robot. **Journal of Field Robotics**, v. 25, n. 10, 841–860, 2008.

CHERUBINI, A.; CHAUMETTE, F. e ORIOLO, G. Visual servoing for path reaching with nonholonomic robots. **Robotica**, v. 29, n. 7, 1037–1048, dezembro 2011.

URL: <http://dx.doi.org/10.1017/S0263574711000221>

CHOSSET, H.M. **Principles of robot motion: theory, algorithms, and implementation**. Bradford Books, 2005.

CHU, K.; LEE, M. e SUNWOO, M. Local path planning for off-road autonomous driving with avoidance of static obstacles. **Intelligent Transportation Systems, IEEE Transactions on**, v. 13, n. 4, 1599–1616, Dec 2012.

CORDEIRO, R.A.; AZINHEIRA, J.R.; DE PAIVA, E.C. e BUENO, S.S. Efeitos da dinâmica tridimensional no controle de trajetória de um veículo robótico terrestre com quatro rodas. In **XIX Congresso Brasileiro de Automática, CBA**. Campina Grande, september 2012.

CÁRDENAS RUEDA, M.A. **Localização híbrida para um veículo autônomo em escala usando fusão de sensores**. 2013. Dissertação (Mestrado). Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica.

CULLINANE, B.; NEMEC, P.; CLEMENT, M.C.; MARIET, R.C.E. e JONSSON, L.I.M. Engaging and disengaging for autonomous driving. julho 17 2014. US Patent App. 14/333,839.

DAIHA, H.; GARCIA, O. e VAQUEIRO, J. Modelos e observadores de estados para dinâmica vertical de veículos terrestres. In **XII Simpósio brasileiro de automação Inteligente**. oct. 2015.

DANIEL, J.; BIROUCHE, A.; LAUFFENBURGER, J.P. e BASSET, M. Energy constrained trajectory generation for adas. In **Intelligent Vehicles Symposium (IV), 2010 IEEE**, pp. 244–249. June 2010.

DE LEMOS, R.A.; GARCIA, O. e FERREIRA, J.V. Local and global path generation for autonomous vehicles using splines. In **Engineering Applications - International Congress on Engineering (WEA), 2015 Workshop on**, pp. 1–6. Oct 2015.

DE PESQUISA ECONÔMICA APLICADA, I. Impactos sociais e econômicos dos acidentes de trânsito em aglomerações urbanas brasileiras: relatório executivo. 2003.

DIAS, J.E.A. **Modelagem longitudinal e controle de velocidade de um carro autônomo**. 2013. Dissertação (Mestrado). MS thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil.

DOS TRANSPORTES, M. Normas para o projeto das estradas de rodagem. 1973.

URL: <http://www.dnit.gov.br/download/rodovias/operacoes-rodoviaras/faixa-de-dominio/normas-projeto-estr-rod-reeditado-1973.pdf>

DOUMIATI, M. **Estimation embarquée des efforts verticaux et latéraux d'un véhicule pour le diagnostic du comportement sur route**. 2009. Tese (Doutorado). Compiègne.

DOUMIATI, M.; BAFFET, G.; LECHNER, D.; VICTORINO, A. e CHARARA, A. Embedded estimation of the tire/road forces and validation in a laboratory vehicle. In **Proceedings of 9th International Symposium on Advanced Vehicle Control, Kobe Japan**, pp. 533–538. 2008.

DOUMIATI, M.; VICTORINO, A.; CHARARA, A. e LECHNER, D. A method to estimate the lateral tire force and the sideslip angle of a vehicle: Experimental validation. In **American Control Conference (ACC), 2010**, pp. 6936–6942. IEEE, 2010.

DUGOFF, H.; FANCHER, P. e SEGEL, L. An analysis of tire traction properties and their influence on vehicle dynamic performance. **SAE Technical Paper 700377**, 1970.

ECHEVERRIA, G.; LASSABE, N.; DEGROOTE, A. e LEMAIGNAN, S. Modular open robots simulation engine: Morse. In **Robotics and Automation (ICRA), 2011 IEEE International Conference on**, pp. 46–51. IEEE, 2011.

ENACHE, N.M.; NETTO, M.; MAMMAR, S. e LUSETTI, B. Driver steering assistance for lane departure avoidance. **Control engineering practice**, v. 17, n. 6, 642–651, 2009.

FALCONE, P.; BORRELLI, F.; ASGARI, J.; TSENG, H. e HROVAT, D. Predictive active steering control for autonomous vehicle systems. **Control Systems Technology, IEEE Transactions on**, v. 15, n. 3, 566–580, May 2007.

FERNANDES, L.C.; SOUZA, J.R.; PESSIN, G.; SHINZATO, P.Y.; SALES, D.; MENDES, C.; PRADO, M.; KLASER, R.; MAGALHÃES, A.C.; HATA, A.; PIGATTO, D.; BRANCO, K.C.; JR., V.G.; OSORIO, F.S. e WOLF, D.F. Carina intelligent robotic car: Architectural design and applications. **Journal of Systems Architecture**, v. 60, n. 4, 372 – 392, 2014.
URL: <http://www.sciencedirect.com/science/article/pii/S1383762113002841>

FERNÁNDEZ-MADRIGAL, J.A. **Simultaneous Localization and Mapping for Mobile Robots: Introduction and Methods: Introduction and Methods**. IGI Global, 2012.

FERREIRA, T.; GARCIA, O. e VAQUEIRO, J. Software architecture for an autonomous car simulation using ros, morse & a qt based software for control and monitoring. In **XII Simpósio brasileiro de automação Inteligente**. oct. 2015.

FIAT. **DICA FIAT:Sensor de posição do pedal do acelerador FIA T TT**. Fiat.

FISCHLER, M.A. e BOLLES, R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. **Communications of the ACM**, v. 24, n. 6, 381–395, 1981.

FORKENBROCK, G.J.; GARROTT, W.R. e BOYD, P. An over view of nhtsa’s recent light

vehicle dynamic rollover propensity research and consumer information program. **ESV, Paper**, n. 488, 2003a.

FORKENBROCK, G.J.; GARROTT, W.R.; HEITZ, M. e O'HARRA, B.C. An experimental examination of j-turn and fishhook maneuvers that may induce on-road, untripped, light vehicle rollover. Relatório técnico, SAE Technical Paper, 2003b.

FREESE, M.; SINGH, S.; OZAKI, F. e MATSUHIRA, N. Virtual robot experimentation platform v-rep: a versatile 3d robot simulator. In **Simulation, modeling, and programming for autonomous robots**, pp. 51–62. Springer, 2010.

GARCIA, O.; FERREIRA, J. e NETO, A. Design and simulation for path tracking control of a commercial vehicle using mpc. In **Robotics: SBR-LARS Robotics Symposium and Robocontrol (SBR LARS Robocontrol), 2014 Joint Conference on**, pp. 61–66. Oct 2014.

GEIGER, A.; LAUER, M.; MOOSMANN, F.; RANFT, B.; RAPP, H.; STILLER, C. e ZIEGLER, J. Team annieway's entry to the 2011 grand cooperative driving challenge. **Intelligent Transportation Systems, IEEE Transactions on**, v. 13, n. 3, 1008–1017, Sept 2012.

GHANDOUR, R.; VICTORINO, A.; CHARARA, A. e LECHNER, D. A vehicle skid indicator based on maximum friction estimation. In **18th International Federation of Automatic Control Conference (IFAC) World Congress, Milano**. 2011.

GLASER, S.; MAMMAR, S. e SENTOUH, C. Integrated driver–vehicle–infrastructure road departure warning unit. **Vehicular Technology, IEEE Transactions on**, v. 59, n. 6, 2757–2771, 2010.

GOEBL, M. e FÄRBER, G. A real-time-capable hard-and software architecture for joint image and knowledge processing in cognitive automobiles. In **Intelligent Vehicles Symposium, 2007 IEEE**, pp. 734–740. IEEE, 2007.

GOMES, G.K. **Controle preditivo em tempo-real para seguimento de trajetória de veículos autônomos**. 2006. Dissertação (Mestrado). Universidade Federal de Santa Catarina.

URL: <https://repositorio.ufsc.br/handle/123456789/88907>

GOMES, G.K. *et al.* **Controle preditivo em tempo-real para seguimento de trajetória de veículos autônomos**. 2006. Dissertação (Mestrado). UNIVERSIDADE FEDERAL DE SANTA CATARINA.

GRAY, A.; GAO, Y.; LIN, T.; HEDRICK, J.; TSENG, H. e BORRELLI, F. Predictive control for agile semi-autonomous ground vehicles using motion primitives. In **American Control Conference (ACC), 2012**, pp. 4239–4244. June 2012.

GROMPONE VON GIOI, R.; JAKUBOWICZ, J.; MOREL, J.M. e RANDALL, G. LSD: a Line Segment Detector. **Image Processing On Line**, v. 2, 35–55, 2012.

GROVER, C.; KNIGHT, I.; OKORO, F.; SIMMONS, I.; COUPER, G.; MASSIE, P.; SMITH, B. *et al.* Automated emergency brake systems: Technical requirements, costs and benefits. **Automated emergency brake systems: technical requirements, costs and benefits**, v. 1, n. 1, 1–109, 2013.

HERNÁNDEZ, R. **Proposta de uma plataforma de testes para o desenvolvimento de veículos autônomos**. 2012. Dissertação (Mestrado). Faculdade de Engenharia Mecânica, Univ. Estadual de Campinas, Campinas, Brazil.

HORN, B.K. The curve of least energy. **ACM Transactions on Mathematical Software (TOMS)**, v. 9, n. 4, 441–460, 1983.

HOWARD, T.M. e KELLY, A. Optimal rough terrain trajectory generation for wheeled mobile robots. **The International Journal of Robotics Research**, v. 26, n. 2, 141–166, 2007.
URL: <http://ijr.sagepub.com/content/26/2/141.abstract>

ILHWAN, K.; LEE, J. e SEUNG-JIN, H. A study of steering system model for steering feel's improvement of steer-by-wire system. In **FISITA World Automotive Congress**. 2008.

JONES, F.D. e RYFFEL, H.H. **Gear design simplified**. Industrial Press Inc., 3 ed., 1984.

JONES, R.W. e WILLIAMS, A.J.S. Electric power steering system with boost curve having portions defined by polynomial equations. agosto 28 2001. US Patent 6,282,472.

KAMNIK, R.; BOETTIGER, F. e HUNT, K. Roll dynamics and lateral load transfer estimation in articulated heavy freight vehicles. **Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering**, v. 217, n. 11, 985–997, 2003.

KEEN, S. **Modeling driver steering behavior using multiple-model predictive control**. 2008. Tese (Doutorado). Department of Engineering, University of Cambridge.

KEEN, S.D. e COLE, D.J. Steering control using model predictive control and multiple internal models. In **Proceedings of the 8th International Symposium on Advanced Vehicle Control, Taipei, Taiwan**. 2006.

KLIER, W.; REIMANN, G. e REINELT, W. Concept and functionality of the active front steering system. **SAE paper**, v. 20042212, 0073, 2004.

KOENIG, N. e HOWARD, A. Design and use paradigms for gazebo, an open-source multi-robot simulator. In **Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on**, v. 3, pp. 2149–2154. IEEE, 2004.

KOON, P. e WHITTAKER, W. Evaluation of autonomous ground vehicle skills. 2006.

KREYSZIG, E. **Advanced Engineering Mathematics**. John Wiley & Sons, 2010. ISBN 9780470458365.

URL: <https://books.google.com.br/books?id=UnN8DpXI74EC>

LAURSEN, L. Volvo to test self-driving cars in traffic. **IEEE SPECTRUM**, <http://spectrum.ieee.org/tech-talk/green-tech/advanced-cars/volvo-to-test-selfdriving-cars-in-traffic#>. **Up-Vr1KXla k. facebook** (last visited Mar. 16, 2014), 2013.

LECHNER, D.; SCHAEFFER, G.; YAHIAOUI, G.; COLINOT, J. e NAUDE, C. On-board estimation of friction potential. In **FISITA World Automotive Congress, Japan**. 2006.

LEE, U.; YOON, S.; SHIM, H.; VASSEUR, P. e DEMONCEAUX, C. Local path planning in a complex environment for self-driving car. In **Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2014 IEEE 4th Annual International Conference on**,

pp. 445–450. June 2014.

LEI-YAN, Y.; PING-LI, Y. e FENG, L. Research on dynamics modeling and genetic algorithm optimization of automobile steer-by-wire system. In **Information and Computing Science, 2009. ICIC '09. Second International Conference on**, v. 3, pp. 78–81. may 2009.

LEI-YAN, Y.; YAO-GUANG, Q. e FENG, L. Research on control strategy and bench test of automobile steer-by-wire system. In **Vehicle Power and Propulsion Conference, 2008. VPPC '08. IEEE**, pp. 1–6. 2008.

LI, L.; WANG, F.Y. e ZHOU, Q. Integrated longitudinal and lateral tire/road friction modeling and monitoring for vehicle motion control. **Intelligent Transportation Systems, IEEE Transactions on**, v. 7, n. 1, 1–19, 2006.

LI, X.; SUN, Z.; ZHU, Q. e LIU, D. A unified approach to local trajectory planning and control for autonomous driving along a reference path. In **Mechatronics and Automation (ICMA), 2014 IEEE International Conference on**, pp. 1716–1721. Aug 2014.

LIU, A. e CHANG, S. Force feedback in a stationary driving simulator. In **Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on**, v. 2, pp. 1711–1716. IEEE, 1995.

MACIEJOWSKI, J.M. **Predictive control: with constraints**. Pearson education, 2002.

MAMMAR, S. e KOENIG, D. Vehicle handling improvement by active steering. **Vehicle system dynamics**, v. 38, n. 3, 211–242, 2002.

MEHDIZADEH, E.; KABGANIAN, M. e KAZEMI, R. A new force feedback for steer-by-wire vehicles via virtual vehicle concept. In **Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on**, pp. 2281–2286. 2011.

MIRANDA NETO, A. **Navegação de Robôs Autônomos baseada em Monovisão**. 2007. Dissertação (Mestrado). Dissertação de Mestrado, Universidade Estadual de Campinas (UNICAMP), Campinas, Brazil.

MULLER, S.; UCHANSKI, M. e HEDRICK, K. Estimation of the maximum tire-road friction coefficient. **Journal of dynamic systems, measurement, and control**, v. 125, n. 4, 607–617, 2003.

ORGANIZATION, W.H. **Global status report on road safety: time for action**. World Health Organization, 2009.

PACEJKA, H.B. e BESSELINK, I.J.M. Magic formula tyre model with transient properties. **Vehicle System Dynamics**, v. 27, n. sup001, 234–249, 1997.

URL: <http://www.tandfonline.com/doi/abs/10.1080/00423119708969658>

PARK, J.U.; BAE, B.H.; LEE, J.W. e KIM, J.H. Design of failsafe architecture for unmanned ground vehicle. In **Control Automation and Systems (ICCAS), 2010 International Conference on**, pp. 1101–1104. IEEE, 2010.

PEDLEY, M. Tilt sensing using a three-axis accelerometer. **Freescale Semiconductor Application Note**, 2013.

PEPY, R.; LAMBERT, A. e MOUNIER, H. Path planning using a dynamic vehicle model. In **Information and Communication Technologies, 2006. ICTTA'06. 2nd**, v. 1, pp. 781–786. IEEE, 2006.

PLAÇA, R.A. Kit didático de simulação do sistema de freios abs. 2011. <Http://www.fatecsantoandre.com.br/kddsds.pdf>.

PROCA, A.B. e KEYHANI, A. Identification of power steering system dynamic models. **Mechatronics**, v. 8, n. 3, 255 – 270, 1998.

URL: <http://www.sciencedirect.com/science/article/pii/S0957415898000038>

PROTOCOL, U.D. Rfc 768 j. postel isi 28 august 1980. **Isi**, 1980.

QUIGLEY, M.; CONLEY, K.; GERKEY, B.; FAUST, J.; FOOTE, T.; LEIBS, J.; WHEELER, R. e NG, A.Y. Ros: an open-source robot operating system. In **ICRA workshop on open source software**, v. 3, p. 5. 2009.

RAFFO, G.; GOMES, G.; NORMEY-RICO, J.; KELBER, C. e BECKER, L. A predictive controller for autonomous vehicle path tracking. **Intelligent Transportation Systems, IEEE Transactions on**, v. 10, n. 1, 92–102, March 2009a.

RAFFO, G.V.; NORMEY-RICO, J.E.; RUBIO, F.R. e KELBER, C.R. Control predictivo en cascada de un vehiculo autónomo. **Revista Iberoamericana de Automática e Informática Industrial (RIAI)**, v. 6, n. 1, 63–74, 2009b.

REIF, K. **Brakes, Brake Control and Driver Assistance Systems**, v. 1. Springer, 2014.

REYES, C.; GARCIA, O.; MEIRELLES, P.S. e FERREIRA, J.V. Estimation of longitudinal and lateral tire forces in a commercial vehicle. In **Engineering Applications - International Congress on Engineering (WEA), 2015 Workshop on**, pp. 1–6. Oct 2015.

RODRIGUEZ RUIZ, M.F. **Desenvolvimento de um sistema de localização híbrido para navegação autônoma de veículos terrestres em ambiente simulado**. 2014. Dissertação (Mestrado). Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica, Campinas, Brazil.

ROESTH, M. Hydraulic power steering system design in road vehicles. **Linköping Studies in Science and Technology**, 2007.

RUSU, R.B. e COUSINS, S. 3D is here: Point Cloud Library (PCL). In **IEEE International Conference on Robotics and Automation (ICRA)**. Shanghai, China, May 9-13 2011.

SHIM, T.; ADIREDDY, G. e YUAN, H. Autonomous vehicle collision avoidance system using path planning and model-predictive-control-based active front steering and wheel torque control. **Proceedings of the Institution of Mechanical Engineers, Part D: Journal of automobile engineering**, p. 0954407011430275, 2012.

SIEGWART, R.; NOURBAKHSI, I.R. e SCARAMUZZA, D. **Introduction to autonomous mobile robots**. MIT press, 2011a.

SIEGWART, R.; NOURBAKHSI, I.R. e SCARAMUZZA, D. **Introduction to autonomous**

mobile robots. The MIT Press, second ed., February 2011b.

SNIDER, J.M. Automatic steering methods for autonomous automobile path tracking. Relatório técnico, Robotics Institute Carnegie Mellon University, february 2009.

THRUN, S.; MONTEMERLO, M.; DAHLKAMP, H.; STAVENS, D.; ARON, A.; DIEBEL, J.; FONG, P.; GALE, J.; HALPENNY, M.; HOFFMANN, G. *et al.* Stanley: The robot that won the darpa grand challenge. **Journal of field Robotics**, v. 23, n. 9, 661–692, 2006.

WANG, H.; KEARNEY, J. e ATKINSON, K. Arc-length parameterized spline curves for real-time simulation. In **5th international conference on Curves and Surfaces**. 2002.

WERLING, M.; ZIEGLER, J.; KAMMEL, S. e THRUN, S. Optimal trajectory generation for dynamic street scenarios in a frenét frame. In **Robotics and Automation (ICRA), 2010 IEEE International Conference on**, pp. 987–993. May 2010.

WONG, T. Hydraulic power steering system design and optimization simulation. 2001.

YIH, P. **STEER-BY-WIRE: IMPLICATIONS FOR VEHICLE HANDLING AND SAFETY**. 2005. Tese (Doutorado). department of mechanical engineering stanford university, USA.

URL: <http://www-cdr.stanford.edu/dynamic/bywire/dissertation.pdf>

YIH, P. e GERDES, J. Modification of vehicle handling characteristics via steer-by-wire. **Control Systems Technology, IEEE Transactions on**, v. 13, n. 6, 965 – 976, nov. 2005.

YIH, P.; RYU, J. e GERDES, J. Vehicle state estimation using steering torque. In **American Control Conference, 2004. Proceedings of the 2004**, v. 3, pp. 2116 –2121 vol.3. 30 2004-july 2 2004.

ZF, G. Electric power steering system for passenger cars and light commercial vehicles. 2012.

APÊNDICE A Automação do veículo autônomo

Este apêndice apresenta o projeto mecatrônico da automação do veículo o qual foi dividido em seis partes funcionais: a automação do sistema do volante da direção, a automação do sistema de frenagem, a automação da aceleração do veículo, a automação da troca de marcha e a arquitetura do sistema embarcado da automação.

A.1 Automação do volante da direção

Esta seção apresenta o projeto mecatrônico do sistema de direção: seleção do atuador, desenho mecânico e firmware do servomotor. A tabela A.1 apresenta uma síntese dos requisitos do motor (atuador do volante) os quais foram obtidos através de uma modelagem da cabine do condutor e de requisitos de velocidade e torque baseados na bibliografia do sistema. Com estes requisitos foi selecionado o atuador comercial e realizado o projeto mecânico. O atuador selecionado tem um sistema embarcado que tem redes de comunicação RS232 e RS485 e pode ser programado como será explicado a seguir.

| Característica | Quantidade (Kg) |
|----------------|-------------------|
| Torque Nominal | Entre 3 y 5Nm |
| Torque pico | >8Nm |
| Velocidade | >100rpm |
| Tensão | 12V |
| Longitude | 200mm |
| Diâmetro | 80mm |
| Sensor | Sensor de posição |
| Peso | <4kg |
| | |

Tabela A.1: Parâmetros requeridos do motor de direção

A.1.1 Seleção do atuador de direção

O ponto de partida para a seleção da automação do sistema de direção é o atuador elétrico que vai controlar a mesma. Portanto, se faz necessário definir seus parâmetros de potência, os quais estão relacionados com as características de velocidade e torque, que são necessários na direção do veículo. Estes parâmetros são apresentados a seguir, baseados na bibliografia.

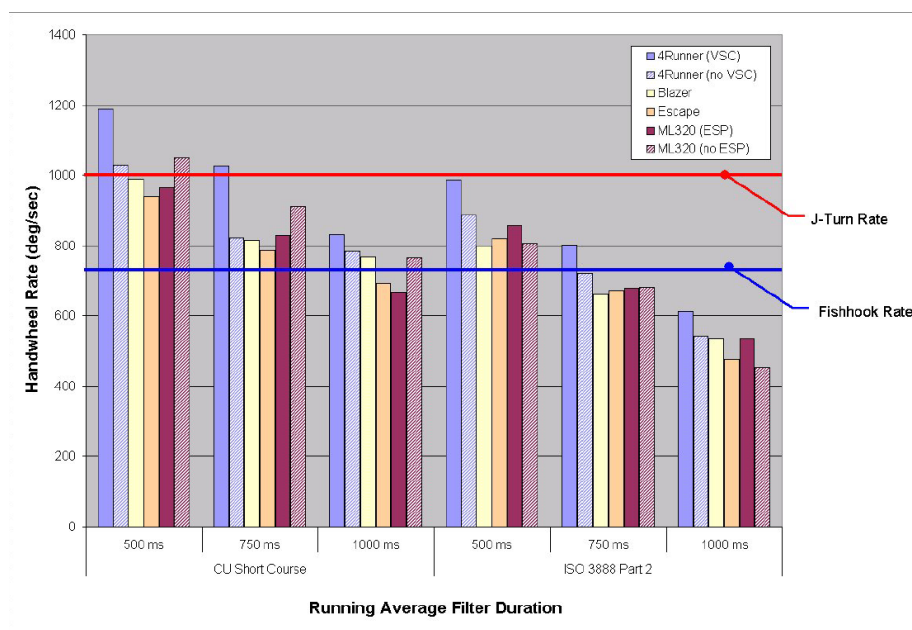


Figura A.1: Velocidades angulares requeridas em testes dinâmicos CU short course e ISO 3888 Part 2-2002 com referências de Fishhook Rate e J-turn Rate. Duração do teste J-Turn entre 287-401ms e do teste Dishook entre 647-906ms (FORKENBROCK *et al.*, 2003b)

Requerimentos de velocidade do atuador de direção

A norma técnica NHTSA (National Highway Traffic Safety Administration) requer nas provas de rollover dos veículos SUV que a direção tenha velocidades de até $700^\circ/s$ (CAMPBELL *et al.*, 2007). Na figura A.1 são mostrados testes com diferentes veículos. A velocidade do teste ($720^\circ/s$) será tomada como a velocidade máxima desejada do atuador do sistema, contudo o atuador pode apresentar velocidades menores devido ao fato que o veículo trabalhará a velocidades menores de 30km/h na etapa inicial.

Requerimentos de Torque do atuador de direção

As condições de força do atuador para a direção são desconhecidas, dado que dependem do sistema hidráulico assistido, da força de atrito de rolamento das rodas do veículo contra o chão que ao mesmo tempo dependem da velocidade do veículo, da estrada, das condições ambientais, do peso do veículo entre outros. No entanto, trabalhos de sistemas "drive by wire" e "EPAS (Electric power assisted steering)" trabalham com forças de torque típicas entre 0 e 2Nm, podendo apresentar em condições extremas até 15Nm por pequenos instantes de tempo (LIU E CHANG, 1995). Em (ILHWAN *et al.*, 2008) é apresentado um sistema drive by wire com

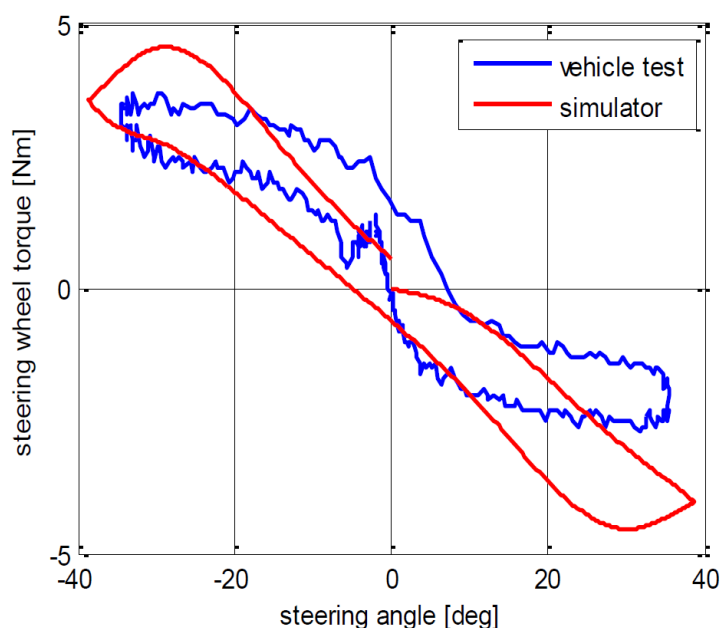


Figura A.2: Ângulo da direção vs. Torque da direção no simulador apresentado em (ILHWAN *et al.*, 2008)

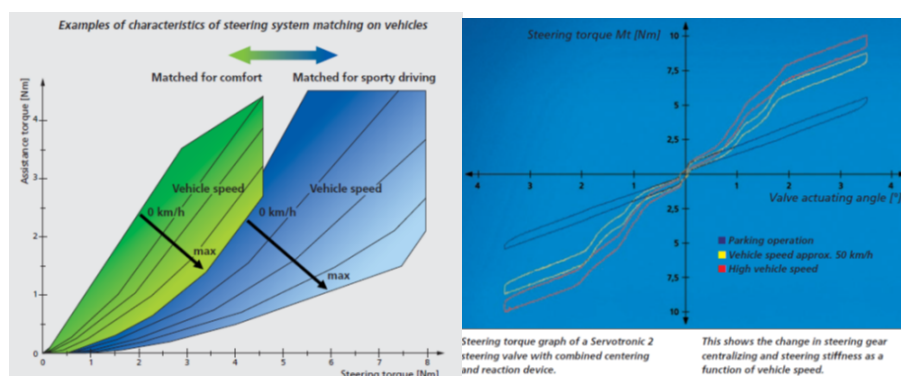


Figura A.3: Torque da direção requerido em sistema EPAS ZF servoelectric(ZF, 2012)

um modelo de simulação projetado em Matlab e validado com HIL (hardware in loop) usando Labview RT cujos resultados de feedback de torque do condutor são mostrados na figura A.2. Na figura A.3 são apresentadas curvas comerciais de sistema EPS (*electric power steering*). Finalmente, na figura A.4 é apresentado o gráfico dos torques da direção vs. as pressões de carga para o projeto de uma direção assistida hidraulicamente (ROESTH, 2007), sobre a mesma é possível apreciar que o torque em rodovias é menor que $2Nm$, em trajetos em cidade é menor que $5Nm$ e em manobras de estacionamento pode chegar à até $8Nm$.

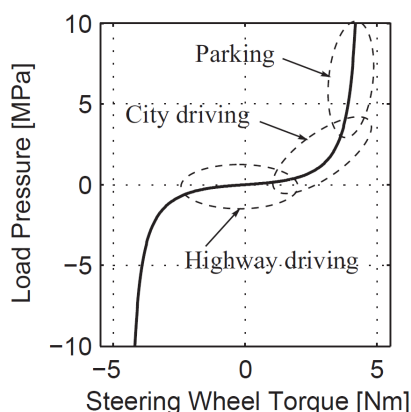


Figura A.4: Curva Boost Curve com diferentes áreas de trabalho dependendo do tipo de viagem.(ROESTH, 2007)

Opções Comerciais de Atuador

A pré-seleção foi limitada a motores DC o que gerou a tabela A.2 de motores comerciais. Estes motores requeriam procurar um fornecedor nacional e com a exceção do primeiro, requerem um sistema eletrônico de servo os quais foram pré-selecionados e mostrados na tabela A.3. O motor selecionado foi o da marca Animatics dado que na primeira tentativa foi encontrado um fornecedor no Brasil, mas finalmente foi decidido ser comprado nos Estados Unidos e importado por custos, visto que o preço do produto importado era mais vantajoso que o do fornecedor nacional. Este motor tem um controlador incluído na carcaça do motor com módulo de comunicação RS232, RS485, I2C, 6 AI ou DIO que serão utilizados na seção de arquitetura de hardware.

O esquema de funcionamento pode ser descrito pela figura A.5, de onde pode ser visto que o sistema permite trabalhar em modo posição, velocidade ou torque, gerando as referências de diferentes fontes como encoders externos, comandos, tabelas, excêntricos ou cames simulados, todos controlados através de um controle PWM e sensores de Encoder de 1000pps internos no sistema (podem ser trocados por um encoder externo) e sensores hall para as fases do motor sem escovas. No modo controle de posição ou velocidade ele implementa um controlador feedback PID a 8khz, com opção de adicionar controle feedforward de velocidade e aceleração.

| | Animatics SM23165DT | MMP S22-346C-12V GP52-022 | MMP BL58-487F-24V GP52-007 | Maxom motor |
|---------------------------------|--|------------------------------------|------------------------------------|--|
| Reference | Brushless Motor | Brushed Motor | Brushless Motor | Brushed Motor |
| Max speed motor(rpm) | 5100 | 4400.000 | 2112.000 | 6490.000 |
| nominal current(A) | 6.1 | 15.400 | 11.200 | 6.590 |
| Voltage(V) | 48 | 12.000 | 24.000 | 12.000 |
| Torque of motor(Nm) | 0.514 | 0.350 | 1.000 | 0.150 |
| Peak torque | 0.840 | 3.000 | 3.800 | 0.450 |
| l reflejado(kg.m ²) | 0.00000000000 | 0.00000000000 | 0.00000000000 | 0.00000000000 |
| Efficiency | 0.937 | 0.873 | 0.823 | 1.289 |
| Electric Power Motor (W) | 292.800 | 184.800 | 268.800 | 79.080 |
| Reference | | | | |
| speed output gear(rad/s) | 33.379 | 20.755 | 32.766 | 18.879 |
| speed output gear(rpm) | 318.750 | 198.198 | 312.889 | 180.278 |
| Torque output gear(Nm) | 6.656 | 5.828 | 5.400 | 4.374 |
| l reflejado(kg.m ²) | | | | |
| n(Gear) | 16.000 | 22.200 | 6.750 | 36.000 |
| Efficiency Gear | 0.810 | 0.750 | 0.800 | 0.810 |
| Nominal Power output (W) | 222.173 | 120.951 | 176.934 | 82.575 |
| Weight(kg) | <3Kg | <3Kg | 3Kg | 2Kg |
| Length (mm) | 136.962 | 174+encoder | 174+encoder | 174+encoder |
| diameter (mm) | 60 | 52 | 58 | 45 |
| Encoder | hasta 1000pps us1000 backlash 16arcmin | hasta 1000pps backlash 45arcmin | hasta 1000pps backlash 30arcmin | hasta 1000pps us1000 backlash 0,8° |
| Notes | Controller included | opcion de freno a 12v | opcion de freno a 12v | |

Tabela A.2: Motores comerciais pré-selecionados

| Controlador | ROBOTEQ | MAXOM | A-M-C |
|--------------------------|-----------------|-------------------|---------------------------------|
| | SDC2130 | APOS 24/5 | MC1XDZR02 |
| Corrente máxima contínua | 15A | 5A | 8,5rms |
| Corrente máxima Pico | 20A | 10A | 14rms |
| Voltagem de operação | 10 a 40 /v | 11 -24V | 10 a 80V |
| Encoder | não diferencial | diferencial | não diferencial |
| Comunicações | usb and RS232 | rs232/rs485 | rs232/rs485 |
| Custo | <US200 | US800 | US600 |
| Observações | API em linux | Opção rede CAN | Opção rede CAN autotuning |

Tabela A.3: Sistemas de controle de motores

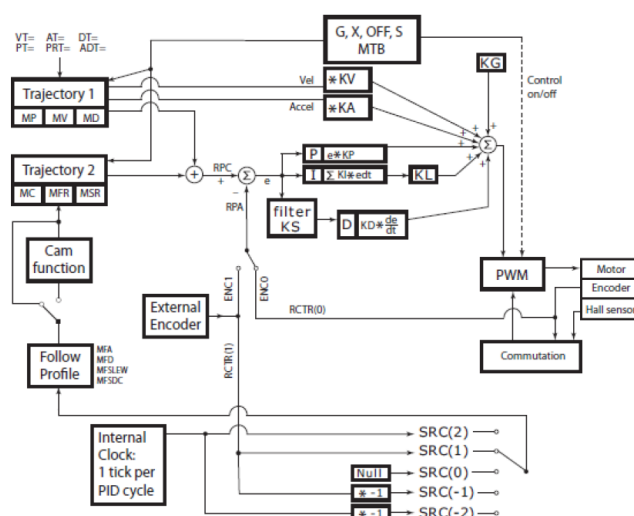


Figura A.5: Teoria de operação do motor da direção. fonte (ANIMATICS, 2014)

A.1.2 Projeto Mecânico

Uma vez definidas as características de força-velocidade requeridas, foi retirado o painel frontal do veículo e planejadas diferentes opções para o posicionamento do motor. Tendo em conta que o sistema deve ficar o menos visível possível ao condutor e que se quer utilizar o ganho de força-velocidade da assistência hidráulica e da transmissão, foram analisados os três pontos mostrados na figura A.6. Na zona verde, teria que ser modificado a peça de acabamento de plástico embaixo do mesmo e os fios do motor ficariam à vista do condutor. Na zona vermelha, dado o pedal do acelerador e do freio não havia espaço suficiente para acoplar diretamente o motor. Portanto a zona roxa foi selecionada.

Selecionada a posição de instalação do motor foi feito um modelo 3D aproximado da cabina do condutor, como o mostrado na figura A.7, que foi utilizado no projeto e desenvolvimento do sistema de transmissão de potência de acionamento da direção. Pela restrição de distância entre eixos foi selecionado desenvolver uma transmissão rígida por caixa de engrenagens. A engrenagens do sistema de transmissão foram adquiridas comercialmente de dentes retos e com material C45. Buscando um módulo do dente baixo para o par de engrenagens apresentar menor folga (JONES E RYFFEL, 1984) e levando em conta o projeto pela norma ANSI/AGMA 2001-D04:2005 para cumprir os requisitos da coluna Animatics na tabela A.2 com fator de segurança que leve em consideração o torque máximo instantâneo do motor, foi projetado o par de engrenagens com a ajuda do assistente de seleção de engrenagens do Autodesk Inventor 2012. Para o acoplamento das engrenagens com a barra de direção foi selecionado uma bucha de fixação



Figura A.6: Localização do motor de direção.

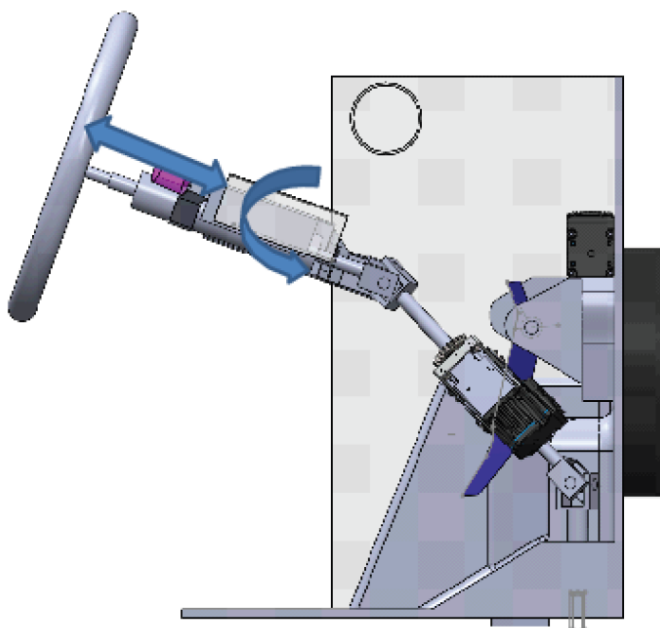


Figura A.7: Modelo 3D do sistema de direção.

cônica para ser o menos intrusivo possível por motivos de segurança. Com os três elementos selecionados foram projetadas as outras peças requeridas, que gerou o projeto da figura A.8 o qual têm os elementos listados na tabela A.4.

A.1.3 Firmware do servomotor

O firmware do servomotor é o algoritmo programado sobre a arquitetura de controle do servomotor que foi implementado para controle das interfaces de comunicação para manipulação do motor. O firmware têm como entrada dois tipos de comando (c) e (b). O comando (c) pode assumir os seguintes valores: apagado (0), velocidade (1), posição (2), torque (3), estado homing (4). O comando (b) é o valor atribuído ao estado escolhido (c) podendo assumir valores entre -10000 e 10000.

Os dados de comando e de valor por segurança vão ser recebidos pelo servomotor através de duas formas visando redundância na comunicação. A primeira forma, com maior prioridade, é pela porta serial RS232 ou pela porta RS485 (usada para depuração a partir do programa do fabricante) no qual a mensagem, composta pelo estado $c = z$ e pelo valor do comando $b = y$ e a variável $m = 1$, será enviado. Note-se que a variável m é zerada pelo motor para ter certeza que esta chegando novos dados. A segunda forma consiste de enviar dois sinais analógicos, a

primeira entrada recebe o valor analógico v_1 e armazena $z = v_1$ com resolução em volts e a segunda entrada recebe o valor analógico v_2 em uma variável inteira em centésimos de volt e armazena na variável $y = (v_2 - 200)50$. Para determinar qual entrada é interpretada pelo motor, o programa com uma frequência de 1khz lê continuamente as variáveis da serial, se por mais de um segundo não chegar o sinal de fim da trama, $m = 1$, ele vai ler as variáveis analógicas e usá-las como comando. Note que usar as variáveis de tensão, apesar de ser uma opção alternativa do sistema, apresenta menor resolução no valor do comando e possíveis erros de ruído elétrico na resolução.

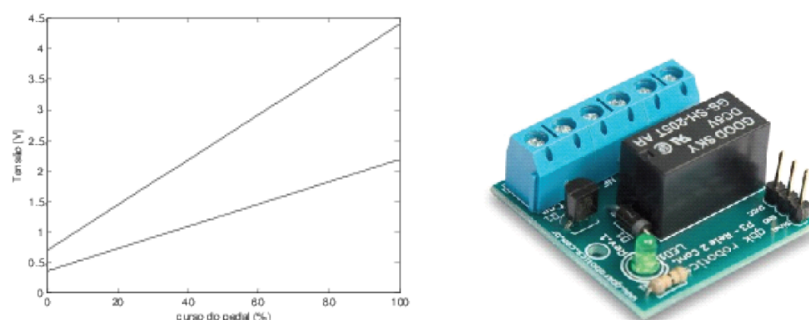
O algoritmo do firmware é listado no Algoritmo 5, tendo em conta que cada uma das funções gera os comandos requeridos pelo motor. Para maiores informações pode ser visto o programa na linguagem do fabricante. Durante todo momento o motor informa os valores de posição e velocidade atuais pelas saídas digitais do servomotor as quais serão discutidas na arquitetura de hardware da automação.

```

Input:  $y, z, m$ 
Output:  $outputs$ 
while 1 do
    if  $m = 1$  then
        print  $output$ ;
         $ii \leftarrow 0$ ;
         $c \leftarrow z$ ;
         $b \leftarrow y$ ;
    else if  $ii > 1000$  then
         $c \leftarrow v_1$ ;
         $b \leftarrow (v_2 - 200)50$ ;
    else
         $ii \leftarrow ii + 1$ ;
    switch  $c$  do
        case 1 do velocidade
        case 2 do posição
        case 3 do torque
        case 4 do homing
        otherwise do ; desligado
    end
    Sleep 1
end

```

Algoritmo 5: Algoritmo do firmware do motor de direção



(a) Tensão dos potenciômetros do pedal do(b) Chave de Relevio duplo de gbk robotics. acelerador.

Figura A.9: Automação do acelerador.

A.2 Automação da acelerador do veículo

De acordo com o manual do usuário do fabricante o acelerador do veículo é comandado por um sistema eletrônico de controle de aceleração. Portanto, o sinal do pedal é transmitido por sinais elétricos. O sinal é gerado por dois potenciômetros(FIAT, ???), um principal e um de segurança, integrados no mesmo invólucro. O sensor é utilizado pela Central de Controle do Motor (CCM) para reconhecer a posição do pedal do acelerador e gerenciar a demanda de torque por parte do condutor do veículo. A figura A.9a mostra o comportamento das tensões que vão ter que ser geradas pelo sistema de controle. Estes sinais foram geradas pelo sistema embarcado através de saídas analógicas as quais estarão ligadas a um relé de duas saídas que permitem realizar a troca com os sinais do pedal do acelerador. Adicionalmente, o sistema embarcado deve ler os dois sinais que estão gerando os potenciômetros. Para realizar este chaveamento foi comprado a solução comercial completa desenvolvido pela empresa Brasileira GBK Robotics o qual é mostrado na figura A.9b.

A.3 Automação da troca de marcha

De acordo com o manual de diagnóstico do fabricante do câmbio Dualogic, a alavanca de comando de câmbios é um circuito eletrônico que gera quatro sinais elétricos (GLS0, GLS1, GLS2, GLS3) pelas quais é comandado as trocas por válvulas eletrohidráulicas. A tabela A.5apresenta os sinais gerados para cada estado da alavanca de comandos. Para gerar nível 1 a saída deve estar entre 2,7V e 3,9V e para gerar nível 0 a saída deve estar entre 0,7V e 2,1V.

Estes sinais foram geradas pelo sistema embarcado através de saídas digitais com PWM

| | GLS0 | GLS1 | GLS2 | GLS3 |
|-------|------|------|------|------|
| Falha | 1 | 1 | 1 | 1 |
| D | 0 | 1 | 0 | 1 |
| - | 0 | 0 | 1 | 1 |
| + | 1 | 0 | 1 | 0 |
| N | 1 | 0 | 0 | 1 |
| R | 1 | 1 | 0 | 0 |
| A/M | 0 | 1 | 1 | 0 |
| NAA | 0 | 0 | 0 | 0 |

Tabela A.5: Sinais de comando da alavanca de comando de câmbios Dualogic. Fonte Manual de diagnóstico do câmbio Dualogic Fiat.

a 10khz, as quais estão ligadas a dois relés de duas saídas que permitem realizar a troca com os sinais da alavanca de comandos. Para realizar este circuito foi comprado a solução comercial completa o qual é mostrado na na figura A.9b.

A.4 Automação da frenagem do veículo

A frenagem do veículo é feita por um sistema hidráulico que ativa o freio de cada uma das rodas de acordo com a força gerada no pedal pelo condutor, que é convertida em uma pressão pelo cilindro mestre. Entre a força gerada e a que chega as rodas existem um circuito eletro-hidráulico comandado por um sistema de controle eletrônico (ECU) que está encarregado das funções de ABS (Sistema de antibloqueio das rodas), funções de EBD (*Electronic Braking Device*) e funções de ESP (Electronic Stability Control) entre outras(REIF, 2014).

A primeira função consta de sensor de velocidade em cada roda para monitorar a velocidade de cada roda. Se detectar que qualquer roda girando a uma velocidade mais lenta do que as outras (isso indica um bloqueio das rodas iminente) ele redireciona as válvulas da roda para reduzir a pressão do freio. Por outro lado, se ele detecta uma roda mais rápida do que as outras, ele ativa as válvulas para aumentar a pressão do freio. Tudo isso nos ajuda a manter o controle do carro mesmo em situação extrema.

A segunda função EBD é um subsistema do ABS e que sempre funciona em conjunto com o sistema de ABS. O trabalho principal do EBD é otimizar a força de frenagem em cada roda individualmente de modo a obter potência máxima de frenagem sem perder o controle. Ele pode alterar a pressão de frenagem em cada roda individualmente de acordo com as condições e distribuição de peso do veículo naquele momento.

A terceira função ESP está projetada para intervir e ajudar sempre que detectar uma perda de controle da direção. Ele monitora continuamente a direção do veículo (por medição do ângulo pelo qual o volante foi girado) versus a direção em que o carro está realmente seguindo. Se o sistema do carro não está indo na direção pretendida, ele aciona a função e tenta recuperar o controle fazendo uma frenagem automática em duas ou quatro rodas em pares diagonais, roda dianteira esquerda com traseira direita e roda dianteira direita com traseira esquerda.

O sistema original de frenagem do carro de testes do laboratório não tem a função de ESP, o que não permite fazer o controle autônomo de frenagem pelo sistema eletro-hidráulico dado que as bombas não podem gerar pressão independentemente da gerada pelo usuário (REIF, 2014). Portanto em uma primeira etapa do projeto a ideia foi por meio de um atuador elétrico aplicar força no pedal. Para isso de acordo a bibliografia foi encontrado que a força exercida pelo condutor no freio pode chegar até 100lbf em 0.1segundos (CAMPBELL *et al.*, 2007), valores que concordam com os utilizados em (BALEEIRO, 2009) que mostraram que a força no freio deve ser de 400N com um curso de deslocamento de 50mm.

Em (NHT, 2003) é mostrado a distribuição estatística da força do pedal em função das condições do pavimento (figura A.10), onde o valor do quinto percentual da distribuição foi assumido como a força máxima nominal requerida o qual concorda com os dados de (GROVER *et al.*, 2013), onde a força do pedal que pode passar em 0,09s de 10 a 500N em condições de emergência. Com este valor como base e com os valores das velocidades de reação obtidas do mesmo documento se fez inicialmente um projeto de um atuador e de um mecanismo de acoplamento ao pedal, o qual não foi implementado, dado que a empresa TRW fez posteriormente a doação de uma ECU de freio com capacidade de ESP. Com isso a atuação mecânica foi descartada e a atuação no freio se voltou a um projeto de controle electro-hidráulico e de redes de comunicações os quais são explicados brevemente na sequência.

A.4.1 Sistema eletro-hidráulico de frenagem

O sistema eletro-hidráulico consta de 12 válvulas de 2 vias, uma bomba de descompressão automática eletro-hidráulica (self priming pump), um acumulador hidráulico, um cilindro mestre e um sensor de pressão hidráulica. O diagrama da figura A.11 apresenta o circuito completo onde FR, FL, RL e RR, significam respectivamente roda dianteira direita, roda dianteira esquerda, roda traseira esquerda e roda traseira direita. As válvulas ISO e DUMP de cada roda realizam as funções de ABS, sendo ISO a válvula encarregada de isolar a pressão da força da roda e DUMP a válvula encargada de descarregar a pressão. As válvulas SUPPLY permitem o

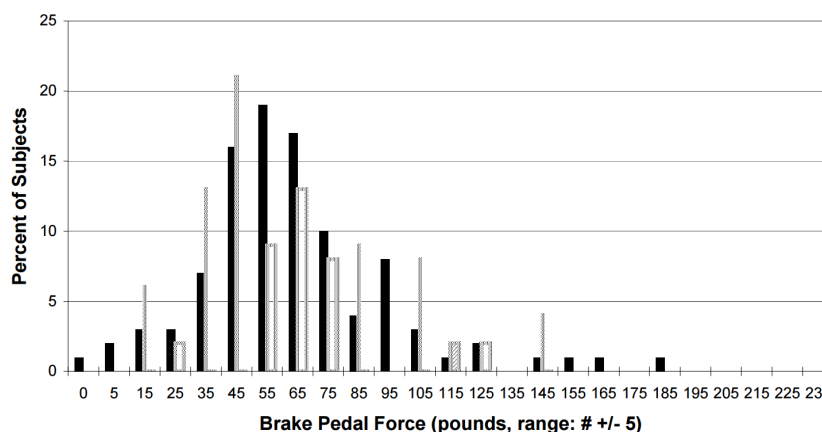


Figura A.10: Distribuição da força máxima do pedal do freio pelas condições do pavimento. Fonte (NHT, 2003)

passo do fluido nas bombas respectivas e as TC ISO são válvulas de troca a altas pressões para isolar a pressão no circuito respectivo sem necessidade de ser fornecida pressão pelo cilindro mestre. Finalmente note que o sensor de pressão só fornece a pressão do usuário.

Para explicar o funcionamento vamos simplificar o circuito contendo apenas os componentes e sensores relativos ao funcionamento do freio da roda dianteira esquerda. Dividiremos as fases do freio em 5 estados nos quais os sinais de controle das válvulas e da bomba estão listados na tabela A.6. No estado zero todos os sinais estão desligados como é mostrado na figura A.12, portanto se uma pressão for gerada no pedal, a configuração atual vai deixar passar livremente a pressão pela válvula ISO_TC_1 e também pela válvula ISO_FL chegando ao freio. No estado um, a bomba MOTOR PUMP estará ligada e a válvula TC_1 SUPPLY estará ativa o que faz com que a bomba MOTOR PUMP consiga gerar uma vazão no sistema, mas como o reservatório de fluido nesta configuração se realimenta pela válvula aberta ISO TC, o sistema não gerará pressão como é mostrado na figura A.13. No estado dois a válvula ISO_TC_1 está acionada, portanto a bomba MOTOR PUMP vai gerar pressão para o freio sem gerar pressão no pedal (figura A.14). Nota-se nesta configuração que a pressão máxima será alcançada rapidamente, portanto a bomba MOTOR PUMP pode ser desligada e a pressão ficará estável. O estado três é um exemplo do funcionamento típico do ABS (figura A.15), onde para evitar um aumento de pressão que acarretará o travamento da roda, aciona-se a válvula ISO FL para interromper a aplicação da pressão nesta roda mas continuar aplicando pressão nas outras rodas. Finalmente o estado quatro, também outro exemplo típico do funcionamento do ABS (figura A.16), para eliminar o travamento da roda pode-se diminuir a pressão aplicada acionando a válvula DUMP FL respectiva. Uma discussão detalhada do funcionamento do sistema ABS pode ser encontrada em (PLAÇA, 2011).

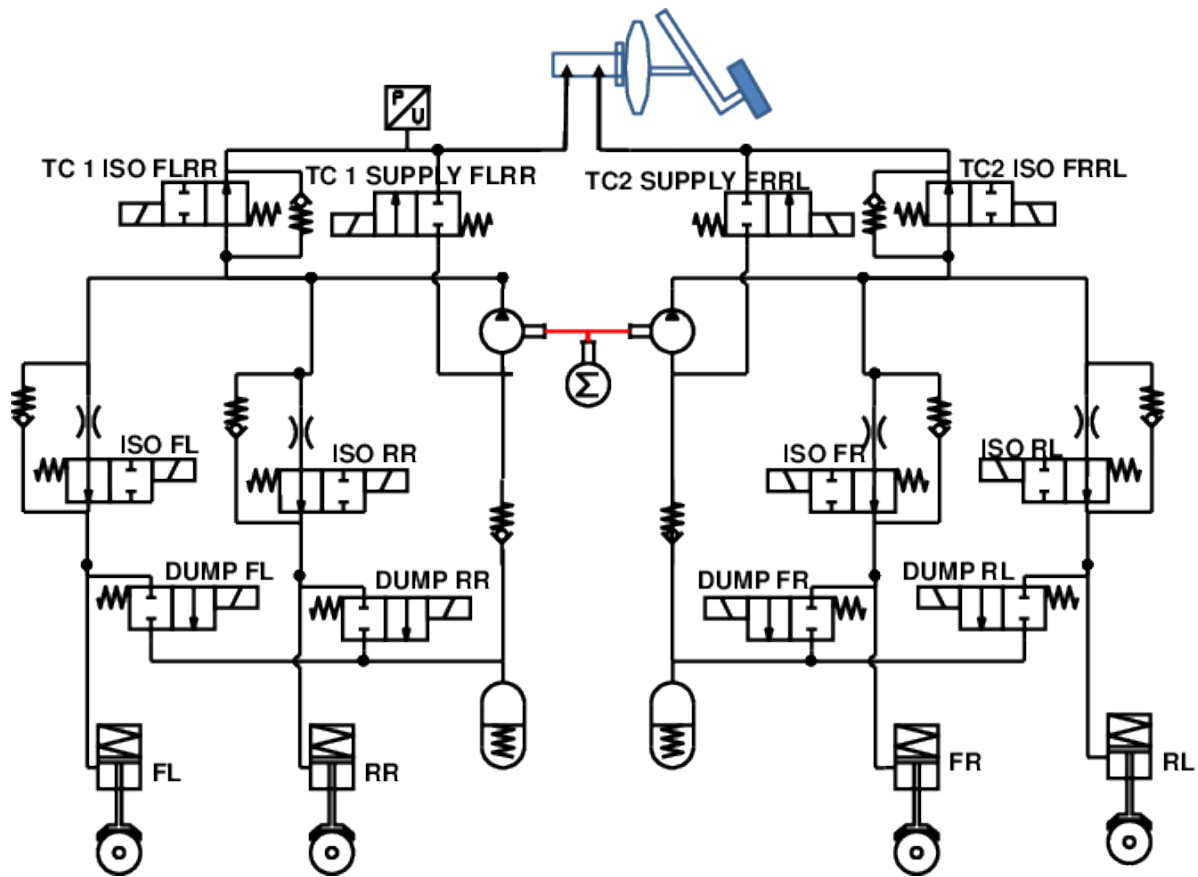


Figura A.11: Diagrama do sistema eletro-hidráulico do ECU do freio. Adaptado de (REIF, 2014)

| Sinal elétrica | estado 0 | estado 1 | estado 2 | estado 3 | estado 4 |
|----------------|----------|----------|----------|----------|----------|
| Motor Pump | 0 | 1 | 1 | 1 | 1 |
| TC_1 Supply | 0 | 1 | 1 | 1 | 1 |
| ISO_TC_1 | 0 | 0 | 1 | 1 | 1 |
| ISO FL | 0 | 0 | 0 | 1 | 1 |
| DUMP FL | 0 | 0 | 0 | 0 | 1 |

Tabela A.6: Estados principais do freio para o funcionamento do freio autônomo.

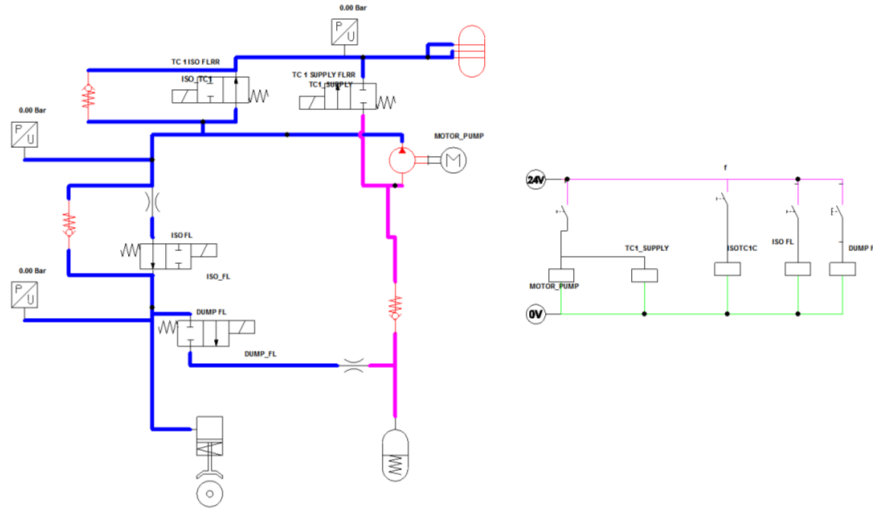


Figura A.12: Diagrama hidráulico do Estado 0 de funcionamento do freio

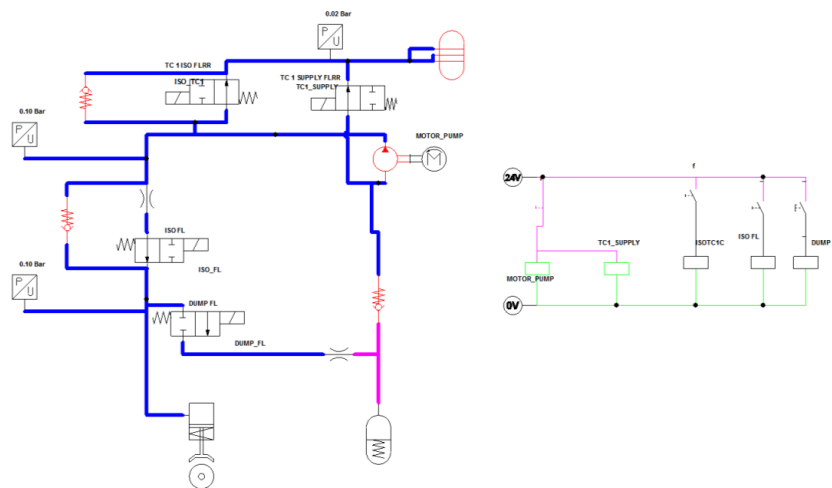


Figura A.13: Diagrama hidráulico do Estado 1 de funcionamento do freio.

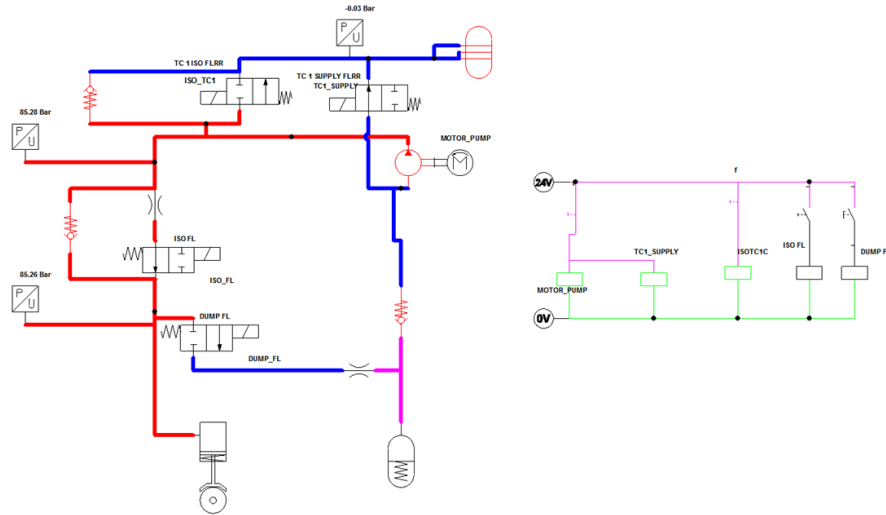


Figura A.14: Diagrama hidráulico do Estado 2 de funcionamento do freio.

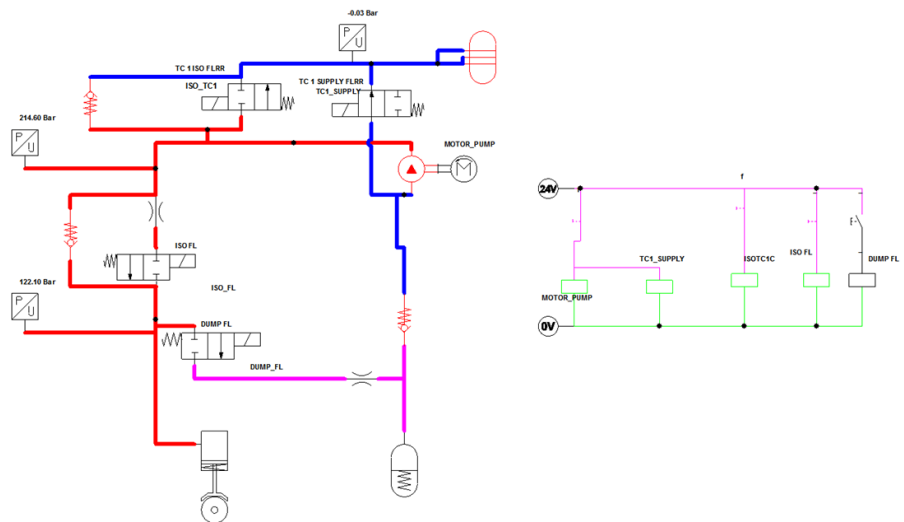


Figura A.15: Diagrama hidráulico do Estado 3 de funcionamento do freio.

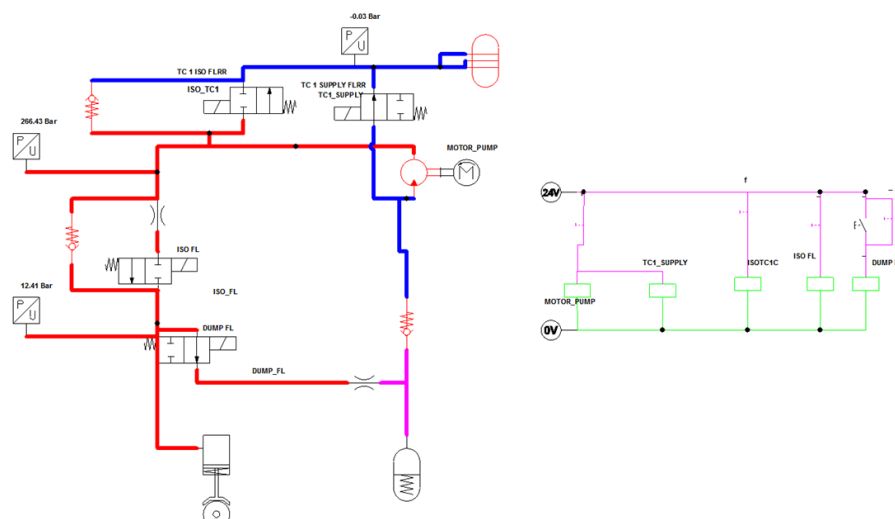


Figura A.16: Diagrama hidráulico do Estado 4 de funcionamento do freio.

| | CAMADA ISO | REDE CAN |
|---|-----------------|------------------------|
| 7 | Aplicação | ISO 14229 /ISO 15765-3 |
| 6 | Apresentação | — |
| 5 | Sessão | ISO 15765-2 |
| 4 | Transporte | ISO 15765-2 |
| 3 | Rede | ISO 15765-2 |
| 2 | Enlace de dados | ISO 11898 |
| 1 | Física | ISO 11898 |

Tabela A.7: Norma técnica correspondente a cada camada do modelo OSI para rede CAN na função de serviços de diagnóstico.

A.4.2 Comunicação com o módulo de frenagem

As ECU dos veículos tem diferentes protocolos de comunicação para coordenar os trabalhos realizados entre eles. A ECU fornecida pela TRW se comunica pela rede CAN (*Controller Area Network*), a rede mais comum no setor veicular. A rede CAN é um protocolo baseado em mensagem desenvolvido em 1983 por Robert Bosch GmbH o qual nas camadas um e dois do modelo ISO está regulado pela norma ISO 11898 e nas seguintes camadas existem diferentes protocolos de acordo com a aplicação. Para o nosso caso a comunicação para o envio de comandos será feita pelo protocolo de diagnóstico automotivo, descrito de acordo a modelo ISO de comunicação pela tabela A.7. No caso da recepção de dados dos sensores e dos estados da ECU do freio não será usada a mesma camada de aplicação pois estas mensagens estão programadas para estarem disponíveis para todos os nós da rede (Broadcast) e por intermédio de um dicionário proprietário do fabricante os dados são interpretados pelo sistema embarcado.

As camadas um e dois são implementadas no hardware, conhecido como *transceiver*, onde a primeira define que a transmissão de sinais elétricos é por um par de cabo com tensão diferencial, conectados todos os nós (elementos ligados a rede) ao barramento e com resistência de 120Ω no nó inicial e final para acoplar impedâncias. A distância dos cabos depende da velocidade de transmissão que para o caso do freio é de um milhão de bits por segundo (1mbit/s). A segunda camada define como cada nó da rede recebe ou envia dados, utilizando uma arbitragem bit a bit sem perdas (lossless bit-wise arbitration) o qual requer que todos os nós estejam sincronizados. A forma em que os bits são organizados consta de sete partes de acordo com as seguintes funções:

- **Começo (start)** - esse campo utiliza um bit (em zero) para marcar o começo da transmissão ou recepção de dados.
- **Estado (Arbitration field)** - esse campo utiliza um identificador (único), que também representa a prioridade da mensagem. Pode ser de dois tipos de acordo ao tipo de mensagem, o formato padrão (descrito no CAN 2.0 A e CAN 2.0 B) e o formato estendido (apenas descrito pelo CAN 2.0 B). A única diferença entre os dois formatos é que a "estrutura padrão" suporta um comprimento de 11 bits para o identificador, e o formato estendido suporta um comprimento de 29 bits para o identificador, composta pelo identificador de 11 bits ("identificador de base") e uma extensão de 18 bits ("extensão identificador").
- **Controle (Control)**- Define a quantidade de informações que vai ser recebida ou enviada (4 bits) e o tipo de mensagem (2 bits, um reservado), no caso da mensagem ser estendida leva os 18 bits e um bit de pedido de transmissão remota.
- **Dados (Data)** - são as informações transmitidas ou recebidas propriamente ditas que podem ter um tamanho até de 64 bits (8 bytes).
- **Verificação (CRC)** - este campo é composto por 16 bits que são utilizados para detectar falhas na transmissão dos dados pelo cálculo da CRC (cyclic redundancy check) da mensagem.
- **Conformação (ACK)**- são dois bits utilizados para indicar que os dados foram recebidos corretamente.
- **Fim (End)**- esse campo tem 7 bits (em um) que são utilizados para indicar o fim da transmissão do "pacote" de dados.

A camada três até cinco são só requeridas quando as mensagens excedem 8 bytes na camada de

aplicação, os quais em nenhum dos dois tipos de mensagens que são trabalhados com o freio são utilizados.

A camada de aplicação que envia os comandos ao servofreio é a norma ISO 14229. Esta norma é conhecida como Serviços de diagnóstico unificados, conhecidos pelo acrônimo UDS (Unified Diagnostic Services), que é utilizado em veículos automotores para o diagnóstico. É uma evolução da norma ISO 14230-3 (KWP2000), ISO 15765-3 (Diagnostics no CAN) e especificação General Motors GMLAN (ISO 15765-2). O objetivo do protocolo é para facilitar o diagnóstico entre um cliente e as unidades de controle eletrônico (servidores) conectados ao barramento do sistema de um carro. O cliente envia uma série de mensagens de chamadas de serviços UDS, de chamadas SID (Service Identifier), que permitem ler a memória falha de uma ECU, manipular os sinais de controle e inclusive fazer atualização (de "flash") do firmware. Os grupos dos serviços e de cada um dos seus indicadores são mostrado na tabela A.8. Os pedidos do cliente são feitos a uma direção e a resposta chega por outra mensagem com direção diferente do servidor.

Para o caso dos mensagens recebidas da ECU, como são a pressão no cilindro mestre, velocidade de guinada, aceleração em x e z, velocidade das rodas e estados da ECU chegam por mensagens de *Broadcast* que são decodificados com a ajuda de um dicionário, que descreve o que fazer com cada direção de memória.

A.5 Sistema embarcado

Como sistema embarcado para o desenvolvimento do controle do hardware o grupo do laboratório tem disponível um MicroAutobox II da empresa Dspace, unidade geralmente usada no meio automotivo. Ele está composto por duas centrais de processamento, a primeira com um sistema operacional de tempo real e arquitetura PowerPC e a segunda um Intel Atom em arquitetura PC com sistema operativo Linux e disco rígido de estado sólido de 64GB como é mostrado na figura A.17.

O sistema embarcado DSpace possui portas de entrada e saída onde será ligado o hardware do sistema de automação do veículo, a tabela A.9 mostra as características do mesmo. O seu sistema computacional em conjunto com um sistema de sensores e saídas digitais geram a arquitetura de hardware do sistema de automação. O sistema de hardware vai estar interligado com o sistema embarcado DSpace por cinco tipos de sinais:

| Grupo da função | Pedido SID | Resposta SID | Serviço |
|--|------------|--------------|------------------------------------|
| Diagnóstico e comandos de comunicações | 0x10 | 0x50 | Diagnostic Session Control |
| | 0x11 | 0x51 | ECU Reset |
| | 0x27 | 0x67 | Security Access |
| | 0x28 | 0x68 | Communication Control |
| | 0x3E | 0x7E | Tester Present |
| | 0x83 | 0xC3 | Access Timing Parameters |
| | 0x84 | 0xC4 | Secured Data Transmission |
| | 0x85 | 0xC5 | Control DTC Settings |
| | 0x86 | 0xC6 | Response On Event |
| | 0x87 | 0xC7 | Link Control |
| Transmissão de dados | 0x22 | 0x62 | Read Data By Identifier |
| | 0x23 | 0x63 | Read Memory By Address |
| | 0x24 | 0x64 | Read Scaling Data By Identifier |
| | 0x2A | 0x6A | Read Data By Identifier Periodic |
| | 0x2C | 0x6C | Dynamically Define Data Identifier |
| | 0x2E | 0x6E | Write Data By Identifier |
| | 0x3D | 0x7D | Write Memory By Address |
| Transmissão de dados armazenados | 0x19 | 0x59 | Read DTC Information |
| | 0x14 | 0x54 | Clear Diagnostic Information |
| entradas / saídas de controle | 0x2F | 0x6F | Input Output Control By Identifier |
| Rotina de ativação remota | 0x31 | 0x71 | Routine Control |
| Upload\Download | 0x34 | 0x74 | Request Download |
| | 0x35 | 0x75 | Request Upload |
| | 0x36 | 0x76 | Transfer Data |
| | 0x37 | 0x77 | Request Transfer Exit |
| | 0x38 | 0x78 | Request File Transfer |

Tabela A.8: Identificador de serviço do protocolo ISO 14229. Adaptado da norma técnica.

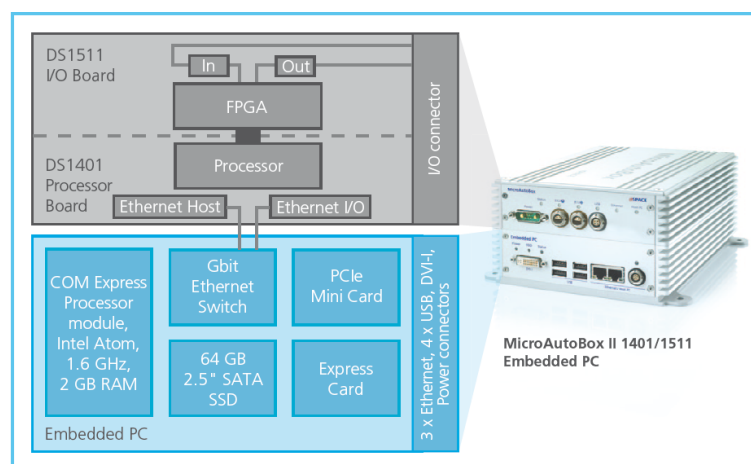


Figura A.17: Arquitetura de processamento do MicroAutobox II. Fonte do folheto do fabricante.

- Ethernet. Será encarregada de comunicar o sistema embarcado com os computadores tipo PC requeridos para os diferentes softwares que vai ter o veículo.
- RS232. Será encarregada de comunicar o sistema embarcado com os controladores dos motores da automação da direção.
- USB. A porta servirá para conectar diferentes dispositivos aos programas do computador.
- IO. São os sinais elétricos entre o sistema embarcado e os outros elementos do sistema de hardware.

A tabela A.10 mostra os sinais ligadas ao sistema embarcado. Note-se que o sistema utilizado é aberto para possibilitar adicionar novos sensores e novos atuadores.

| Parameter | | Specification | |
|---------------------|--|---|--|
| MicroAutoBox II | | 1401/1511 | 1401/1511/1512 |
| Processor | | ■ IBM PPC 750GL, 900 MHz (incl. 1 MB level 2 cache) | |
| Memory | | <ul style="list-style-type: none"> ■ 16 MB main memory ■ 6 MB memory exclusively for communication between MicroAutoBox and PC/notebook ■ 16 MB nonvolatile flash memory containing code section and flight recorder data ■ Clock/calendar function for time-stamping flight recorder data | |
| Boot time | | ■ Depending on flash application size. Measurement examples: 1 MB application: 160 ms; 3 MB application: 340 ms | |
| Inter- faces | Host interface | <ul style="list-style-type: none"> ■ 100/1000 Mbit/s Ethernet connection (TCP/IP). Fully compatible with standard network infrastructure. LEMO connector. ■ Optional XCP on Ethernet interface to support third-party calibration and measurement tools | |
| | Real-time I/O interface | ■ 100/1000 Mbit/s Ethernet connection (UDP/IP). RTI Ethernet (UDP) Blockset (optional) for read/write access. LEMO connector. | |
| | USB Interface | ■ USB 2.0 interface for long-term data acquisition with USB mass storage devices. LEMO connector. | |
| | CAN interface | ■ Two dual CAN interfaces; 4 CAN channels in total | |
| | Serial interface (based on CAN processor) | <ul style="list-style-type: none"> ■ 2 x RS232 interface ■ 2 x serial interface usable as K/L line or LIN interface | |
| | Dual-port memory interface (ECU interface) | ■ 2 x dual-port memory interface, 16 K x 16-bit DPRAM | |
| | FlexRay interface | – | ■ 2 slots ¹⁾ for FlexRay modules (i.e. 4 FlexRay channels) |
| Programmable FPGA | | – | ■ Xilinx® Spartan®-6 LX150 ²⁾ |
| Analog input | Resolution | ■ 16 16-bit channels | ■ 16 16-bit channels (additional channels with DS1552, p. 9) |
| | Sampling | ■ 16 parallel channels with 1 MSPS conversion rate | |
| | Input voltage range | ■ 0 ... 5 V | |
| Analog output | Resolution | ■ 4 12-bit channels | ■ 4 12-bit channels (additional channels with DS1552, p. 9) |
| | Output voltage range | ■ 0 ... 4.5 V | ■ 0 ... 4.5 V |
| | Output current | ■ 5 mA max. sink/source current | |
| Digital I/O | General | <ul style="list-style-type: none"> ■ FPGA-based digital I/O ■ RTI software support for bit I/O, frequency, and PWM generation/measurements | |
| | Bit I/O | <ul style="list-style-type: none"> ■ 40 inputs ■ 40 outputs, 5 mA output current | <ul style="list-style-type: none"> ■ 40 inputs (additional channels with DS1552, p. 9) ■ 40 outputs, 5 mA output current (additional channels with DS1552, p. 9) |
| | PWM generation/measurement | <ul style="list-style-type: none"> ■ Input / output logic levels: 5 V TTL or levels up to 40 V (depending on V_{BAT}), selectable ■ All channels fully configurable as frequency or PWM inputs/outputs ■ PWM frequency 0.0003 Hz ... 150 KHz, duty cycle 0 ... 100%, up to 21-bit resolution | |
| Signal conditioning | | <ul style="list-style-type: none"> ■ Signal conditioning for automotive signal levels, no power driver included ■ Overvoltage protection ■ Overcurrent and short circuit protection | |

Tabela A.9: Características técnicas do sistema embarcado 1401/1511/1512 do MicroAutobox II. Fonte do folheto do fabricante.

| | | |
|------------------|------------------------------------|---------------|
| MicroAutobox | entrada | entrada |
| DIgP ch 4 in | Boton emergencia | |
| DIgP ch 1 in | Motor Dir | encoder 1 |
| DIgP ch 2 in | Motor Dir | encoder 2 |
| Analog ch 1 out | Motor Dir | DIO 3 |
| Analog ch 2 out | Motor Dir | DIO 2 |
| Analog ch 5 in | Motor Dir/SGI | DIO 1 /SGI AO |
| serial 1 TXD out | Motor Dir | RX |
| serial 1 RXD in | Motor Dir | TX |
| GND | Motor Dir, Freno, | GND |
| | Acelerador, troca de marcha | GND |
| DIgP ch 1 out | acelerador | Sinal DO |
| VSENS | acelerador, freno, troca de marcha | 5VCC |
| Analog ch 3 out | acelerador | NF1 |
| Analog ch 4 out | acelerador | NF2 |
| Analog ch 1 in | acelerador | NA1 |
| Analog ch 2 in | acelerador | NA2 |
| CAN 1 high | freno | CAN 1 high |
| CAN 1 low | freno | CAN 1 low |
| DIgP ch 2 out | troca de marcha | Sinal DO PWM |
| DIgP ch 3 out | troca de marcha | Sinal DO PWM |
| DIgP ch 4 out | troca de marcha | Sinal DO PWM |
| DIgP ch 5 out | troca de marcha | Sinal DO PWM |
| DIgP ch 6 out | troca de marcha | Sinal DO |

Tabela A.10: Sinais de entrada/saída utilizadas no MicroAutobox.

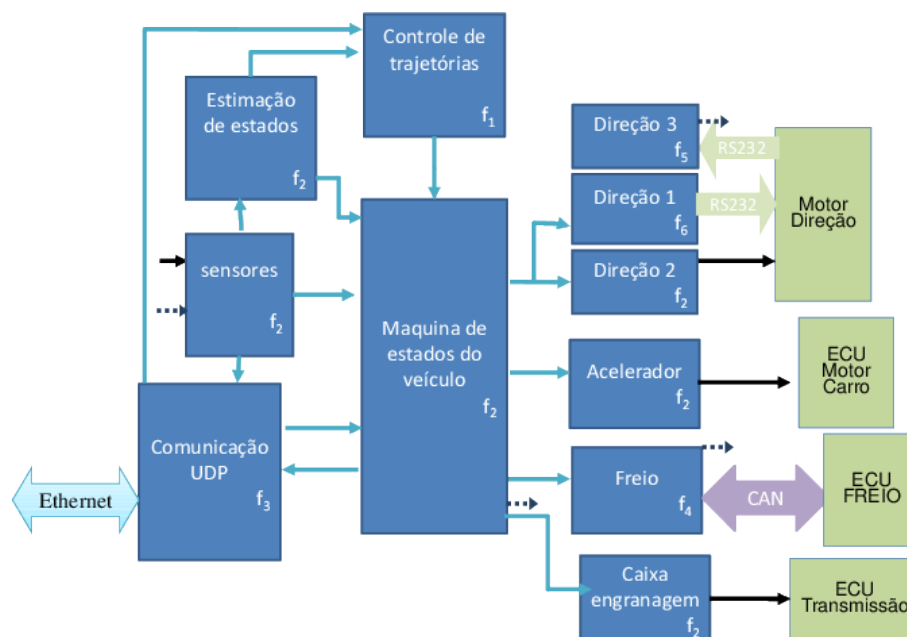


Figura B.1: Software do sistema embarcado do microautobox. Linha preta (sinal elétrico) e azul (vetor de dados)

APÊNDICE B Software no sistema embarcado MicroAutobox

O software do sistema embarcado foi projetado em torno de sua função principal que é responder aos possíveis estados em que pode estar o veículo, levando-se em conta a prioridade entre as tarefas e os tempos de execução que elas requerem. Nessa ordem de ideias as funções principais projetadas são representadas pela figura B.1, onde as linhas azuis são vetores de dados, as pretas são sinais elétricos e f_i significa o processo ou *thread* com sua respectiva ordem de prioridade na qual a função é executada.

A programação do sistema embarcado do microautobox é programado por Simulink do suite de Matlab, onde o programa representado por blocos é transformado em linguagem c e enviado ao sistema embarcado pela rede. Os processos implementados são de dois tipos, síncronos e assíncronos e podem ser comandados por software ou por hardware. A seguir são listados todos os processos requeridos:

- f_0 . Interrupção assíncrona do sistema para a comunicação Ethernet (RX/TX), deve ter a máxima prioridade e tem frequência indeterminada.
- f_1 . Processo síncrono encarregado de implementar os algoritmos de controle de veículo, implementado através do hardware por um timer a 100hz.

- f_2 . Processo principal do sistema implementado através de hardware por um timer configurado a uma frequência de 1khz, único encarregado das tarefas de configuração e interface com o hardware.
- f_3 . Interrupção assíncrona por recepção (RX) da comunicação UDP do MicroAutobox encarregada de ler os comandos do PC e transmitir (TX) de volta para ele a informação requerida pelo mesmo socket de recepção.
- f_4 . Processo síncrono encarregado de realizar a comunicação com a rede CAN a uma frequência de 100hz.
- f_5 . Interrupção assíncrona por recepção (RX) da comunicação RS232 para receber o estado do controlador do motor de direção.
- f_6 . Processo projetado síncrono, mas com capacidade assíncrona configurado através de uma interrupção de software a 100hz.

Nas seções a seguir são explicados os módulos de software implementados.

B.1 Funções da direção

O sistema da direção vai ser comandado por três variáveis (Z), (Y) e (U). No software o modo de operação (Z) pode assumir os seguintes estados: apagado (0), velocidade (1), posição (2), torque (3), estado homing (4) e cooperativo (5). Ainda no software o modo de operação (Y) e o modo cooperativo (U), podem assumir os valores entre -1 e 1 . Como o firmware do motor só suporta os primeiros quatro modos de operação, foi necessário desenvolver por software uma transformação das variáveis feita pela equação (B.1) de modo a funcionar com as duas formas de funcionamento com redundância já implementadas no firmware como explicado na seção A.1.3. A primeira diferença é que pelo software nas duas saídas analógicas agora são enviados os novos comandos dados pela equação $V_1 = 2y + 2$ e $V_2 = z + 0.1$, e a segunda diferença é que os valores em ASCII dos sinais analógicos escritos no comando a ser enviado pela porta RS232 são multiplicado por 1000, dado que o servomotor trabalha somente com valores inteiros.

$$z = \begin{cases} Z & Z < 4 \\ 3 & Z = 5 \end{cases} \quad y = \begin{cases} Y & Z < 4 \\ \min(1, |Y + U|) \text{sign}(Y + U) & Z = 5 \end{cases} \quad (\text{B.1})$$

Adicionalmente, existe uma terceira função no software que processa a informação recebida do motor pela porta serial e que roda por interrupção toda vez que são recebidos dois bytes pela porta. Esta informação recebida vai sendo armazenada até encontrar um Byte 13, depois a informação é convertida de ASCII para inteiros. Neste momento portanto foi recebido um vetor de dois valores contendo o estado do motor e o valor do torque aplicado pelo motor.

B.2 Função do acelerador

O sistema de aceleração vai ser comandado por três variáveis da mesma forma como foi explicado o funcionamento das funções da direção na seção anterior, onde a primeira variável é o modo de operação (Z) que pode assumir agora somente os estados: direto (0), velocidade (1), posição (2) e cooperativo (5), a segunda variável é o valor do modo de operação (Y) e a última variável é o valor medido pelo pedal do acelerador (U), onde os dois últimos podem assumir valores entre 0 e 1. Têm como saída o bit de ativação e os valores de tensão $V_1 = 3.69x + 0.7$ e $V_2 = 1.83x + 0.35$ e uma saída digital (d) para selecionar entre a sinal do usuário e o sinal do sistema embarcado. O hardware não suporta todos os modos de operação, portanto a transformação das variáveis é feita pela equação (B.2).

$$x = \begin{cases} Y & Z = \{0,2\} \\ \min(1, |Y + U|) \text{sign}(Y + U) & Z = 5 \\ \min(1, |x + Y/T_s|) \text{sign}(x + Y/T_s) & Z = 1 \end{cases} \quad d = \begin{cases} 1 & Z = \{1,2,5\} \\ 0 & \end{cases} \quad (\text{B.2})$$

B.3 Função da troca de marcha

O funcionamento da troca de marcha vai ser comandado por duas variáveis comando e valor, os quais geram as transições entre estados. Na figura A.5 é mostrado como podem ser realizadas as transições, dado que só quando se está no estado q_3 pode-se passar ao estados de q_4, q_5 e q_6 . Note-se que no momento atual não se tem nenhuma retroalimentação do estado que se encontra a ECU do veículo, portanto não é recomendado trabalhar em manual, o estado q_6 , ao menos que sejam conhecidas as revoluções do motor.

Esta função funciona com a frequência f_2 e um contador no momento sintonizado em 100

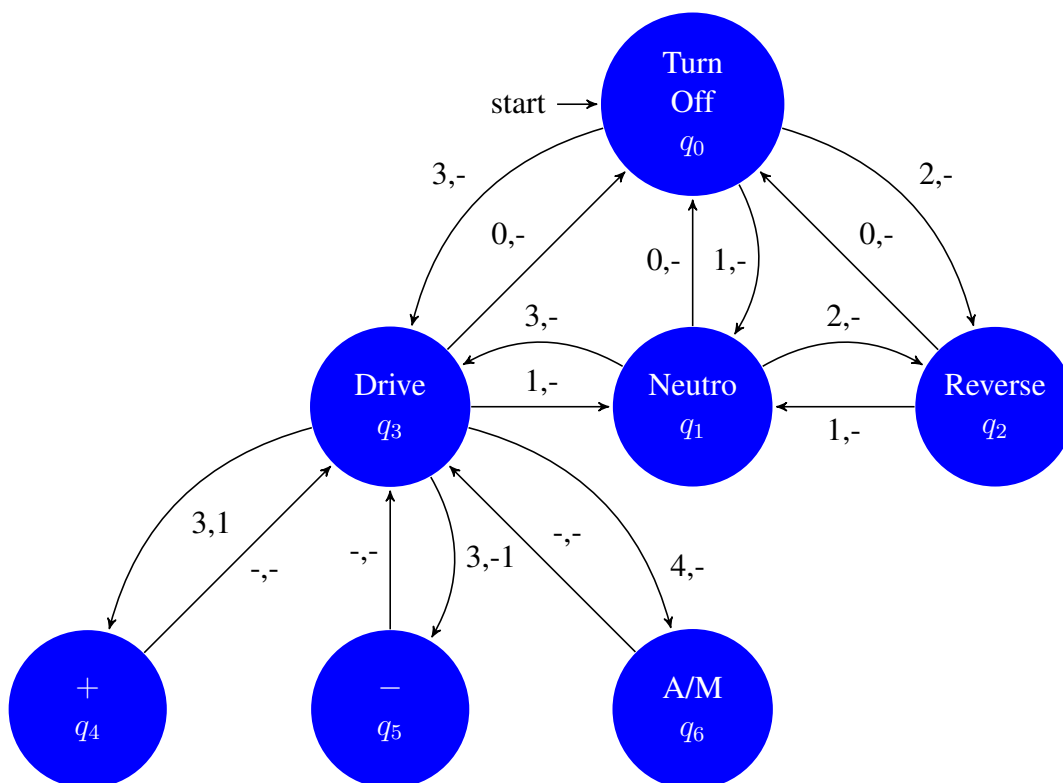


Figura B.2: Estados do envio de sinais para a troca de marcha. a troca de estados é comandada por dois sinais: comando e valor

ciclos para todos os estados, portanto a troca de estados é avaliada cada $f_2/100$. A função de utilizar um contador é que é possível que certas transições requeriam maior o menor tempo para sua transição, portanto em cada transição o número de ciclos pode para a próxima mudança de estado variar de forma independente.

B.4 Função do Freio

O sistema de freio vai ser comandado por duas variáveis, a primeira variável é o modo de operação (Z) que pode assumir os seguintes estados: direto (0), velocidade (1), posição (2) e cooperativo (5) e a segunda variável é o valor do modo de operação (Y) entre 0 e 1 e vai retornar as variáveis listadas na tabela B.1 que conseguem ser obtidas do módulo do freio. O valor do freio (x) deve ser enviado por meio das mensagens disponíveis da ECU do freio para ativação/desativação das válvulas e ou da bomba, mas antes e de forma sequencial é necessário iniciar uma sessão para mante-la aberta. Portanto o envio de dados vai ter dois estados como é mostrado na figura B.3. Como o valor enviado pela rede CAN é binário (ligado ou desligado) o sinal é modulado pela porcentagem de tempo útil (y de acordo a equação (B.3)) por um sinal

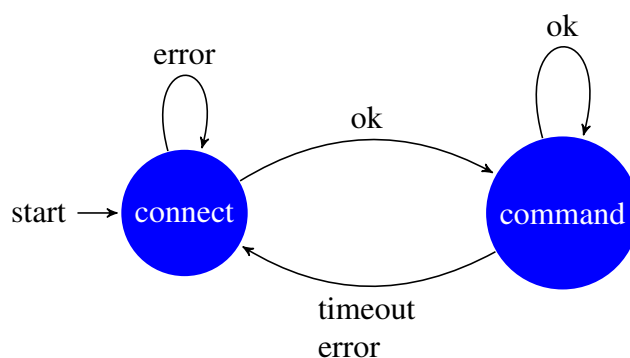


Figura B.3: Estados no envio de pacotes ao ECU do freio pela rede CAN

| Variável | Unidade | Frequência (hz) |
|------------------------------------|------------------|-----------------|
| velocidade roda dianteira/esquerda | km/h | 100 |
| velocidade roda dianteira/direita | km/h | 100 |
| velocidade roda traseira/esquerda | km/h | 100 |
| velocidade roda traseira/direita | km/h | 100 |
| aceleração em z | gravidade | 50 |
| aceleração do veículo(calculada) | m/s ² | 50 |
| aceleração a_x | m/s ² | 50 |
| pressão do cilindro mestre | Bar | 50 |
| velocidade de guinada | degraus/s | 50 |

Tabela B.1: Variáveis transmitidas pela ECU do freio pela rede CAN

tipo PWM (*pulse width modulation*) com frequência portadora f_7 que deve ser menor e múltiplo de f_4 (frequência de envio/recepção da REDE CAN). Ao diminuir a frequência f_7 o tempo de resposta fica menor e o sistema responde mais rápido, mas a resolução de y vai ser muito menor pelo limitante de f_7 .

$$y = \begin{cases} Y & Z = \{0,2\} \\ \min(1, |Y + U|) \text{sign}(Y + U) & Z = 5 \\ \min(1, |x + Y f_4|) \text{sign}(x + Y/T_s) & Z = 1 \end{cases} \quad (\text{B.3})$$

B.5 Função de sensores

A função de sensores é reunir todas as informações disponíveis do sistema e disponibilizá-las para os outros módulos. As informações sendo lidas estão sendo geradas em dois vetores chamados de sensores do carro e sensores do freio, vetores que contém as variáveis das tabelas

| Variável | Unidade | Frequência (hz) |
|------------------------------|------------------|-----------------|
| tempo | ms | f_2 |
| Estado do carro | ver (B.4) | f_2 |
| posição do motor direção | graus | f_2 |
| velocidade do motor direção | graus/s | f_2 |
| Estado do motor da direção | | $\sim f_6$ |
| valor do pedal do acelerador | 0-1 | f_2 |
| valor da aceleração | 0-1 | f_2 |
| contador do acelerômetro | — | 200 |
| a_x | m/s ² | 200 |
| a_y | m/s ² | 200 |
| a_z | m/s ² | 200 |
| Pressão atmosférica | kPA | 200 |

Tabela B.2: Variáveis dos sensores do carro

| Variável | Unidade | Frequência (hz) |
|------------------------------------|------------------|-----------------|
| tempo* | ms | f_4 |
| estado do freio | | 100 |
| valor do freio | 0-1 | 100 |
| velocidade roda dianteira/esquerda | km/h | 100 |
| velocidade roda dianteira/direita | km/h | 100 |
| velocidade roda traseira/esquerda | km/h | 100 |
| velocidade roda traseira/direita | km/h | 100 |
| aceleração em z | gravidade | 50 |
| aceleração do veiculo(calculada) | m/s ² | 50 |
| aceleração a_x | m/s ² | 50 |
| pressão do cilindro mestre | Bar | 50 |
| velocidade de guinada | degraus/s | 50 |

Tabela B.3: Variáveis do sensores do freio

B.2 e B.3. Note que nem todos os sensores tem a mesma taxa de amostragem, pois no vetor sensores do carro o acelerômetro e o sensor de pressão tem uma frequência de 200hz, a qual pode ser verificada pela variável de contador do acelerômetro que se acrescenta cada vez que o dado é gerado.

B.6 Função da comunicação UDP

A comunicação UDP é um protocolo da camada de transporte. Ele é descrito na RFC 768(PROTOCOL, 1980) e permite que a aplicação escreva um datagrama encapsulado num pacote IPv4 ou IPv6, e então enviado ao destino. Mas não há qualquer tipo de garantia que o

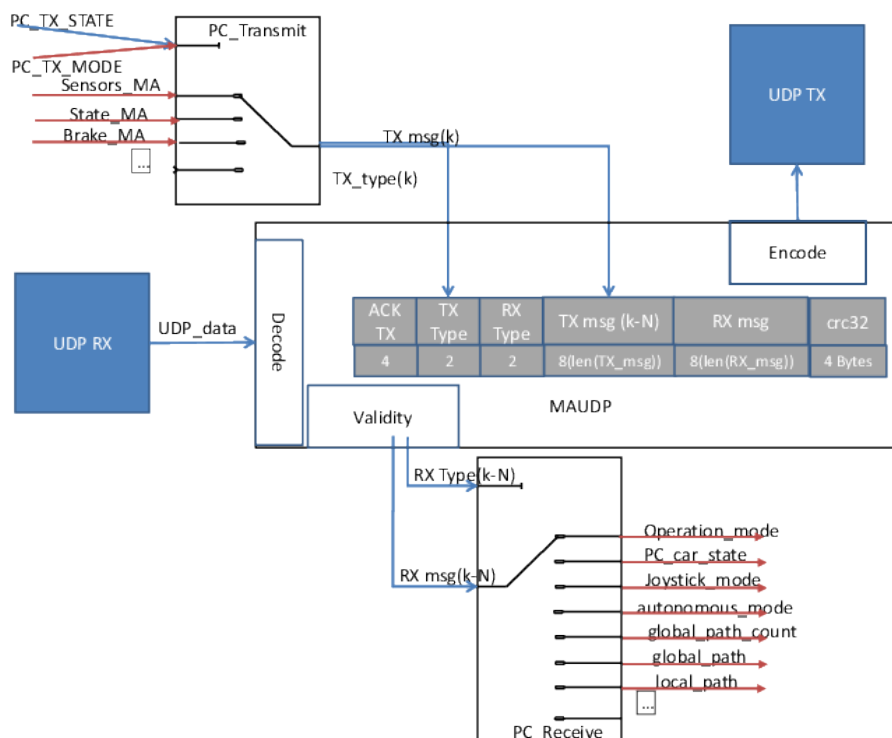


Figura B.4: Diagrama da comunicação UDP do microautobox.

pacote irá chegar ou não. Ele é o único suportado pelo sistema embarcado, portanto a confiabilidade do dado enviado vai depender das funções que os processam e serão explicados a seguir. Dado que a transmissão de pacotes de dados será a uma frequência muito alta, o tamanho da trama de dados é um valor importante pois não pode ser muito grande, e dado as limitações nas funções do sistema embarcado não pode ser variável. Para dar um exemplo 12 variáveis tipo double de envio e 10 variáveis tipo double de chegada requerem 176 bytes, os bytes da camada de IPv4 são 20 bytes, os bytes da camada UDP são 8 bytes e os bytes do nosso protocolo são 8 bytes, então resultam em 212 bytes por pacote, que tem que ser processados pelo sistema embarcado sem interferir com as outras funções do mesmo, pois ele só tem um processador e vai consumir a uma frequência de 200hz uma largura de banda maior que 350kbps, largura de banda que requer a maior prioridade da rede, dada a função do carro.

A função de comunicação desenvolvida é dividida em cinco funções como é mostrado na figura B.4. As funções em azul são do sistema embarcado e as outras são explicadas a seguir:

- **PC Transmist.** encarregada de selecionar qual mensagem vai enviar ao computador, para isso se vale de uma variável chamada `PC_TX_MODE` e de um estado chamado `PC_TX_STATE`, porque diferentes tipos de mensagem no MicroAutobox podem ser enviados de forma sequencial em vários pacotes. No momento está programado de acordo

| PC_TX_MODE | PC_TX_STATE | VALOR | TX_TYPE |
|------------|-------------|------------------|---------|
| 1000 | 0 | Sensor do carro | 1000 |
| | 1 | Sensor do freio | 1100 |
| 2000 | 0 | Estados do Carro | 2000 |

Tabela B.4: Tipos de dados enviados pelo microautobox

| RX_TYPE | VALOR | Método verificação |
|---------|----------------------------|------------------------------------|
| 0 | Modo de operação | o ultimo é o valido |
| 1 | estado do carro | tempo de recebido |
| 20 | Modo manual | troca o modo de operação |
| 30 | Modo Joystick | tempo de recebido + tempo validade |
| 40 | Modo autônomo configuração | tempo de recebido + tempo validade |
| 41 | Modo autônomo global path | contador |
| | contador global_path | troca ao chegar um dado de 41 |
| 41 | Modo autônomo local path | tempo de recebido + tempo validade |
| ... | ... | ... |

Tabela B.5: Tipos de dados enviados pelo microautobox

a tabela B.4.

- **PC Receive.** Ela pega o último pacote válido recebido e o converte no pacote correspondente de acordo com o tipo de dado, para o qual tem memória de todas as mensagens de saída enviadas, porque mesmo que não chegue uma mensagem ele sempre deve garantir as saídas pela forma como foi programado no Simulink. O método de checar as mensagens vai ser de diferentes formas de acordo com a função da mensagem como é explicado na tabela B.5. Nesta tabela é possível ver que algumas variáveis dependem de contadores administrados pela função, outras dependem de um período de validade e outras dependem simplesmente do último valor até chegar um novo.
- **MAUDP.** A função decodifica a mensagem recebida (RX msg(k)) em bytes tratando os arranjos de dados como double, depois ele valida a informação recebida comparando o checksum (com CRC32) recebido com o calculado e validando com ACK para informar que o dado enviado anteriormente RX msg(k-1) esteja correto. Se ocorre um erro, a função vai tentar enviar/receber a mensagem por N vezes como é mostrado na figura B.5 como é visto desde o que está recebendo o problema, pois o sinal tem um atraso (t_1) de uma amostra mais o tempo de transporte e quando ocorre a falha ele vai

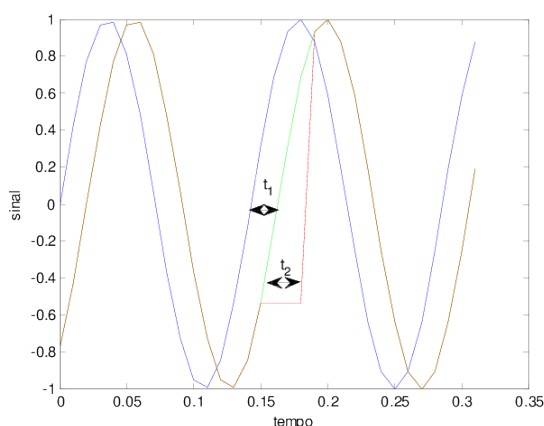


Figura B.5: Comportamento de uma variável recebida no protocolo maudp. Azul sinal onde foi enviada verde(o marrom) sinal sem erros, vermelha sinal com erro.

tentar validar o mesmo dado até um timeout (t_2). Finalmente ele faz o processo contrário codifica a mensagem a enviar (TX msg(k)) e calcula o ACK com o dado recebido (ACK=TX_msg_recebido (k)-TX_msg(k-1)) e finalmente calcula o checksum da mensagem a enviar. Neste caso ele também pode tentar enviar o mesmo dado M vezes, mas pelas características de seu funcionamento sempre vai tentar enviar o último dado.

Nota-se que esta função foi programada em linguagem C e compilada para o Simulink com um compilador visual de Windows e portanto o gerador do código a ser implementado no MicroAutobox do Simulink transforma esta função em linguagem C no código do sistema embarcado.

B.7 Função da máquina de estados

Os modos de operação concebidos para o funcionamento do sistema embarcado são quatro: modo manual (q_2), modo Joystick (q_3), modo Automático (q_4) e modo Cooperativo (q_5), mais um estado inicial (q_1) requerido para começar qualquer outro estado e mais um estado de emergência (q_0). A transição entre eles é um tema aberto e sem única resposta, o qual está gerando até patentes como apresentada pelo Google (CULLINANE *et al.*, 2014). Neste ponto tendo em conta que o sistema embarcado sozinho não vai ter capacidade de percepção do ambiente exterior, só do estado do veículo, nem de planejamento ou cognitivo, o sistema embarcado vai entregar ao usuário o controle do veículo depois de tentar resolver a emergência até onde está no seu alcance. Na mesma ordem de ideias, dado que o modo de operação vai depender do usuário, todos os estados devem partir de um modo manual. Este modo de funcionamento pode ser representado pelo diagrama de estados da figura B.6, onde as transições são comandadas por

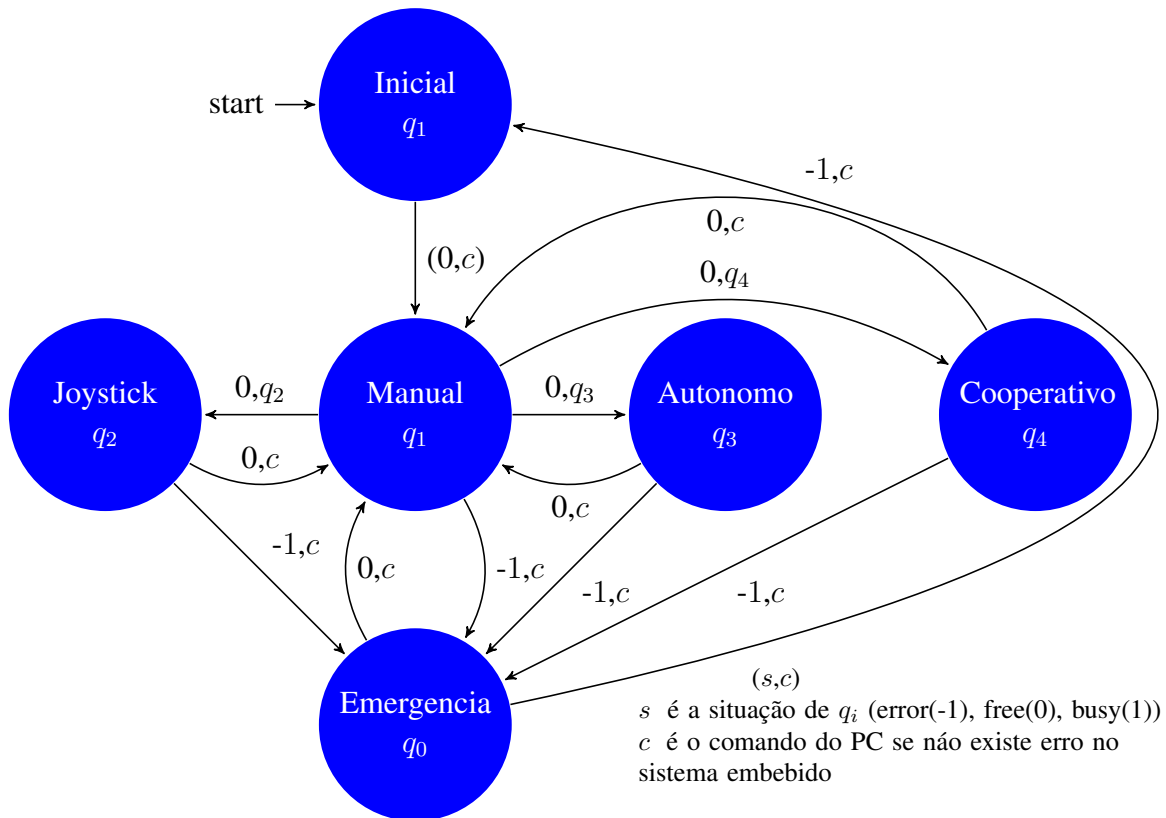


Figura B.6: Estados do sistema embarcado

duas variáveis: a primeira (s), gerada pelo estado, que indica se está em operação e não é seguro trocar de comando (busy), indica se acabou uma tarefa ou é seguro trocar de estado (free) ou indicar se apresenta algum erro (error); a segunda (c) chama o modo de operação solicitado pela comunicação UDP sempre que não tenha sido detetado um erro interno do sistema embarcado ou se o botão de emergência não esteja acionado. Este modo de operação solicitado será então convertido internamente pelo sistema embarcado pelo estado q_0 ou q_2 .

Os erros no sistema embarcado possíveis, listados na tabela B.6, são somados ao estado do sistema para gerar o estado do sistema embarcado (E_{MA}) (item dois do vetor de sensores da tabela B.2) de acordo a equação (B.4)

$$E_{MA} = \sum \text{codigo}_{erro} + 2^{q_i} \quad (\text{B.4})$$

| erro | código |
|---------------------|--------|
| Estado | 512 |
| Recepção do PC | 1024 |
| Transmissão do PC | 2048 |
| Pedal do acelerador | 4096 |
| Direção (TX) | 8192 |
| Direção(RX) | 16384 |
| RX/TX freio | 32768 |
| Estimador de estado | 65536 |
| Sistema de controle | 131072 |

Tabela B.6: Erros no sistema embarcado

APÊNDICE C Software desenvolvido em ROS

Neste Apêndice são apresentados os nós desenvolvidos no ROS da automação e da instrumentação do VILMA01. Utilizando a metodologia de ROS os programas estão divididos em pacotes com o objetivo de agrupá-los de acordo a sua função.

O pacote *vilma_ma_ros* tem a função de comunicar-se com o hardware do veículo, assim como de conter os nós requeridos para testá-lo. Todos os programas têm sido desenvolvidos com controle de versões (https://gitlab.com/vilma_unicamp/vilma_ma_ros) com contrato de licença Apache License Version 2, exceto o nós de conexão ao cabo ODB que tem licença GPL\GNU v2 pelas bibliotecas utilizadas. A explicação das funcionalidades implementadas em cada um dos nós deste pacote são listadas a seguir:

- **vilma_ma_ros_node.** O programa controla a transferência de informação com o sistema embarcado do MicroAutobox. Ele implementa o mesmo protocolo descrito na seção Função da comunicação UDP através da biblioteca BOOST ASIO usando uma comunicação por Socket com timeout para garantir um tempo máximo de processamento menor que a frequência de ciclo f_3 . Ele foi desenvolvido em C++ com os conceitos de programação orientada a objetos em um programa *multithread*. Na primeira *thread* existe uma tarefa cíclica através de um *timer* de ROS que chama uma instância da classe que implementa o protocolo de comunicação. Note-se que este programa funciona como o cliente do protocolo e portanto determina o período do ciclo f_3 do sistema embarcado. O protocolo de comunicação recebe como entrada um dos tópicos recebidos pela outra *thread* e retorna o mensagem recebida do sistema embarcado multiplexando no tópico requerido. Na segunda *thread* são recebidos por meio de uma subscrição a tópicos os mensagens que devem ser enviados, os quais são guardados em fila de dados, para não perder informação e garantir a troca de informação entre as *threads*.

O programa se inscreve a tópicos tipo *Float64Array* chamados *vilma_ma_ros/sensors_pc*, *vilma_ma_ros/joystick_ma* e *vilma_ma_ros/planned_path*. A prioridade de envio é a mesma listada aqui. Finalmente, as mensagens publicados até o momento são *vilma_ma_ros/sensors_ma*, *vilma_ma_ros/state_ma* e *vilma_ma_ros/sensors_brake*.

- **vilma_ma_ros_obd.py** O programa desenvolvido em python utiliza as bibliotecas desenvolvidas para o projeto pyOBD (<https://github.com/roflson/pyobd>) para ler as informações de velocidade do carro, revoluções do motor, porcentagem do pedal e carga do

motor através de um cabo tipo LM324 ligado por USB ao computador. Esta informação é publicada a uma frequência de 1hz no tópico *odb_data* de tipo *Float64Array*.

- ***vilma_ma_cell.py*** (Obsoleto). Este programa faz uma requisição HTTP a um servidor web que roda no celular para solicitar informação sobre o GPS, o acelerômetro, e a orientação do celular. O programa no celular foi desenvolvido em Qpython mas o período do ciclo e a estabilidade da recepção das requisições não eram muito bons.
- ***vilma_qpython_cell.py*** O programa desenvolvido em python é uma segunda versão do nó *vilma_ma_cell.py*, no qual o celular é um cliente UDP, que envia informação codificada em JSON a um tempo de ciclo determinado por ele mesmo ao servidor (o computador) no qual roda este aplicativo. Podem existir perdas de pacotes no envio de dados mas a transmissão é mais rápida e consome menos recursos no celular porque o pacote não requer confirmação para ser liberado, o que tem permitido chegar a tempos de ciclo de até 50hz . No momento o programa do celular, desenvolvido também em Qpython, envia a orientação do celular e a informação do GPS quando está disponível (cada segundo aproximadamente). A informação do GPS é publicada no tópico *cell/fix* de tipo *sensor_msgs/NavSatFix*. Outra funcionalidade que tem o programa é a capacidade de receber mensagens curtas e a frequência de ciclo baixas ($\sim 0,1\text{hz}$), através da subscrição ao tópico *cell_speak*, os quais são enviados a outra *thread* no celular para serem convertidos em voz.

A informação da orientação no momento tem a finalidade de ser usada na localização dado que até o momento, não temos uma unidade AHRS (*attitude and heading reference system*) que seria o indicado para poder ser sincronizado por hardware com outros sensores, portanto com a finalidade de “sincronizar” a informação do GPS u-blox neste programa é publicada a informação da posição global, assim como a orientação do celular no tópico *vilma_ma_ros/sensors_pc*. Para isso este programa recebe os tópicos de posição (*vilma_ublox/fix*) e velocidade (*vilma_ublox/fix_velocity*) do GPS, usando o pacote *message_filter::TimeSynchronizer* para sincronizar a informação. Esta informação é convertida a UTM pela biblioteca *geodesy:utm*. Note-se que esta funcionalidade deve ser temporal, pois a ideia é que a fusão dos sensores seja feita em um programa de localização.

- ***vilma_ma_ros_drive.py*** (Obsoleto) Programa desenvolvido em python usado nas etapas iniciais do sistema quando o MicroAutobox estava no laboratório para monitorar o funcionamento do sistema de forma remota. Para o anterior ele conecta-se aos tópicos */rosout* e *vilma_ma_ros/sensors_ma* para publicar a cada 2 minutos o valor atual do segundo tópico e todas as mensagens do primeiro tópico são armazenados previamente numa fila de

dados numa folha de *Google spreadsheet* . No momento o carro na rua não tem acesso a internet portanto esta funcionalidade está desligada.

- **`vilma_ma_ros_joystick.py`** Este programa desenvolvido em python recebe informação assíncrona do tópico *spacenv/joy* e envia de forma síncrona por um *timer* de ROS a cada *20ms* o tópico *vilma_ma_ros/joystick_ma*. A funcionalidade deste programa é poder controlar os atuadores do veículo (freio, acelerador e direção) pelo *gamepad* ligado ao computador PC. Note-se que os comandos de freio e acelerador enviados são em modo posição e o comando da direção é em modo velocidade. Por segurança no *gamepad* só estarão ativos os comandos ao mover a alavanca requerida (ver manual de funcionamento do apêndice E) ao eixo desejado e pressionar um botão ao mesmo tempo.
- **`vilma_ma_root.launch`** . Este é um arquivo de configuração em XML onde são configurados quais programas devem rodar ao iniciar o PC com o servidor de ROS e com que parâmetros.

Outros pacotes requeridos, desenvolvidos por terceiros e de uso livre, são listados a seguir:

- **`axis_camera package driver`**. Se conecta a uma câmera IP marca axis e publica o tópico *image_raw/compressed* de tipo *sensor_msgs/Image*.
- **`space_nav_node`**. Nó do pacote *joystick_drivers* que lê a informação do Mouse 3D usado e publica os tópicos *spacenv/joy* , *spacenv/offset* , *spacenv/twist* e *spacenv/rot_offset* do movimentos do mesmo.
- **`ublox package driver`** este nó se conecta pela porta USB ao GPS u-blox instalado no veículo. Publica os tópicos *vilma_ublox/fix*, *vilma_ublox/navposllh* , *vilma_ublox/fix_velocity* , *vilma_ublox/navstatus* , *vilma_ublox/navsol* e *vilma_ublox/navvelned* descritos na documentação do pacote. Note-se que o u-blox tem duas portas de comunicação, uma USB e outra serial. Pela porta serial o programa foi muito instável nos testes realizados.
- **`robot_upstart`** . Este pacote é usado para configurar o arquivo de *launch* que deve iniciar com o Ubuntu de tal forma que os nós iniciem de forma automática.

Para ilustrar a interação entre os programas na figura C.1 é apresentado um diagrama dos programas em tempo de execução nos testes de controle do veículo através do *gamepad*. Nesta figura os ovais são os nós em funcionamento, os quadrados pequenos são os nomes dos tópicos

APÊNDICE D Testes da arquitetura de hardware e software da Automação

Cada um dos elementos físicos foi testado como é descrito na tabela D.1 assim como cada uma das funções do sistema embarcado como é mostrado nas tabelas D.2 e D.3. Nestas tabelas é descrita a funcionalidade de acordo ao projetado nos apêndices A e B , é feito uma descrição de como foi testado e é apresentada uma conclusão com os trabalhos futuros em cada uma das funcionalidades da automação.

| Funcionalidade | Descrição teste | Conclusão |
|---------------------------|---|---|
| Sistema mecânico direção | Depois de utilizar o sistema durante um mês foi retirado e montado de novo. | O parafuso de fixação do eixo do motor estava solto e gerava problemas na engrenagem motora, portanto foi adicionado uma arruela para evitar o deslocamento longitudinal do eixo. |
| firmware motor da direção | Em modo programação foram testadas as funcionalidades de zero do motor, controle de posição, controle de velocidade e controle de torque. | Pelo comportamento elástico da roda do carro o zero do motor pode não ser achado corretamente |
| | Em modo depuração foi testada a comunicação com o MicroAutobox | Algumas vezes a comunicação RS232 não conseguia iniciar, portanto o MicroAutobox espera um tempo antes de iniciar o envio de dados ao motor. A entrada análoga tem ruído elétrico, é recomendado filtrar os sinais do motor. |
| Sensor de torque | Processo de calibração descrito na seção 6.1.2 | O sensor tem um comportamento linear, mas o sensor é muito dependente da temperatura. |
| Acelerador | Testes com o carro em neutro e movimento | A voltagem do pedal do usuário não estão em 0% na sua condição zero o qual deve ser levado em conta nas estratégias cooperativas |
| Troca de marcha | Testes com o veículo parado para obter os valores de voltagem e os tempos requeridos | Os comandos são executados corretamente, tempos de resposta 0,1s entre Neutro-Drive-Ré, entre os estados +, - e A/M de 1s. Não existe realimentação de que o comando foi realizado corretamente. |
| Freio | Testes dos sensores da ECU pelas mensagens Broadcast da rede CAN | O sensor de velocidade de uma roda gera falha, problema na ECU. A taxa de transmissão tem desvios padrões menores a 0,05hz |
| | Teste de frenagem liga/desliga através do protocolo UDS da rede CAN | O barulho é muito alto, mas é possível controlar as válvulas e a bomba do freio por PWM com uma frequência portadora 10hz. Não é o método recomendado |
| | Teste de frenagem pela rede CAN e protocolo proprietário OCM | E possível controlar a velocidade do motor e a corrente das válvulas, mas até o momento só funciona com o carro a velocidade zero. |

Tabela D.1: Testes funcionais na automação do veículo

| Funcionalidade | Descrição teste | Conclusão |
|--|---|---|
| Sensores do MicroAutobox | Obtenção de parâmetros de conversão | |
| Comunicação RX RS232 no MicroAutobox | foram enviados diferentes tipos de pacotes do motor de direção | O sistema funciona bem, mas dado que a interrupção é a cada dois bytes, isto consome muitos recursos do MicroAutobox. Deveria ser testado outras arquiteturas de software |
| Comunicação TX RS232 no MicroAutobox | Foram enviados diferentes comandos ao motor a diferentes frequências de ciclo | |
| Troca de marcha no MicroAutobox | Testes na máquina de estados. Figura B.2 | Foi necessário utilizar diferentes tempos de transição entre os estados |
| Acelerador no MicroAutobox | Testes caixa preta(entrada saída) e medições das saídas análogas | diferenças da curva teórica do potenciômetro do pedal |
| Função dos comandos da direção no MicroAutobox | Testes caixa preta(entrada saída) e medições das saídas análogas | Ruido elétrico é detetado no programa do motor |
| Função do freio no MicroAutobox | Foi comparada a informação da rede CAN enviada e recebida | Realizado através do cabo de depuração Vector |
| | Testes na máquina de estados | Foi necessário criar o estado de reset. Existem dúvidas na interpretação dos erros do protocolo UDS |
| Comunicação UDP no MicroAutobox | Testes caixa preta nas funções PC_transmit , PC_receive, MAUDP (Figura B.5) | A verificação com 100% da mensagem gera um atraso de um período de ciclo. |
| | Testes de comunicação com o programa no ROS | Existe um bug conhecido, que o programa de depuração controlDesk , utiliza a mesma interrupção de hardware para TX/RX de dados |

Tabela D.2: Testes funcionais nas funcionalidades do sistema embarcado do veículo

| Testes na máquina de estados | Conclusão |
|-----------------------------------|---|
| Testes na transição do estados | <p>Foi necessário adicionar um wachdog na comunicação UDP, porque em caso de não receber nenhuma informação do sistema Linux ele vai entrar em estado Manual.</p> <p>Em caso de estar pressionado o botão de emergência, se não está no estado de inicialização vai passar a modo manual.</p> <p>Se o estado gera um erro o sistema vai passar a modo manual.</p> |
| Testes no estado de inicialização | <p>Foi necessário realizar um algoritmo para determinar o zero do Encoder, este estado não pode ser comandado pelo botão de emergência.</p> <p>Falta as rotinas de inicialização do freio em caso de requerer trabalhar no modo OCM</p> |
| Testes no estado Joystick | <p>Todas mensagens devem ter um tempo de validade, quando ele é sobrepassado o sistema entra em modo manual.</p> |
| Testes no estado de emergência | <p>Foram realizadas interfaces de depuração que permitem manipular as saídas neste estado.</p> |
| Estado automático e ADAS | <p>No momento os dois dependem dos resultados da interrupção de controle.</p> |

Tabela D.3: Testes funcionais nas funcionalidades do sistema embarcado do veículo

- Ligue o botão chave do sistema.
- Espere terminar a rotina de zero do motor de direção aproximadamente 50s.
- Coloque a marcha na posição de drive.
- Pressione o botão direito do mouse e acelere girando para frente o mouse 3D.
- Para girar a direção faça giros em z no mouse 3D.
- Para frear gire para atrás o mouse 3D.
- Em todo momento que você soltar o botão do mouse o sistema fica só guardando dados, em caso que isso não aconteça aperte o botão de emergência.
- Se detecta alguma coisa errada desligue o sistema automático com o botão do console.
- Por favor esperar aproximadamente um minuto antes de ligar de novo o botão do console.
- Se deseja guardar um banco de dados do percorrido é necessário entrar por ssh ao computador em linux do MicroAutobox e escrever *rosbag record -A*

Note-se que se ligar o sistema automático sem ligar o carro, o zero da direção não pode ser obtido mas o sistema continua funcionando para fazer depuração ou programação. Porém se desligar o carro depois de fazer a rotina de inicialização, o sistema continua funcionando até desligar o botão do console.

E.2 Modo Programação

Nesta seção será apresentada qual é a infraestrutura de hardware/software requerida para programar os diferentes elementos do veículo.

E.2.1 Motor de direção

O motor de direção tem um software e uma linguagem de programação proprietário descrito no documento *Class 5 SmartMotor Users Guide*. O software é chamado SmartMotor Interface e é mostrado na figura E.2. Na parte configuração deve ser configurado a porta serial pela qual está conectado o motor, o segundo permite enviar comandos diretamente ao motor pelo terminal e a terceira parte amostra uma parte do programa realizado para o veículo.

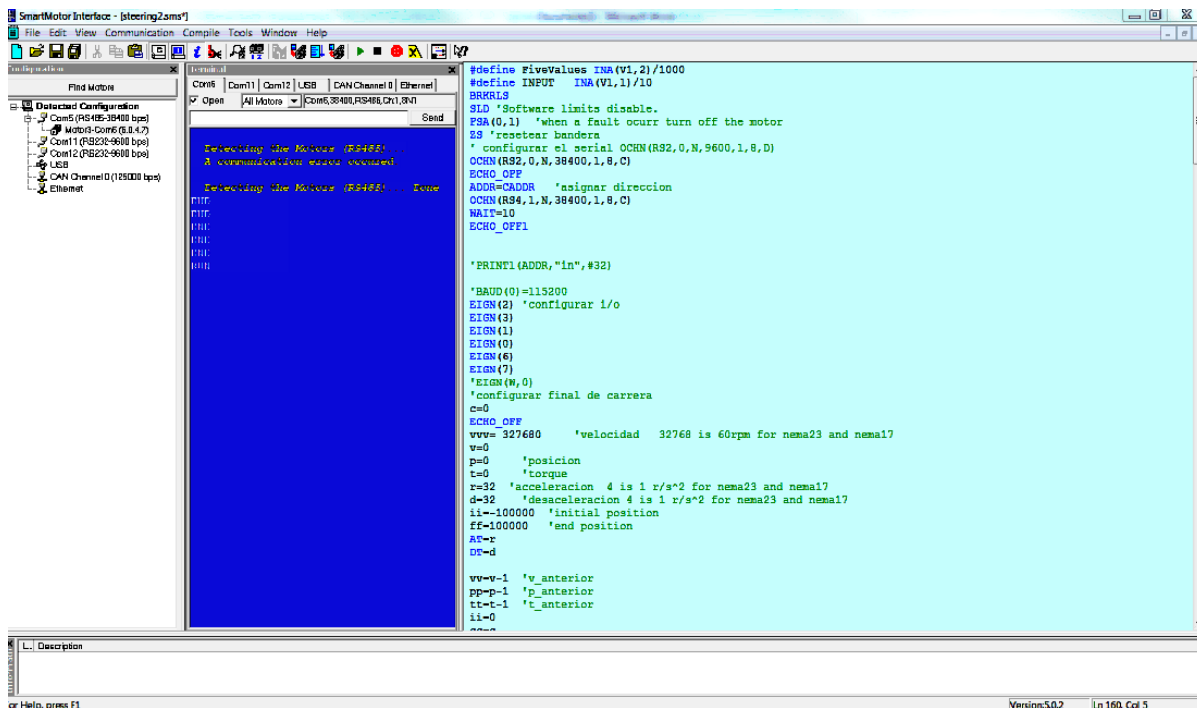


Figura E.2: Interface do programa SmartMotor Interface para programação do motor Animatics

Procedimento para conexão segura ao motor para programação:

- Desligue o cabo do conector 15-Pin D-Sub I/O do motor ou desligue o cabo de energia do MicroAutobox que é de mais fácil acesso.
- Ligue a porta RS232 da porta 7-Pin Combo D-Sub Power & I/O ao computador.
- Ligue o carro e ligue o botão do console do sistema e ligue o botão de emergência.
- No software do motor, *click* direito na porta serial em *properties* e configurar a porta serial de acordo a figura.

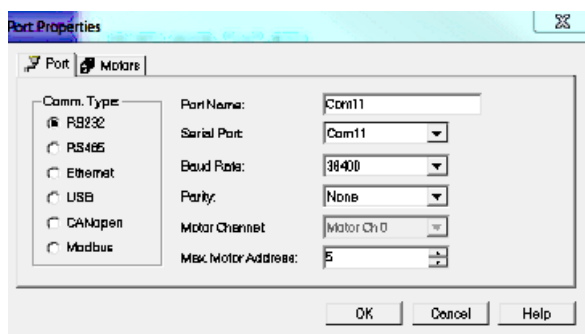


Figura E.3: Configuração porta RS232 no smartMotorInterface

- Click direito na porta serial configurada e dar click em *detect motors on RS-232 chain*. O motor deve ser detetado no canal 3.
- Abrir o programa *steering2.sms* na pasta Animatics e programar de acordo ao manual do motor.

O programa esta composto por 6 funções, a inicial onde se encontra a configuração e a malha infinita, dentro dele esta a rotina de envio e recepção de dados. As outras funções descritas no programa como C10, C11, C12, C13 e C14 realizam as rotinas de controle apagado, controle de velocidade, controle de posição, controle de torque e zero do motor (*Homing*) respetivamente.

- Pressionar F5 para enviar o programa ao motor. Se recomenda fazer o envio do programa com o carro ligado dado que o motor requer mover a direção do carro ao iniciar.

E.2.2 Sistema Embarcado do MicroAutobox

O sistema embarcado do MicroAutobox utiliza Simulink para programação do dispositivo e requer uma chave USB de licença para poder compilar o software. No momento todo o desenvolvido tem sido feito com Matlab 2013a de 32 bits (as bibliotecas de UDP não suportam Matlab de 64 bits) com o software DSpace RCP & HIL 2013A. O procedimento para programar o software é o seguinte:

- Ligue o botão do console do sistema e ligue o botão de emergência. Note-se que por segurança é melhor não ligar o carro.
- Conetar o PC com o software a um ponto da rede do MicroAutobox. Não é recomendado conetar-se por WIFI dado que se apresentam muitos problemas de comunicação.
- **Recomendado.** Conectar-se ao PC em Linux do MicroAutobox por ssh (192.168.140.4 porta 22000) e executar o comando *sudo service vilma stop*
- Compilar o programa no Simulink com Build model (CTRL+B).
- Abrir o programa ControlDesk 5.0 e abrir o projeto *vilma_BETA_Testing*
- Procurar a aba inferior Platform/Device dar click direito no MicroAutobox e seleccionar *Load Real-Time Application to flash*. Procurar o arquivo *sdf* do programa a enviar.
- Normalmente o programa ControlDesk fica desligado do MicroAutobox por aproximadamente 30s, de acordo ao fabricante isso é normal.

- Aperte o botão do console para desligar o sistema, espere ao redor de um minuto e inicie o sistema novamente.

NOTA. O IP do sistema embarcado do MicroAutobox é 192.168.140.3, esta direção responde a *ping* o qual pode ser útil se ocorrer algum problema no programa enviado.

E.2.3 Programas do computador com Linux no MicroAutobox

O computador está rodando Linux Ubuntu server 14.04 com kernel 3.08 e ROS Indigo. O último tentativa de atualizar o kernel do linux gerou um problema com a placa de video do sistema, portanto é necessário ter cuidado antes de atualizar o sistema. O procedimento recomendado para programar algum software rodando no PC desde Windows é o seguinte:

- Ligue o botão do console do sistema e ligue o botão de emergência. Note-se que por segurança é melhor não ligar o carro.
- Conectar o PC com o software a um ponto da rede do MicroAutobox.
- Conectar-se ao PC em Linux do microautobox por ssh (192.168.140.4 porta 22000) usando Putty. O usuário é **robot** e a senha é xxxxxx
- Executar o comando `sudo service vilma stop` . Este passo apaga todos os programas do VILMA
- Pode-se iniciar a modificar os programas. Recomendado abrir o programa WINSOCP para modificar os programas em Windows e no Linux fazer uma conexão sftp.
- Os programas estão localizados em `catkin_lma/vilma_ma_ros`.
- Uma vez terminado de modificar os programas fazer `catkin_make` na interface ssh.
- Se foi modificado o programa `vilma_ma_ros_node` é necessario voltar a gerar as permissões de administrador no executável. Portanto na interface ssh deve-se digitar partindo da `~/catkin_lma` conforme listado a seguir

```

1 cd devel/lib/vilma_ma_ros      # cd to the directory with your node
2 sudo chown root:root vilma_ma_ros_node # change ownship to root
3 sudo chmod a+rx vilma_ma_ros_node    # set as executable by all
4 sudo chmod u+s vilma_ma_ros_node     # set the setuid bit

```

- o Finalmente é necessário configurar os arquivos que rodam ao iniciar o computador através do pacote *robot_upstart* de ROS.

```
1 rosrun robot_upstart install vilma_ma_ros/launch/vilma_ma_root.launch
  --master http://192.168.140.4:11311 --interface eth0
```

E.2.4 Instrumentação do extensômetro da direção (MicroStrain)

Para a configuração e calibração da instrumentação do sensor de torque é utilizado o software *NodeCommander* da empresa *MicroStrain*. Nele é possível configurar as saídas analógicas, assim como toda a informação do sensor ligando a estação base (*WSDA -Base 101*) por USB ao computador. No programa vai aparecer na seção *controller* a estação associada a uma porta serial e como filho dele o nó 263 que corresponde a instrumentação do extensômetro (*SG-Link®-mXRS*).

O nó de aquisição esta programado *low duty cycle* a 128hz.

E.2.5 Sensor de GPS u-blox

Para a configuração e calibração do sensor de GPS abrimos o programa *u-center* da empresa *u-blox*.

E.3 Modo Depuração

O modo depuração se define como acessar ao que esta acontecendo nos programas durante a sua execução. Nesta seção será apresentada qual é a infraestrutura de hardware/software requerida nos elementos do veículo.

E.3.1 Motor de direção

Para ligar-se ao motor no modo de execução , não pode ser usado a porta RS-232 , porque a porta é utilizada para comunicação com o sistema embarcado do MicroAutobox, portanto foi habilitado a porta RS485 que tem o dispositivo para acessar o motor pelo programa do

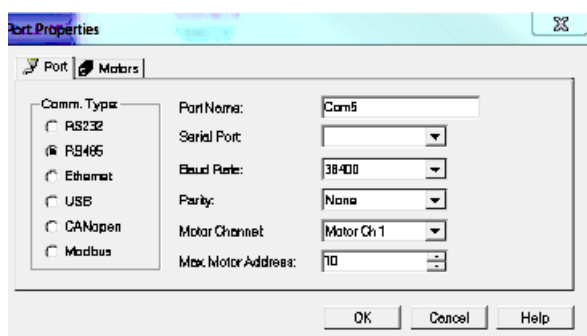


Figura E.4: Configuração porta RS485 no smartMotorInterface

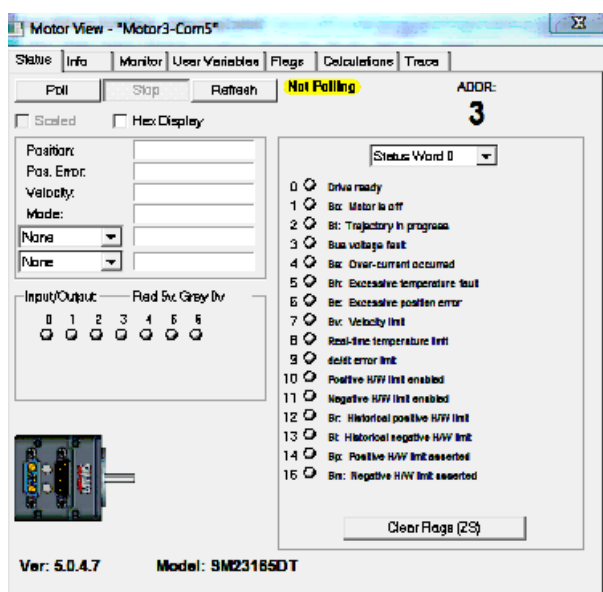


Figura E.5: Monitorar as variáveis do motor no smartMotorInterface

fabricante. O procedimento é o seguinte:

- Ligue o carro e ligue o botão do console do sistema
- Ligue o cabo RS485 a um conversor USB , usando uma resistência de 120Ω entre elas.
- No software do motor, click direito na porta serial em *properties* e configurar a porta serial de acordo a figura E.4.
- Click direito na porta serial configurada e dar click em *detect motors on RS-485 chain*. O motor deve ser detetado no canal 3.
- A interface mais usada para monitorar as variáveis do motor se acessa fazendo *click* direito no motor e pressionar *motor view*, como é mostrado na figura E.5

- Também pode-se acessar as variáveis do motor e modifica-las pelo terminal da porta serial selecionada.

E.3.2 Câmera Axis

Para acessar a câmera axis, existem muitos programas para configura-la mais o mais utilizado é abrir no Chrome o endereço IP 192.168.140.10, ali vai entrar em uma interface onde pode configurar os parâmetros da câmera e visualiza-la no tempo de execução. O usuário deve acessar com username: *root* e a senha *xxxxxxx*.

E.3.3 Sistema Embarcado do MicroAutobox

O procedimento para depurar o software e o seguinte:

- Ligue o carro e ligue o botão do console do sistema.
- Conetar o PC com o software a um ponto da rede do MicroAutobox. Não é recomendado conectar-se por WIFI dado que se apresentam muitos problemas de comunicação.
- Abrir o programa ControlDesk 5.0 (figura E.6) e abrir o projeto *vilma_BETA_Testing*.
- Pressionar *HOME -> GO-ONLINE* . As variáveis dos sensores são listados a direita, depois a informação da máquina de estados e a possibilidade de gerar comandos (em modo emergência) aos atuadores é permitida. Para manipular a máquina de estados desde a interface é necessário por a variável *CD Activate* em 1 e depois selecionar o estado com *Operation Mode CD*.
- Adicionalmente existe uma lista de variáveis que tem o tempo de processamento de cada ciclo de processo, onde períodos maiores a 100us podem gerar problemas no sistema.
- Existe uma interface adicional para monitorar todos os comandos da rede CAN, e uma janela depuração para a informação sobre a função UDP.
- A interface pode ser modificada em tempo de execução para adicionar variáveis ou gravar dados. Para isso abrir a aba variáveis no menu abaixo. De acordo a organização do

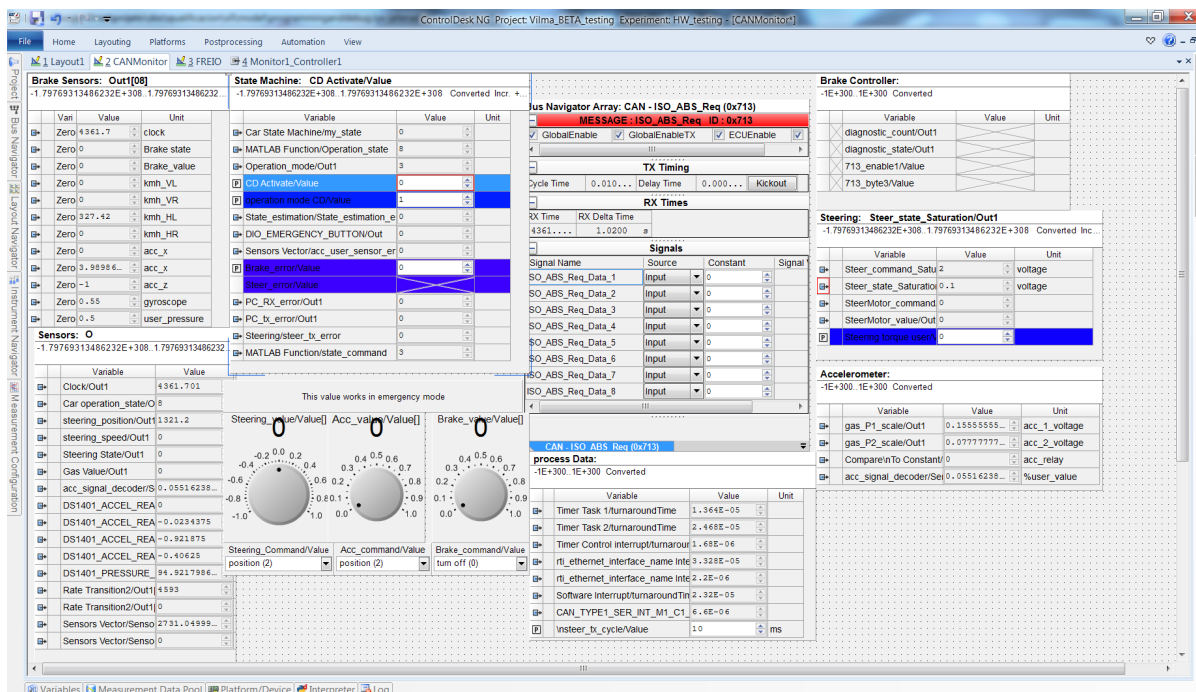


Figura E.6: Programa ControlDesk para depuração do programa do sistema embarcado

programa no Simulink pode-se procurar a variável que se quer monitorar ou em caso de constantes modificar.

- o Existe um **Bug** que o programa do Linux pode deixar de funcionar corretamente se o ControlDesk não for desligado corretamente *GO-OFFLINE*. O mais recomendado é depois de feito a depuração desligar o Controldesk e desligar o PC da rede antes de ligar o sistema novamente.

O **Bug** descrito acontece porque a comunicação UDP com o ROS utiliza a mesma interrupção da comunicação com o CONTROLDESK o anterior gera que o sistema embarcado fique esperando um *timeout*. Se o Controldesk e desligado incorretamente e no programa no Linux não tem conseguido fechar a porta UDP completamente e voltar a abrir algum tempo depois (uns 30s).

E.3.4 Programas do computador com Linux no MicroAutobox

A depuração dos programas no Linux, consiste em observar que acontece com todos os tópicos ou serviços que transmitem a informação entre eles. Para isso o ROS fornece duas tipos de ferramenta, a primeira realizada diretamente no computador através de uma conexão *ssh*, na que temos acesso com *rostopic* a ler (*echo*), obter estatísticas dos tópicos(*hz*, *info*),

ou até publica-los (não recomendado em operação). A segunda opção consiste em acessar a todas as ferramentas de visualização de interação entre programas de forma gráfica, para isso requeremos outro PC com ROS indigo(ou uma máquina virtual) no que faremos o seguinte:

- ligar o PC a rede do MicroAutobox. O melhor é atribuir um IP estático na faixa de 192.168.140.xxx, onde xxx deve ser um valor diferente de 3(sistema embarcado) , 4 (PC do MicroAutobox), 8 (o celular de Olmer) ou 10 (câmera axis). A mascara deve ser 255.255.255. Note-se que se o router está ligado a rede gerará a direção IP requerida por DHCP.
- Depois executaremos a seguinte linha de comandos:

```

1 unset ROS_HOSTNAME
2 export ROS_MASTER_URI=http://192.168.140.4:11311
3 export ROS_IP=IP_DO_PC
4 rqt

```

Nota. Como não existe um servidor de DNS na rede é recomendado definir o *host* do embarcado no arquivo */etc/hosts*

```

1 192.168.140.4 VILMA-PC

```

- Na figura seguinte podemos ver que existem uma grande quantidade de *Plugin* para visualizar a informação em tempo real

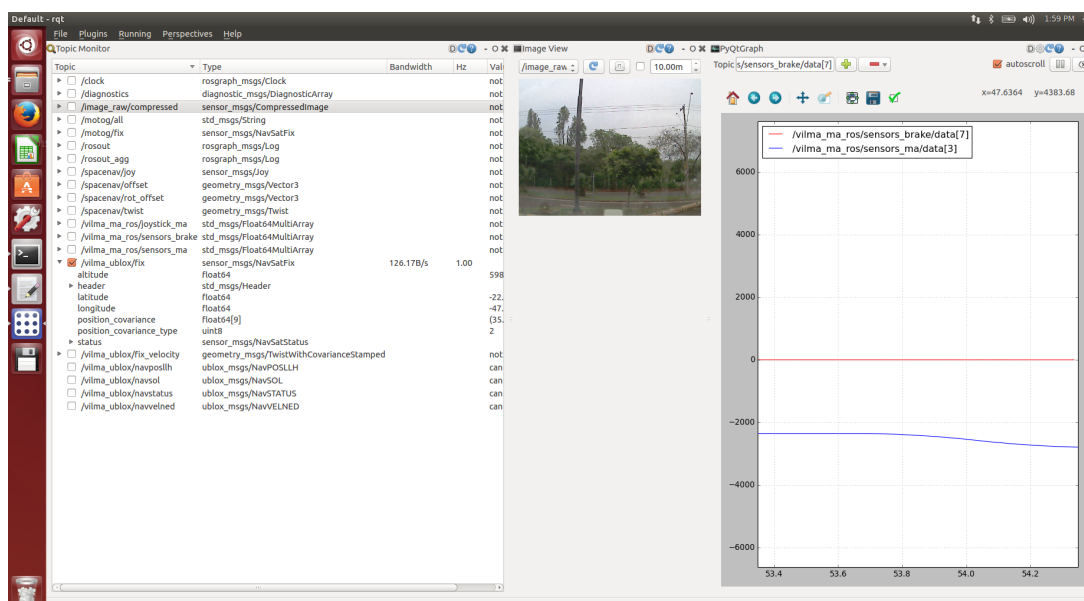


Figura E.7: Programa *rqt* com *plugins* para monitorar informação do ROS

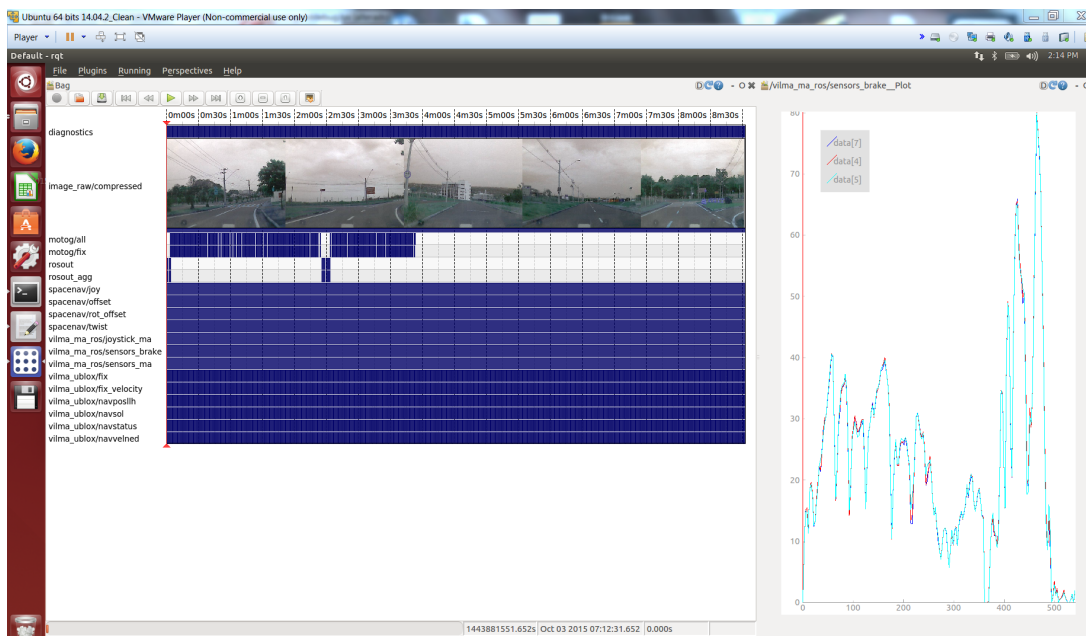


Figura E.8: Programa *rqt_bag* para reprodução de dados adquiridos previamente

- Se requeremos guardar toda a informação de um percurso podemos fazer isso com o comando *rosvag record -A* ou podemos executar o programa *roslaunch vilma_ma_ros vilma_ma_sensors_save.launch*. O arquivo gerado pode ser aberto em rqt pelo plugin rosvag.

Note-se que o mais recomendado é gerar o arquivo *.bag* no PC do MicroAutobox e depois fazer offline a transmissão do arquivo (5min aproximadamente 500MB), porque a capacidade de processamento do PC do MicroAutobox assim como o tráfego de rede podem gerar problemas no sistema.

- Para exportar um tópico em formato csv podemos executar o seguinte comando:

```
1 rostopic echo -b 2015-09-29-09-34-40.bag -p /motog/all >motog_ma.csv
```

- Para exportar as imagens podemos fazer o seguinte:

```
1 rosrn image_view extract_images image:=/image_raw _image_transport:=
  compressed _sec_per_frame:=0.01
```

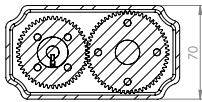
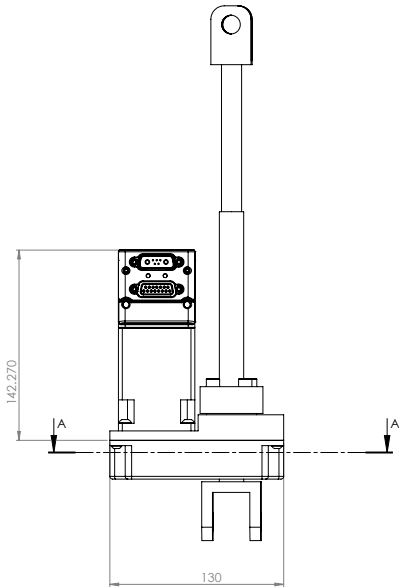
- Para exportar os dados em formatos específicos, por exemplo o formato que recebe <http://www.gpsvisualizer.com/> podemos usar as bibliotecas de *rosvag*, o seguinte programa gera *Waypoints* com a informação do GPS do celular e depois gera uma trajetória com a informação do GPS *u-blox*. Note-se que como o celular envia toda a informação em JSON ela é decodificada no programa, passo não necessário quando é utilizado as mensagens criados com ROS.

```
1 import rosbag import json
2 bag = rosbag.Bag('2015-10-03-11-27-32.bag')
3 print 'type\tlatitude\tlongitude\tname\tdesc'
4 for topic, msg, t in bag.read_messages(topics=['/motog/all']):
5     try:
6         data=json.loads(msg.data)
7         print 'W\t',data["location"]["gps"]["latitude"],'\t',
            data["location"]["gps"]["longitude"],'\t','gps'
8     except ValueError:
9         data=[]
10
11 print 'type\tlatitude\tlongitude\tname\tdesc'
12 for topic, msg, t in bag.read_messages(topics=['/vilma_ublox/fix']):
13     print 'T\t',msg.latitude ,'\t',msg.longitude ,'\t',msg.altitude
14 bag.close()
```

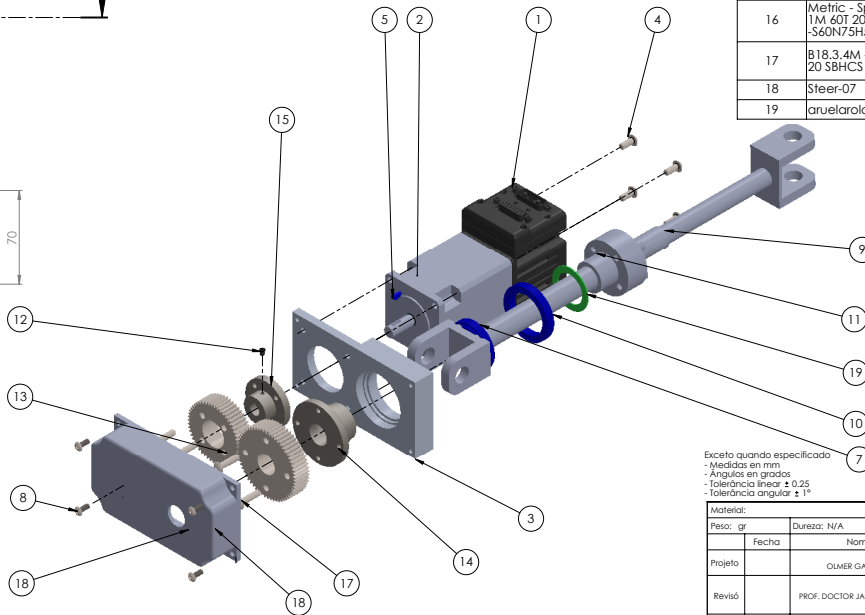
Note-se que o que definimos como depuração não é revisar o programa internamente. Para isso o Linux tem ferramentas como GDB ou Valgrind.

Alguns momentos pode ser necessário ligar o computador com Linux a internet para o anterior podemos usar o cartão sem fio dele. Para isso temos que usar o programa wpa_supplicant do Linux

APÊNDICE F Desenho técnico da automação da direção



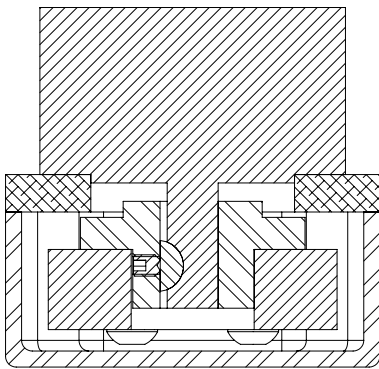
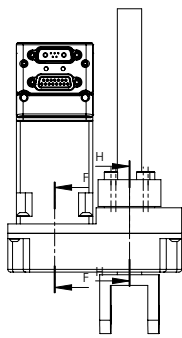
SECTION A-A



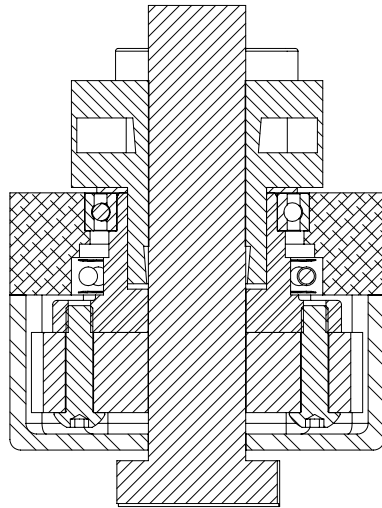
Exceto quando especificado
 - Medidas em mm
 - Ângulos em graus
 - Tolerância linear ± 0,25
 - Tolerância angular ± 1°

| Nº DO ITEM | Nº DA PEÇA | DESCRIÇÃO | QTD. |
|------------|--|--|------|
| 1 | SM23165x | SM23165D and SM23165DT | 1 |
| 2 | gh23sp016 | gearhead animatics OEM nema 23 16:1 | 1 |
| 3 | portamotor | Porta motor | 1 |
| 4 | B18.3.4M - 5 x 0.8 x 12 SBHCS --N | | 4 |
| 5 | AI KEYS Woodruff R B17.2-303 | | 1 |
| 7 | 61806-2RS1 | Rolamento D=42, d=30, w=7 | 1 |
| 8 | B18.3.4M - 4 x 0.7 x 10 SBHCS --C | | 4 |
| 9 | bratransmission | | 1 |
| 10 | 61807-2RS1 | Rolamento D=47, d=35, w=7 | 1 |
| 11 | ANEL DE FIXAÇÃO ATI A02 (SIT LOCK 3) 18X26 | Sistema conico (são 3 peças+4 parafusos) | 1 |
| 12 | B18.3.6M - M4 x 0.7 x 5 Hex Socket Type I Cup Pl. SS --C | | 1 |
| 13 | Metric - Spur gear 1M S2T 20PA 15FW --S52N75HS0L24N | | 1 |
| 14 | eixosteering | | 1 |
| 15 | eixomotorv2 | | 1 |
| 16 | Metric - Spur gear 1M 60T 20PA 15FW --S60N75HS0L18N | | 1 |
| 17 | B18.3.4M - 5 x 0.8 x 20 SBHCS --C | | 8 |
| 18 | Steer-07 | Tampa Motor | 1 |
| 19 | aruelarolamento | | 1 |

| | | | | | |
|-------------|--------------------------------------|--------------------------|---|---------------|--------|
| Materia: | Dureza: N/A | Acabado superficial: N/A | UNIVERSIDADE ESTADUAL DE CAMPINAS Laboratório de Mobilidade Autônoma | | |
| Peso: gr | Nome | Firma | | | |
| Projeto | OLIVER GARCIA | | Automatização Fiat Punto | | |
| Revisão | PROF. DOCTOR JANITO VAQUERO | | | | |
| Escala: 1:2 | Nome: Conjunto motor direção | | Ref. steer-01 | Quantidade: 1 | Rev. 1 |
| A2 | Pertence a: Automatização Fiat Punto | | Nº. Plano: steer-01 | Hoja 1 de 2 | |



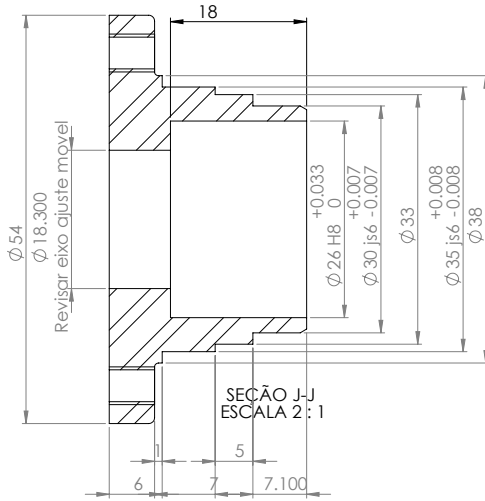
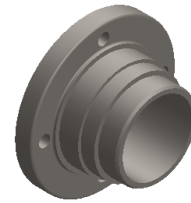
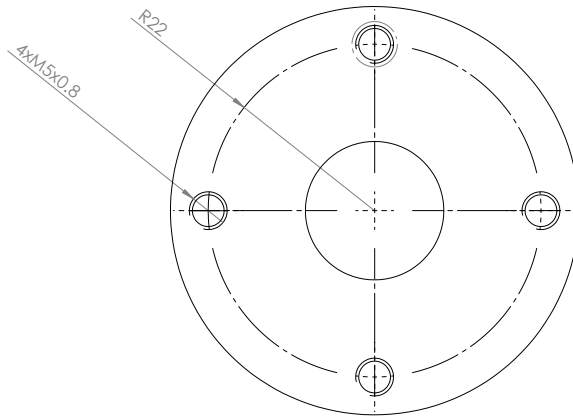
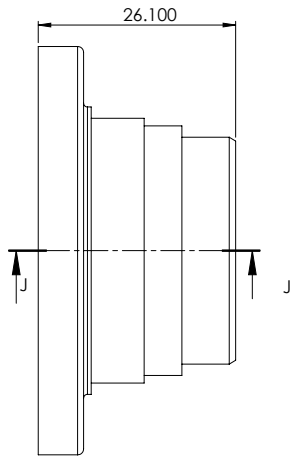
SECÃO F-F
ESCALA 2 : 1



SECÃO H-H
ESCALA 2 : 1

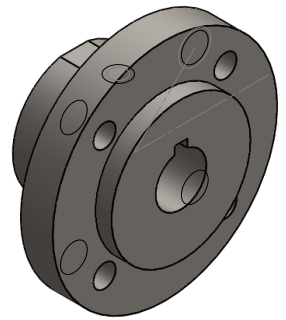
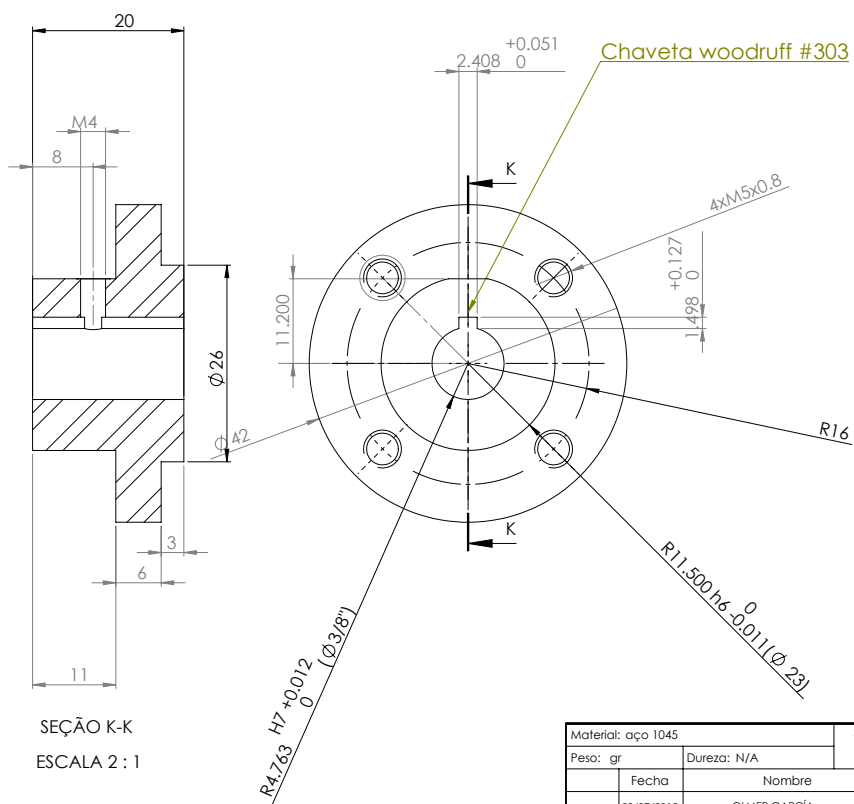
Exceto quando especificado
 - Medidas em mm
 - Ângulos em graus
 - Tolerância linear $\pm 0,25$
 - Tolerância angular $\pm 1^\circ$

| Material: | | Acabado superficial: | | UNIVERSIDADE ESTADUAL DE CAMPINAS | | |
|-------------|--------------------------------------|----------------------|--|------------------------------------|-------------|--------|
| Peso: gr | Dureza: N/A | N/A | | Laboratório de Mobilidade Autônoma | | |
| Fecha | Nome | Firma | | | | |
| Projeto | OLIVER GARCIA | | | Automatização Fiat Punto | | |
| Revisó | PROF. DOCTOR JANITO VAQUERO | | | | | |
| Escala: 1:2 | Nome: Conjunto motor direção | | | Ref. steer-01 | Quantid.: 1 | Rev. 1 |
| A2 | Pertence a: Automatização Fiat Punto | | | Nº. Plano: steer-01 | Hoja 2 de 2 | |



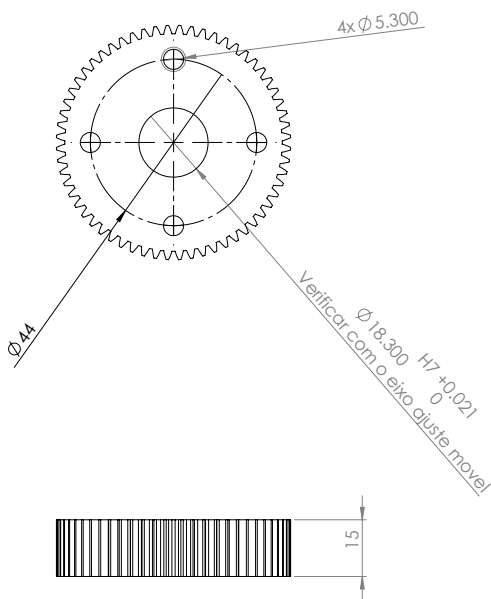
Exceto quando especificado
 - Medidas em mm
 - Ângulos em graus
 - Tolerância linear $\pm 0,25$
 - Tolerância angular $\pm 1^\circ$

| | | | | | | |
|--------------------|-------------|------------------------------------|-------|---|-------------|--------|
| Material: aço 1045 | | Acabado superficial: N/A | | UNIVERSIDADE ESTADUAL DE CAMPINAS Laboratório de Mobilidade Autônoma | | |
| Peso: gr | Dureza: N/A | | | | | |
| Projeto | 22/07/2012 | Nome OLMER GARCIA | Firma | Automatização Fiat Punto | | |
| Revisó | | PROF. DOCTOR JANITO VEQUEIRO | | | | |
| Escala: | 1:1 | Nome: | | Ref. Steer-03 | Cantidad: 1 | Rev. 1 |
| A3 | | Pertence a: Conjunto motor direção | | Nº. Plano: Steer-03 | Hoja 1 de 7 | |



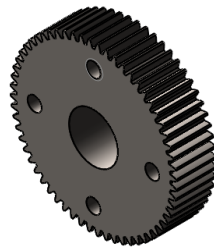
Exceto quando especificado
 - Medidas em mm
 - Ângulos em graus
 - Tolerância linear ± 0.25
 - Tolerância angular $\pm 1^\circ$

| | | | | | |
|--------------------|-------------|---------------------------------|------------------------|---|--------------------------|
| Material: aço 1045 | | Acabado superficial: N/A | | UNIVERSIDADE ESTADUAL DE CAMPINAS Laboratório de Mobilidade Autônoma | |
| Peso: gr | Dureza: N/A | | | | |
| Projeto | 22/07/2012 | Nome | OLMER GARCIA | Firma | |
| Revisó | | PROF. DOCTOR JANITO VEQUEIRO | | | Automatização Fiat Punto |
| Escala: | 1:1 | Nome: | | Ref. Steer-03 Steer-03 | Quantid.: 1 |
| A3 | | Perfenece a: | Conjunto motor direção | Nº. Plano: Steer-03 | Hoja 2 de 7 |
| | | | | Rev. 1 | |



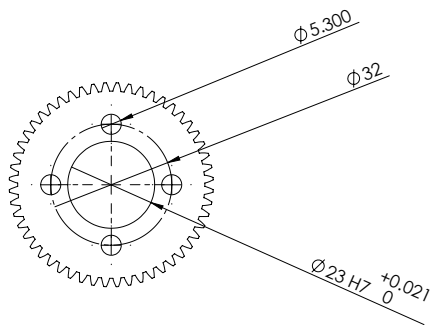
| | |
|--|---------------------------------|
| Número de dentes | 60 |
| norma DIN782 aço 1045 | |
| Módulo | 1 |
| Ângulo de hélice | 0° |
| Ângulo de pressão | 20° |
| Passo diametral | 60 |
| Diâmetro externo | 62 |
| Diâmetro raiz | -- |
| Excentricidade máxima entre el par de engranajes | 0,05 |
| Tratamiento térmico | Cementado profundidade de 0.2mm |

Feito a partir de engranagem padrão
 CÓDIGO BE401010060
 Proveedor A.T.I Brasil



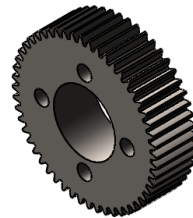
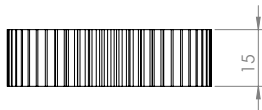
Exceto quando especificado
 - Medidas em mm
 - Ângulos em grados
 - Tolerância linear ± 0.25
 - Tolerância angular $\pm 1^\circ$

| | | | | | | |
|----------|------------|-------------------------------------|--|--------------------------------------|-------------|--------|
| Materia: | | Acabado superficial: | | UNIVERSIDADE ESTADUAL DE CAMPINAS | | |
| Peso: gr | | Dureza: N/A | | | | |
| Projeto | 22/07/2012 | Nome: OLIVER GARCÍA | | Firma | | |
| Revisó | | PROF. DOCTOR JANITO VEQUEIRO | | Automatização Fiat Punto | | |
| Escala: | 1:1 | Nombre: | | Ref. Steer-03 | Cantidad: 1 | Rev. 1 |
| A3 | | Pertenece a: Conjunto motor direção | | Nº. Plano: Steer-03 | Hoja 3 de 7 | |



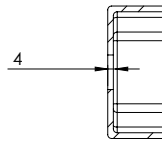
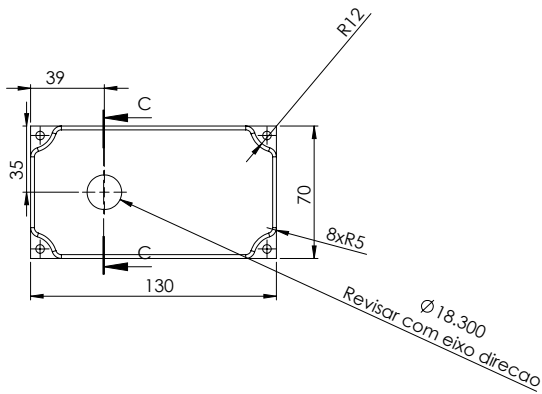
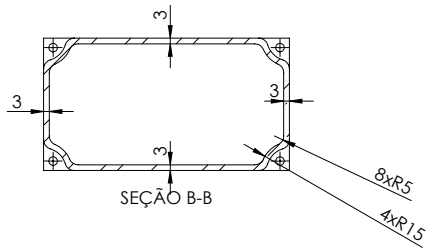
| | |
|--|--|
| Número de dentes | 52 |
| Norma Din782 aço 1045 | |
| Módulo | 1 |
| Ângulo de hélice | 0° |
| Ângulo de pressão | 20° |
| Passo diametral | 52.00 |
| Diâmetro externo | 54 |
| Diâmetro raiz | -- |
| Excentricidade máxima entre el par de engranajes | 0.05 |
| Holgura entre dientes | + 0.060 +0.020 |
| Tratamiento térmico | Cementado profundidad e de 0.2mm |

Feito a partir de engranagem padrão
 CÓDIGO BE401010052
 Proveedor A.T.I Brasil

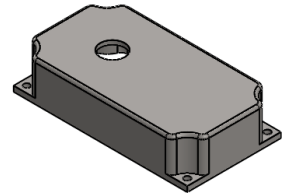


Exceto quando especificado
 - Medidas en mm
 - Ángulos en grados
 - Tolerância linear ± 0.25
 - Tolerância angular $\pm 1^\circ$

| | | | | |
|-----------|-------------|-------------------------------------|---------------------------|---|
| Material: | | Acabado superficial: N/A | | UNIVERSIDADE ESTADUAL DE CAMPINAS |
| Peso: gr | Dureza: N/A | Nome | Firma | |
| Projeto | 22/07/2012 | OLMER GARCÍA | | Laboratório de Mobilidade Autônoma |
| Revisó | | PROF. DOCTOR JANITO VEQUEIRO | | Automatização Fiat Punto |
| Escala: | 1:1 | Nome: | Ref. Steer-03 Steer-03 | Quantidad: 1 |
| A3 | | Pertenece a: Conjunto motor direção | Nº. Plano: Steer-03 | Rev. 1 Hoja 4 de 7 |

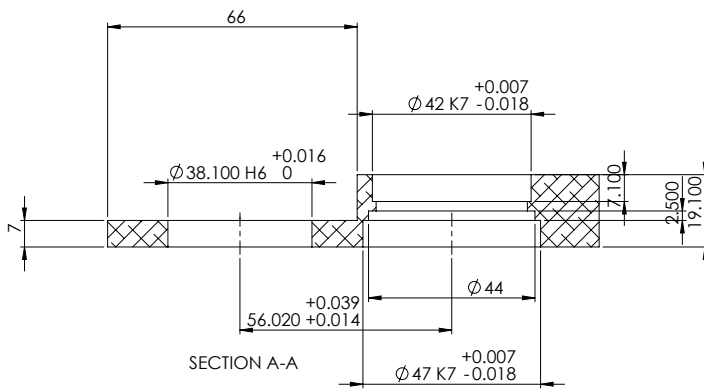
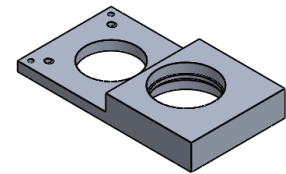
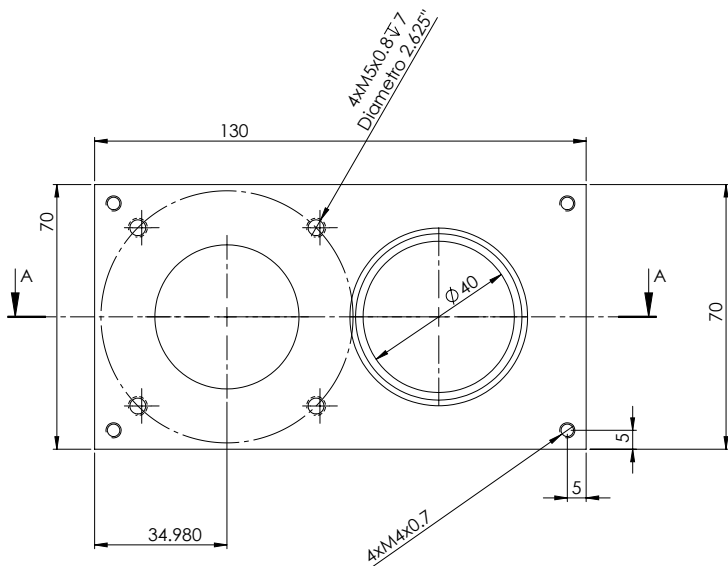


SEÇÃO C-C



Exceto quando especificado
 - Medidas em mm
 - Ângulos em graus
 - Tolerância linear ± 0.25
 - Tolerância angular $\pm 1^\circ$

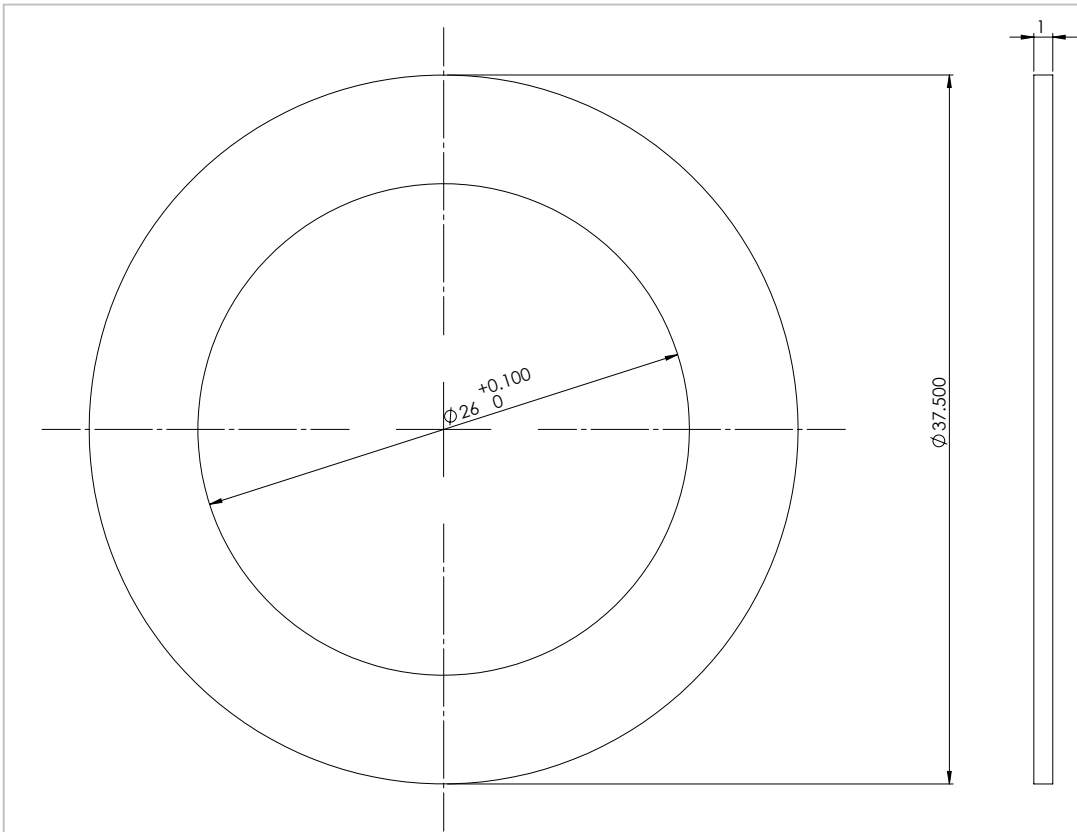
| | | | | | | |
|--------------------|-------------|--------------------------|------------------------------|---|---------------|--------|
| Material: Alumínio | | Acabado superficial: N/A | | UNIVERSIDADE ESTADUAL DE CAMPINAS Laboratório de Mobilidade Autônoma | | |
| Peso: gr | Dureza: N/A | | | | | |
| Projeto | 22/07/2012 | Nome | OLMER GARCIA | Firma | | |
| Revisó | | Nome: | PROF. DOCTOR JANITO VEQUEIRO | | | |
| Escala: | 1:2 | Nome: | | Ref. Steer-03 | Quantidade: 1 | Rev. 1 |
| A3 | | Perfenece a: | Conjunto motor direção | Nº. Plano: Steer-03 | Hoja 5 | de 7 |



NOTA
 1: Em rolamentos Radio max 0.2
 2: Redondeo de radio 2mm máx.

Exceto quando especificado
 - Medidas em mm
 - Ângulos em graus
 - Tolerância linear ± 0.25
 - Tolerância angular $\pm 1^\circ$

| | | | | |
|---------------------------|--|------------------------------|-------------|---|
| Material: Aluminio | | Acabado superficial: N/A | | UNIVERSIDADE ESTADUAL DE CAMPINAS |
| Peso: gr | Dureza: N/A | | | |
| Projeto | 22/07/2012 | Nome: OLIVER GARCÍA | Firma: | Laboratório de Mobilidade Autônoma |
| Revisó | | PROF. DOCTOR JANITO VEQUEIRO | | Automatização Fiat Punto |
| Escala: 1:1 | Nome: | Ref. Steer-03 | Cantidad: 1 | Rev. 1 |
| A3 | Perfenece a: Conjunto motor direção | Nº. Plano: Steer-03 | Hoja 6 de 7 | |



Exceto quando especificado
 - Medidas em mm
 - Ângulos em graus
 - Tolerância linear ± 0.25
 - Tolerância angular $\pm 1^\circ$

| | | | | | | | |
|-----------|-------------|-------------------------------------|--|----------------------|-------------|---|--|
| Material: | | Aluminio | | Acabado superficial: | | N/A | |
| Peso: gr | Dureza: N/A | | | | | UNIVERSIDADE ESTADUAL DE CAMPINAS Laboratório de Mobilidade Autônoma | |
| Projeto | Nome | Firma | | | | | |
| Projeto | 22/07/2012 | OLMER GARCIA | | | | Automatização Fiat Punto | |
| Revisó | | PROF. DOCTOR JANITO VEQUEIRO | | | | | |
| Escala: | 1:1 | Nome: eixo Steering | | Ref. Steer-03 | Cantidad: 1 | Rev. 1 | |
| A3 | | Pertenece a: Conjunto motor direção | | Nº. Plano: Steer-03 | Hoja 7 de 7 | | |

APÊNDICE G Modelos Matemáticos do veículo

Neste anexo são apresentados os modelos matemáticos do veículo usados no desenvolvimento dos algoritmos de planejamento e controle. Na primeira parte é apresentado o modelo cinemático e na seguinte é apresentado o modelo dinâmico do veículo.

G.1 Modelo Cinemático do Veículo

A equação do movimento para o modelo de bicicleta cinemática assume que os pneus da esquerda e da direita são uma única roda no centro dos eixos dianteiro e traseiro, como se mostra na figura G.1. Também é assumido que os pneus não tem nenhum deslizamento lateral e que só a roda dianteira é orientável (δ_f). Restringindo o modelo para o movimento num plano, as equações de restrição não-holonômicas (o que significa que não pode deslizar, de modo que os pneus não podem mover-se na direção normal) para os pneus dianteiros e traseiros (SNIDER, 2009) é dado pelas equações (G.1) e (G.2)

$$\dot{x}_f \sin(\theta + \delta_f) - \dot{y}_f \cos(\theta + \delta_f) = 0 \quad (\text{G.1})$$

$$\dot{x} \sin(\theta) - \dot{y} \cos(\theta) = 0 \quad (\text{G.2})$$

na qual (x, y) é a posição do centro da roda traseira no sistema de coordenadas global, (x_f, y_f) é a posição do centro da roda dianteira no sistema de coordenadas global e θ é a orientação do veículo com respeito ao eixo x no sistemas de coordenadas global. Como a roda dianteira é localizada a uma distância L da roda traseira, (x_f, y_f) pode ser descritos como:

$$x_f = x + L \cos(\theta_p) \quad (\text{G.3})$$

$$y_f = y + L \sin(\theta_p) \quad (\text{G.4})$$

obtendo (\dot{x}_f, \dot{y}_f) de (G.3) e (G.4) e substituindo (\dot{x}_f, \dot{y}_f) em (G.1) tem-se:

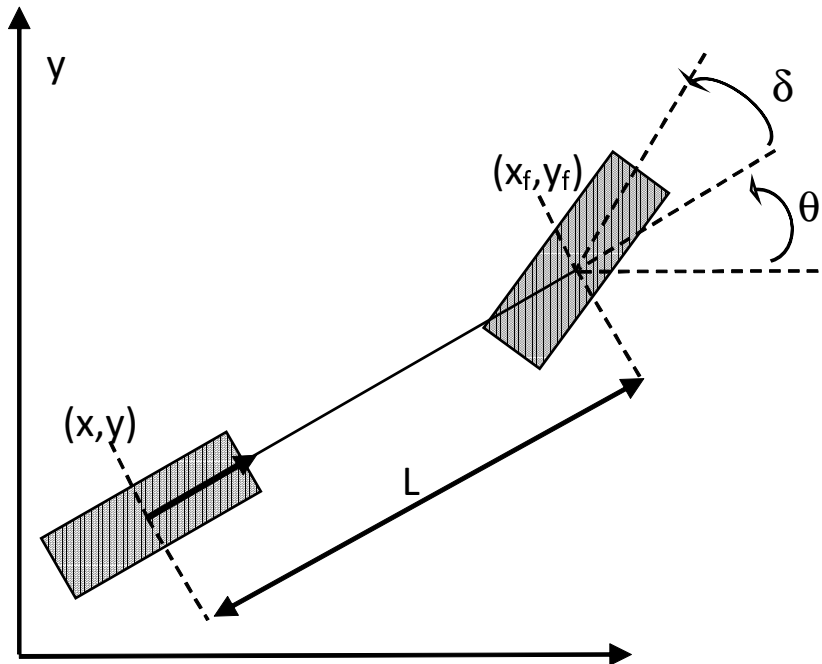


Figura G.1: Desenho do modelo cinemático da bicicleta

$$\dot{x} \sin(\theta_p + \delta_f) - \dot{y} \cos(\theta_p + \delta_f) - \dot{\theta} L \cos \delta_f = 0 \quad (\text{G.5})$$

A restrição não holonômica da roda traseira, equação (G.2) é satisfeita quando

$$\dot{x}_r = v_x \cos \theta_p \quad (\text{G.6})$$

$$\dot{y}_r = v_x \sin \theta_p \quad (\text{G.7})$$

nas quais v é qualquer número escalar que corresponda a velocidade longitudinal. Aplicando (G.6) e (G.7) a (G.5) é possível achar a solução de $\dot{\theta}_p$:

$$\dot{\theta}_p = \frac{v_x \tan \delta_f}{L} \quad (\text{G.8})$$

Portanto, as equações (G.6), (G.7) e (G.8) descrevem as variáveis de estado no sistema de coordenadas global com respeito ao centro da roda traseira do veículo. Em forma matricial temos o seguinte sistema de equações de estado não linear (G.9)

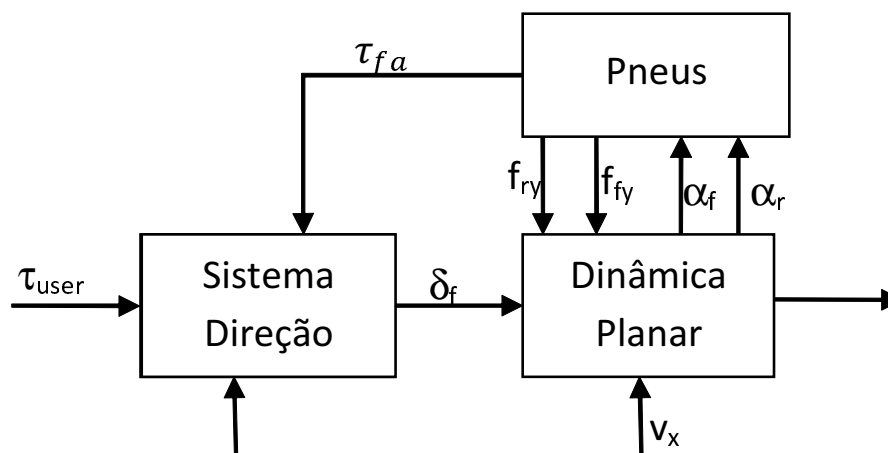


Figura G.2: Diagrama de blocos do modelo dinâmico do veículo

$$\dot{\mathbf{x}}_{\text{ref}} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_p \end{bmatrix} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ r_p \end{bmatrix} = \begin{bmatrix} v_x \sin \theta_p \\ v_x \sin \theta_p \\ \frac{v_x \tan \delta_f}{L} \end{bmatrix} \quad (\text{G.9})$$

Neste modelo temos como entrada a velocidade linear e o ângulo de direção.

G.2 Modelo Dinâmico do Veículo

O modelo dinâmico do veículo é dividido em três blocos (Figura G.2), baseado em (AVAK, 2004) que tem três entradas: o torque do condutor τ_{user} , a voltagem do motor de direção v_m e a velocidade longitudinal do veículo v_x . Os blocos são: dinâmica planar do veículo, modelo dos pneus e modelo da direção e serão apresentados a seguir.

G.2.1 Dinâmica Planar do Veículo

Dois tipos de modelo foram estudados sobre a dinâmica planar do veículo, o primeiro é baseado no modelo de bicicleta que assume que os pneus da esquerda e da direita são uma única roda no centro dos eixos dianteiro e traseiro (SNIDER; YIH; AVAK; YIH *et al.*, 2009; 2005; 2004; 2004) e segundo baseado em (DOUMIATI *et al.*; DOUMIATI *et al.*, 2008; 2010) que utiliza um veículo de quatro rodas simplificado. O primeiro foi selecionado como modelo de simulação, devido a problemas numérico na simulação no MicroAutobox do segundo modelo, portanto a seguir é explicado o modelo dinâmico de bicicleta que representa o veículo.

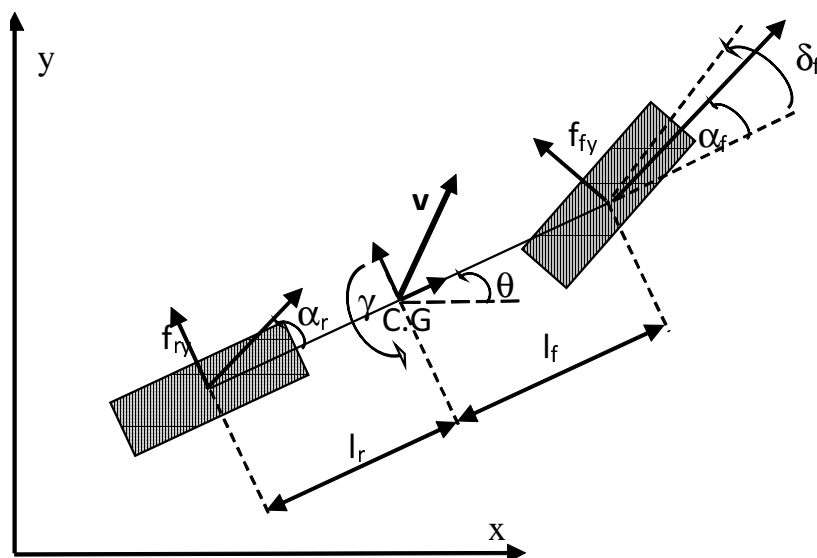


Figura G.3: Desenho da dinâmica planar da bicicleta

Assumindo que o veículo é um só corpo rígido, as acelerações angulares sobre o eixo y e z (*pitch* e *roll*) são desprezíveis e a aceleração longitudinal y vertical é zero. As forças que atuam sobre o veículo são mostradas na figura G.3. As forças externas atuando sobre o veículo são as forças laterais dos pneus com o chão f_{fy} e f_{ry} , portanto aplicando a segunda lei de Newton:

$$m_v \mathbf{a}_y = \cos \delta_f f_{fy} + f_{ry} \quad (\text{G.10})$$

na qual, m_v é a massa do veículo e \mathbf{a}_y é o vetor de aceleração lateral mais a aceleração de Coriolis causada pela velocidade de rotação do corpo ($\mathbf{a} = \gamma \mathbf{x} \mathbf{v} + \dot{\mathbf{v}}$) descrita pela equação (G.11)

$$\mathbf{a}_y = v_x \gamma + \dot{v}_y \quad (\text{G.11})$$

na qual γ é a velocidade angular do veículo. Juntando (G.10) e (G.11) temos:

$$m(\dot{v}_y - \gamma v_x) = \cos \delta_f F_{fy} + f_{ry} \quad (\text{G.12})$$

A segunda equação dinâmica sai da rotação do eixo z aplicando a segunda lei de Newton da aceleração angular com respeito ao centro de gravidade (CG).

$$J_v \dot{\gamma} = f_{fy} l_f \cos \delta_f - f_{ry} l_r \quad (\text{G.13})$$

na qual J_v é o momento de inércia do veículo com respeito ao centro de massa, l_f é a distância do CG a eixo dos pneus dianteiras e l_r é a distancia do CG a eixo dos pneus traseiros.

Nas equações anteriores v_x e v_y estão no sistema de coordenadas local, portanto se eles são transladados ao sistema global de coordenadas temos (PEPY *et al.*, 2006):

$$\dot{x} = v_x \cos \theta_c - v_y \sin \theta_c \quad (\text{G.14})$$

$$\dot{y} = v_x \sin \theta_c + v_y \cos \theta_c \quad (\text{G.15})$$

$$\dot{\theta} = \gamma \quad (\text{G.16})$$

Fazendo x, y, θ, v_x e γ as variáveis de estado; v_x, F_{ry}, F_{fy} e δ_f as entradas e (G.13), (G.12), (G.14), (G.15) e (G.16) as equações de estado em forma matricial temos a seguinte sistema de equações de estado não linear :

$$\dot{\mathbf{x}}_{\text{car}} = \begin{bmatrix} \dot{v}_y \\ \dot{\gamma} \\ \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{\cos \delta_f f_{fy} + f_{ry}}{m_v} + \gamma v_x \\ \frac{f_{fy} l_f \cos \delta_f - f_{ry} l_r}{J_v} \\ v_x \cos \theta - v_y \sin \theta \\ v_x \sin \theta + v_y \cos \theta \\ \gamma \end{bmatrix} \quad (\text{G.17})$$

Além disso, como é apresentado na figura G.2 duas variáveis de saída do modelo de carro são necessários para o modelo dos pneus. O ângulo de deslizamento frontal α_f é definido como o ângulo entre a linha de centro do pneu dianteiro e a direção da velocidade do pneu dianteiro, e o ângulo de deslizamento traseiro α_r é definido como o ângulo entre a linha de centro da pneu traseiro e a direção da velocidade do pneu traseiro. Com respeito as variáveis de estado temos:

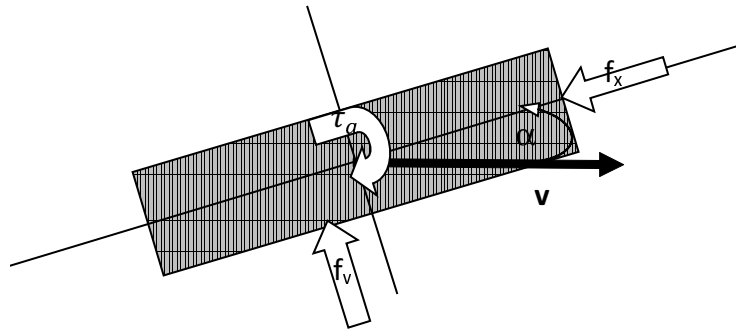


Figura G.4: Ângulos de derrapagem na rodas

$$\alpha_f = \text{atan} \left(\frac{-v_y + l_f \gamma}{v_x} \right) - \delta_f \quad (\text{G.18})$$

$$\alpha_r = \text{atan} \left(\frac{-v_y + l_r \gamma}{v_x} \right) \quad (\text{G.19})$$

Se os ângulos de deslizamento são pequenos podemos aproximar por (G.20) e (G.21)

$$\alpha_f = \frac{-v_y + l_f \gamma}{v_x} - \delta_f \quad (\text{G.20})$$

$$\alpha_r = \frac{-v_y + l_r \gamma}{v_x} \quad (\text{G.21})$$

G.2.2 Modelo dos Pneus

Os pneus são, talvez, o componente de maior influência para a dinâmica planar de um veículo e o mais difícil de modelar porque além de suportar o veículo fora da influencias do amortecimento e das irregularidades da estrada, os pneus proporcionam as forças longitudinais (f_x) e laterais (f_y) necessários para alterar a velocidade e a direção do veículo, as quais por sua vez geram um torque de alinhamento (τ_f). Estas forças são produzidas pela deformação do pneu que entra em contato com a estrada durante a aceleração, frenagem e curvas. Estimadores de forças laterais e longitudinais foram estudados no trabalho (REYES *et al.*, 2015).

Muitos modelos são propostos na literatura para representar essa interação, desde aborda-

gens físicas baseados em mola-amortecedor à abordagens empíricas como são as fórmulas de Pacejka (PACEJKA E BESSELINK, 1997) e o modelo de Dugoff (DUGOFF *et al.*, 1970). No trabalho foi utilizado o modelo de pneu Pacejka para simulação onde não há princípios físicos específicos para a estrutura das equações escolhidas, mas elas se encaixam a uma grande variedade de construções de pneus e condições de funcionamento. Para a força lateral e o momento de alinhamento (figura G.4) duas variáveis são necessários, o ângulo de deslizamento, que é o ângulo entre a sua direção de movimento (do vetor velocidade v) e o plano da roda, e a força vertical sobre os pneus (f_z) que podem ser calculadas por meio da dinâmica vertical do veículo como é mostrado em (DOUMIATI *et al.*, 2008). Estimadores de dinâmica vertical do veicular foram estudados no trabalho (DAIHA *et al.*, 2015).

As saídas deste bloco, que são as forças laterais e gráficas que apresentam o comportamento da força com respeito a torque de alinhamento da roda dianteira, são mostradas nas figuras G.5 e G.6 calculadas usando os parâmetros utilizados em (YIH E GERDES, 2005). Sobre os gráficos é possível observar que as forças podem ser aproximadas a linhas retas para ângulos de deslizamento ao redor de 3° e, portanto, nesta faixa as forças laterais e o torque de alinhamento podem ser aproximados por (G.22), (G.23) e (G.24)

$$f_{fy} \cong c_f f_z \alpha_f \quad (\text{G.22})$$

$$f_{ry} \cong c_r f_z \alpha_r \quad (\text{G.23})$$

$$T_{fa} = t_p f_f = c_{mf} \alpha_f \quad (\text{G.24})$$

nas quais c_r e c_f são os coeficientes de rigidez lateral (*cornering stiffness*) achados por mínimos quadrados a partir do gráfico e t_p e a distância (trail) pneumática e mecânica, definida como distância perpendicular entre o eixo de direção e do ponto de contato entre a roda dianteira e o chão, o qual também é calculado através de mínimos quadrados sobre o gráfico.

G.2.3 Sistema Mecânico da Direção

A direção do Fiat Punto tem uma direção hidráulica assistida. Por conseguinte, o modelo tem três módulos: o atuador da direção, o sistema mecânico e o sistema hidráulico, como é

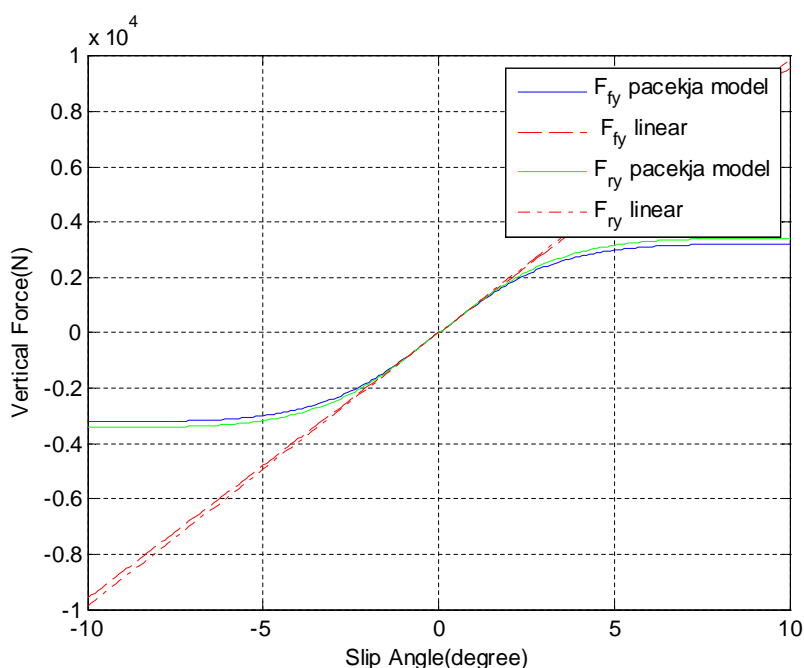


Figura G.5: Forças laterais das rodas dianteiras e traseiras

apresentado em (YIH, 2005), com algumas modificações no sistema mecânico. A relação entre os sistemas é descrito na figura G.7. As entradas do sistema são: o torque do condutor (τ_{user}), a razão de vasão da bomba hidráulica (Q_s), a qual será ignorada no modelo, a voltagem RMS do servo motor (v_m) e o torque de alinhamento da roda dianteira (τ_{fa}).

Atuador da Direção

O atuador tem um servomotor DC sem escovas e uma engrenagem planetária. O motor é descrito por um modelo simplificado do motor onde a dinâmica elétrica do motor é desconsiderada.

$$\tau_m = k_i I_m = K_i \left(\frac{V_m + k_E \omega_m}{R} \right) \quad (G.25)$$

na qual, I_m é a corrente do motor, R é a resistência de armadura, k_E é o ganho de voltagem, k_i é o ganho de corrente torque e ω_m é a velocidade angular do motor.

O trem de engrenagem é descrito pelo raio de redução (r_g) e a eficiência (η), portanto:

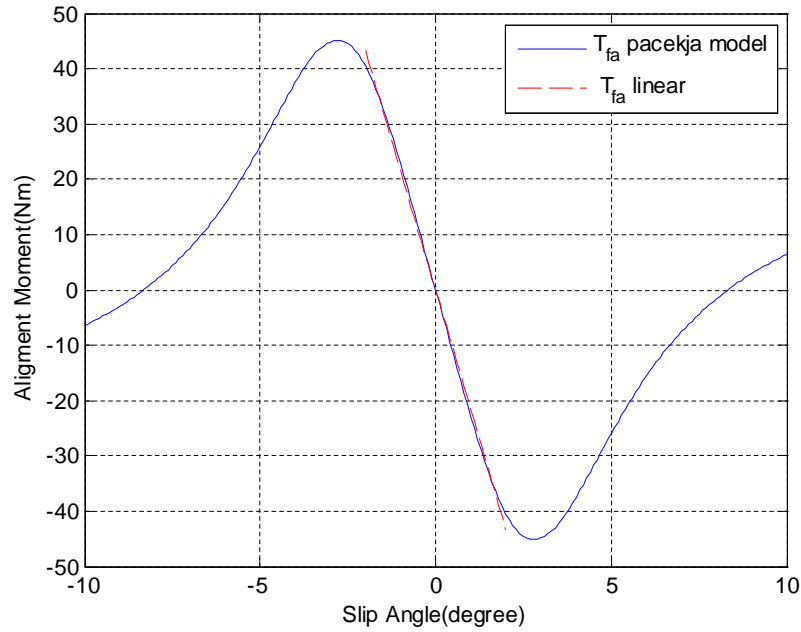


Figura G.6: Torque de alinhamento da roda dianteira equivalente do veículo.

$$\tau_j = \eta \cdot r_g \tau_m \quad \omega_m = r_g \dot{\delta}_j \quad (\text{G.26})$$

na qual, $\dot{\delta}_j$ é a velocidade angular da barra de união da direção. Os valores são obtidos a partir do fabricante o motor.

Sistema Mecânico da Direção

O desenho esquemático do sistema dinâmico é mostrado na figura G.8. Assumindo que a barra de direção e a barra de união são corpos rígidos, que a inércia do mecanismo de direção (válvula hidráulica, sistema cremalheira e o rack) são um só corpo rígido e desconsiderando a dinâmica das uniões universais, aplicando a segunda lei de Newton para a aceleração angular na barra da união e no mecanismo de direção temos (G.27) e (G.28)

$$J_1 \ddot{\delta}_j = -C_m \dot{\theta}_j - C_s \dot{\delta}_s - f_1(\tau_k, \delta_j) + \tau_m \eta r_g + \tau_{user} \quad (\text{G.27})$$

$$(J_r) \ddot{\delta}_f = -C_r \dot{\delta}_f + r_s \tau_k - \tau_{fa} + \tau_p \quad (\text{G.28})$$

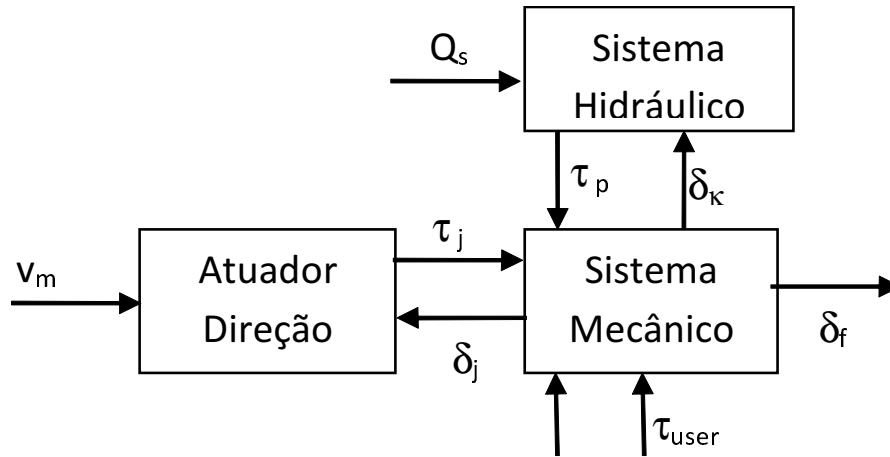


Figura G.7: Diagrama de bloco dos sistemas do modelo mecânico da direção.

na qual, $J_1 = J_m \cdot r_g^2 + J_j + J_s$; J_m, J_j, J_s, J_t, J_r são a inércia, C_m, C_s e C_r são os coeficientes de atrito viscoso no motor (m), a barra de união (j) a barra de direção (s) e barra equivalente do rack (r); r_s é a redução de velocidade causada pelo mecanismo de cremalheira; τ_p é o torque gerado pelo sistema hidráulico; τ_k é o momento gerado pela barra de torção do sistema hidráulico descrito pela equação:

$$T_k = k_t(\delta_k - r_s \delta_f) \quad (\text{G.29})$$

Finalmente como as barras de união, a barra de direção e a barra do rack estão juntas por uma união universal com ângulos β_1 e β_2 (figura G.9), o torque do usuário pode ser transformado a priori então $\delta_s = \delta_j$ e é necessário transformar δ_k e δ_j é o torque da barra torção.

$$\delta_k = \tan^{-1} \left(\frac{\tan \delta_j}{\cos \beta_2} \right) \quad (\text{G.30})$$

$$f_1(\tau_k, \delta_j) = \left(\frac{\cos \beta_2}{1 - \sin^2 \beta_2 \cos^2 \delta_j} \right) \tau_k \quad (\text{G.31})$$

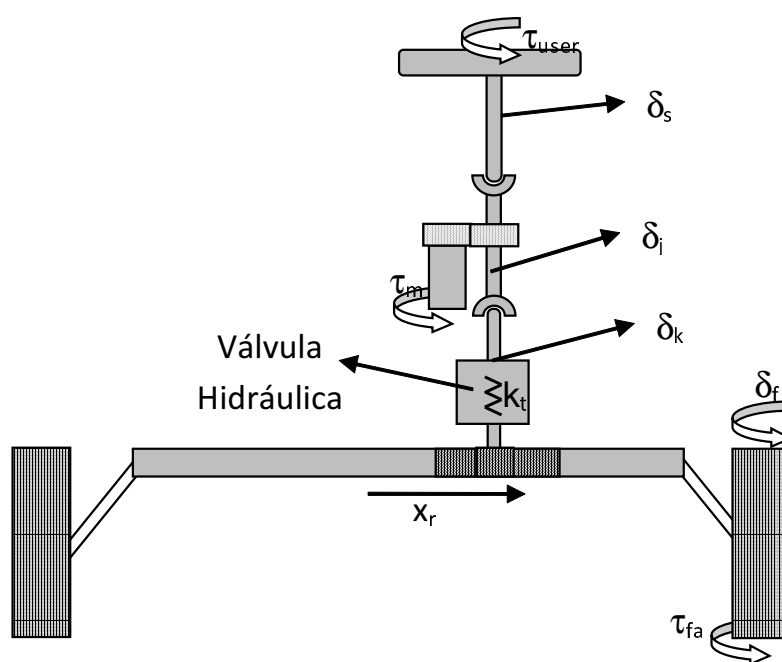


Figura G.8: Desenho do sistema mecânico da direção

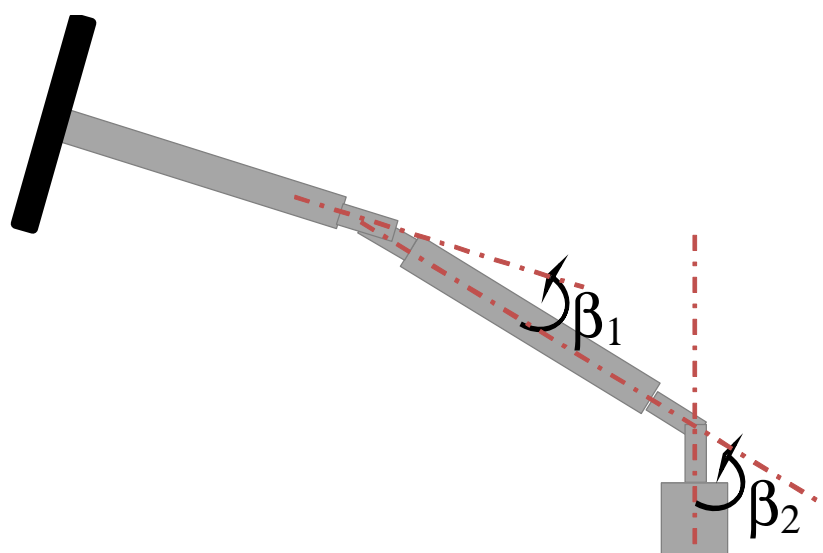


Figura G.9: Desenho dos ângulos das uniões universais no sistema de direção

Sistema Hidráulico

O modelo matemático do sistema hidráulico é descrito em detalhes em (PROCA E KEYHANI, 1998) (ROESTH, 2007) e pode ser representado pela figura G.10. Os principais componentes do sistema de energia hidráulica auxiliar são a bomba hidráulica, a válvula de carretel rotativo, e o pistão. A válvula de carretel rotativo consiste na barra de torção, tambor interior, e manga exterior (YIH, 2005).

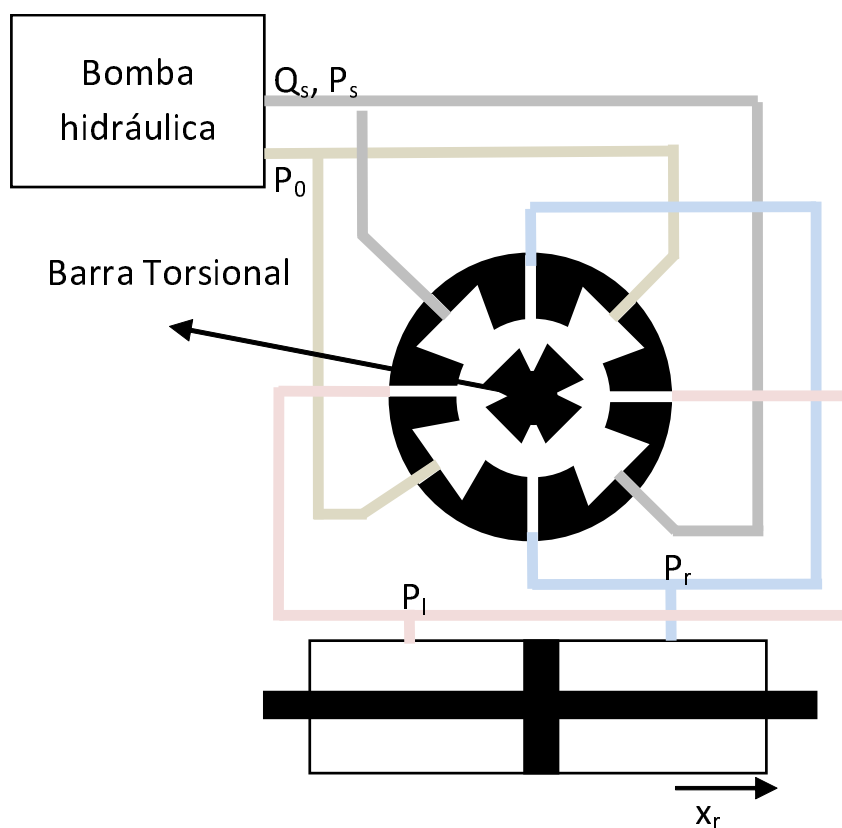


Figure G.10: Desenho esquemático do sistema hidráulico da direção

O diferencial de pressão no interior do cilindro é proporcional em forma não linear para o ângulo de barra de torção. Embora, o torque de direção é a força sobre o pistão que aqui serão aproximados linearmente por $\tau_p = r_h \tau_k$ omitindo a dinâmica do sistema hidráulico.

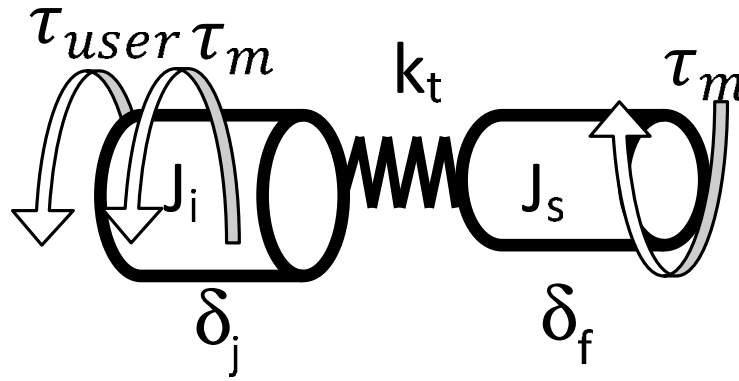


Figura G.11: Desenho do Modelo Dinâmico Simplificado da direção

Representação de Estados do sistema de direção

Tendo como variáveis de estado $\mathbf{x}_s = \begin{bmatrix} \delta_j & \dot{\delta}_j & \delta_f & \dot{\delta}_f \end{bmatrix}^T$ e entradas v_m , τ_{user} e τ_{fa} temos a representação de estados (G.32)

$$\dot{\mathbf{x}}_s = \begin{bmatrix} \dot{\theta}_j \\ \ddot{\theta}_j \\ \dot{\delta}_f \\ \ddot{\delta}_f \end{bmatrix} = \begin{bmatrix} \frac{1}{J_1} \left(- (C_m + C_{seq}) \dot{\delta}_j - \left(\frac{\cos \beta_2}{1 - \sin^2 \beta_2 \cos^2 \delta_j} \right) k_t \left(\tan^{-1} \left(\frac{\tan \delta_j}{\cos \beta_2} \right) - r_s \delta_f \right) \dots \right. \\ \left. + \eta r_g \left(k_i \frac{v_m + k_E r_g \delta_j}{R} \right) + \tau_{user} \right) \\ \dot{\delta}_f \\ \frac{1}{J_t + J_r} \left(-C_r \dot{\delta}_f + r_s (1 + r_h) k_t \left(\tan^{-1} \left(\frac{\tan \delta_j}{\cos \beta_2} \right) - r_s \delta_f \right) - T_{fa} \right) \end{bmatrix} \quad (G.32)$$

Se o efeito das uniões universais não é levado em conta, o sistema poderia ser aproximado pelo esquema da figura G.11 e descrito pelas representação de estados linear (G.33)

$$\dot{\mathbf{x}}_s = \begin{bmatrix} \dot{\theta}_j \\ \ddot{\theta}_j \\ \dot{\delta}_f \\ \ddot{\delta}_f \end{bmatrix} \simeq \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_t}{J_j} & -\frac{c_j + k_e r_g^2 k_i \eta}{J_j} & \frac{k_t r_s}{J_j} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_t}{J_s} r_h r_s & 0 & -\frac{k_t}{J_s} r_h r_s^2 & -\frac{c_s}{J_w} \end{bmatrix} \mathbf{x}_s + \begin{bmatrix} 0 & 0 & 0 \\ \frac{r_g k_i \eta}{J_s R} & \frac{1}{J_s} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{-1}{J_w} \end{bmatrix} \begin{bmatrix} V_m \\ \tau_{user} \\ T_{fa} \end{bmatrix} \quad (G.33)$$

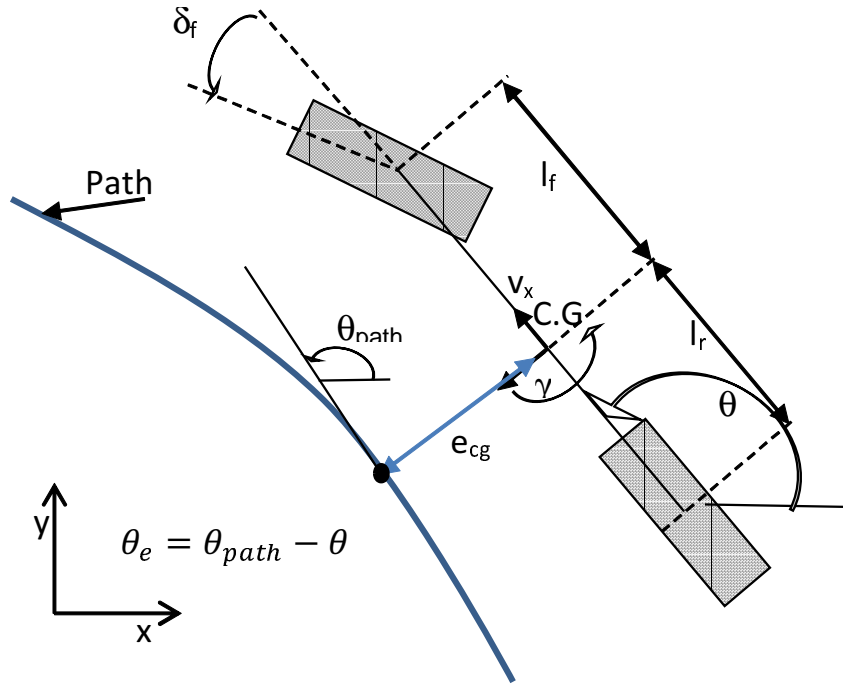


Figura G.12: Modelo de bicicleta no sistema de coordenadas da trajetória

G.3 Modelo dinâmico do carro referenciado ao modelo cinemático

O modelo da dinâmica planar do veículo pode ser referenciado à trajetória desejada como é apresentada na figura G.12. Este modelo permite definir os erro lateral, e_{cg} , e o erro de orientação, θ_e como variáveis de estado. Assumindo pequenos ângulos de deslizamento, α_f e α_r e pequenos ângulos de direção δ_f , pode-se usar a aproximação linear das forças dos pneus: equações (G.22), (G.23) e (G.24); assim como as equações dos ângulos (G.20), (G.21), portanto a representação em variáveis de estado \mathbf{x}_c é aproximada ao modelo da equação (G.34)

$$\dot{\mathbf{x}}_c = \begin{bmatrix} \dot{v}_y \\ \dot{\gamma} \\ \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \approx \begin{bmatrix} \frac{-(c_f+c_r)}{m_v v_x} v_y + \left(\frac{-l_f c_f + l_r c_r}{m_v v_x} - v_x \right) \gamma + \frac{c_f}{m_v} \delta_f \\ \frac{(-l_f c_f + l_r c_r)}{J_v v_x} v_y - \frac{(l_f^2 c_f + l_r^2 c_r)}{J_v v_x} \gamma + \frac{l_f c_f}{J_v} \delta_f \\ v_x \cos \theta - v_y \sin \theta \\ v_x \sin \theta + v_y \cos \theta \\ \gamma \end{bmatrix} \quad (\text{G.34})$$

Linearizar o modelo da equação (G.34) requereria restringir o modelo a pequenos ângulos de orientação do veículo ou gerar uma família de modelos lineares de acordo ao ângulo de

orientação, portanto as variáveis de estado serão transformadas aos erros lateral e de orientação com respeito ao caminho desejado. De acordo a (SNIDER, 2009), assumindo a velocidade longitudinal constante, a velocidade de orientação da trajetória $r_p(s)$ é definida em termos da curvatura como:

$$r_p(s_p) = \kappa(s_p)v_x = \frac{d\theta}{ds}v_x \quad (\text{G.35})$$

Portanto a aceleração lateral da trajetória $\dot{v}_y(s_p)$ (por exemplo se a trajetória é um círculo de raio r , $\kappa = r^{-1}$ e $\dot{v}_y = v_x^2 r^{-1}$ é a aceleração centrífuga) é definida como:

$$\dot{v}_y(s_p) = \kappa(s_p)v_x^2 = r_p(s_p)v_x$$

Definindo e_{cg} como o erro ortogonal entre a trajetória e o C.G do veículo

$$\begin{aligned} \ddot{e}_{cg} &= (\dot{v}_y + v_x\gamma) - \dot{v}_y(s_p) = \dot{v}_y + v_x(\gamma(s_p) - r_p(s_p)) \\ &= \dot{v}_y + v_x\dot{\theta}_e \end{aligned} \quad (\text{G.36})$$

e

$$\dot{e}_{cg} = v_y + v_x \sin(\theta_e) \quad (\text{G.37})$$

com o erro de orientação $\theta_e = \theta - \theta_p(s_p)$. Substituindo v_y e γ na (G.17) pelas variáveis de estado $\mathbf{x}_e = [e_{cg} \ \dot{e}_{cg} \ \theta_e \ \dot{\theta}_e]^T$ usando (G.36), (G.37) e assumindo que θ_e é pequeno tem-se a representação de estados (G.38)

$$\begin{aligned} \dot{\mathbf{x}}_e &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{(c_r+c_f)}{mv_x} & \frac{(c_r+c_f)}{mv_x} & \frac{l_r c_r - l_f c_f}{mv_x} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{l_r c_r - l_f c_f}{J_v v_x} & -\frac{l_r c_r - l_f c_f}{J_v} & -\frac{l_r^2 c_r + l_f^2 c_f}{J_v v_x} \end{bmatrix} \mathbf{x}_e \\ &+ \begin{bmatrix} 0 \\ \frac{c_f}{m} \\ 0 \\ \frac{l_f c_f}{m} \end{bmatrix} \delta_f + \begin{bmatrix} 0 \\ \frac{l_r c_r - l_f c_f}{mv_x} - v_x \\ 0 \\ -\frac{l_r^2 c_r + l_f^2 c_f}{J_v v_x} \end{bmatrix} r_p(s) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1 \end{bmatrix} \dot{r}_p(s) \end{aligned} \quad (\text{G.38})$$