On control of parallel robots for high dynamic performances: from design to experiments

Hussein Saied

To cite this version:


HAL Id: tel-02934747
https://tel.archives-ouvertes.fr/tel-02934747v2
Submitted on 14 Sep 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
On Control of Parallel Robots for High Dynamic Performances: From Design to Experiments

Présentée par Hussein SAIED
Le 29 Novembre 2019

Sous la direction de François PIERROT, Clovis FRANCIS et Ahmed CHEMORI, Maher EL RAFeI

Devant le jury composé de

Edouard LAROCHE, Professeur, Université de Strasbourg, lCube
Mohamed DJEMAI, Professeur, Université Polytechnique Hauts-de-France, LAMIH
Olivier COMPANY, Maître de Conférences, Université de Montpellier, LIRMM
Hassan SHRAIM, Professeur, Université Libanaise, CRSI
François PIERROT, Directeur de recherche, CNRS, LIRMM
Clovis FRANCIS, Professeur, Université Libanaise, CRSI
Ahmed CHEMORI, Chargé de recherche, CNRS, LIRMM
Maher EL RAFeI, Maître de conférences, Université Libanaise, CRSI

Rapporteur
Rapporteur (President)
Examinateur
Examinateur
Directeur de thèse
Directeur de thèse
Co-encadrant de thèse
Co-encadrant de thèse
## Contents

List of Figures ................................. 1  
List of Tables ................................. 7  
Acknowledgements ........................... 9  
General introduction ....................... 11  

1 Context, problem formulation and state of the art .......................... 17  
   1.1 Introduction .............................................. 18  
   1.2 Robotic manipulators ................................. 19  
   1.3 Parallel versus serial manipulators ...................... 20  
      1.3.1 Singularities ........................................... 21  
      1.3.2 Workspace ............................................. 23  
      1.3.3 Payload/weight ratio ................................... 23  
      1.3.4 Stiffness and dynamic performance ................. 23  
      1.3.5 Accuracy ................................................ 24  
   1.4 A historical overview of parallel robots .................. 24
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5</td>
<td>Potential applications of parallel manipulators</td>
<td>29</td>
</tr>
<tr>
<td>1.5.1</td>
<td>Industrial applications</td>
<td>29</td>
</tr>
<tr>
<td>1.5.2</td>
<td>Motion Simulators</td>
<td>31</td>
</tr>
<tr>
<td>1.5.3</td>
<td>Machining applications</td>
<td>32</td>
</tr>
<tr>
<td>1.5.4</td>
<td>Space applications</td>
<td>33</td>
</tr>
<tr>
<td>1.5.5</td>
<td>Synchrotrons</td>
<td>34</td>
</tr>
<tr>
<td>1.5.6</td>
<td>Medical applications</td>
<td>34</td>
</tr>
<tr>
<td>1.5.7</td>
<td>Agricultural applications</td>
<td>35</td>
</tr>
<tr>
<td>1.6</td>
<td>Control problem formulation</td>
<td>36</td>
</tr>
<tr>
<td>1.6.1</td>
<td>Control challenges of parallel manipulators</td>
<td>37</td>
</tr>
<tr>
<td>1.6.1.1</td>
<td>Nonlinear complex dynamics of PKMs</td>
<td>37</td>
</tr>
<tr>
<td>1.6.1.2</td>
<td>Uncertainties in PKMs: structured and unstructured</td>
<td>39</td>
</tr>
<tr>
<td>1.6.1.3</td>
<td>Actuation Redundancy in PKMs</td>
<td>40</td>
</tr>
<tr>
<td>1.7</td>
<td>Dynamic modeling of parallel manipulators</td>
<td>41</td>
</tr>
<tr>
<td>1.7.1</td>
<td>The standard inverse dynamic model</td>
<td>42</td>
</tr>
<tr>
<td>1.7.2</td>
<td>Properties of the dynamic model</td>
<td>43</td>
</tr>
<tr>
<td>1.7.2.1</td>
<td>Property of mass and inertia matrix</td>
<td>44</td>
</tr>
<tr>
<td>1.7.2.2</td>
<td>Property of Coriolis and centrifugal matrix</td>
<td>44</td>
</tr>
<tr>
<td>1.7.2.3</td>
<td>Property of gravity vector</td>
<td>45</td>
</tr>
<tr>
<td>1.7.2.4</td>
<td>Linear formulation property of the dynamics</td>
<td>45</td>
</tr>
<tr>
<td>1.8</td>
<td>Overview of motion control solutions for parallel manipulators</td>
<td>46</td>
</tr>
<tr>
<td>1.8.1</td>
<td>Joint space versus Cartesian space control</td>
<td>48</td>
</tr>
<tr>
<td>1.8.2</td>
<td>Kinematic control strategies</td>
<td>50</td>
</tr>
<tr>
<td>1.8.2.1</td>
<td>PD/PID control</td>
<td>50</td>
</tr>
<tr>
<td>1.8.2.2</td>
<td>Nonlinear PD/PID control</td>
<td>51</td>
</tr>
<tr>
<td>1.8.2.3</td>
<td>Decentralized sliding mode control</td>
<td>52</td>
</tr>
<tr>
<td>1.8.2.4</td>
<td>$L_1$ adaptive control</td>
<td>54</td>
</tr>
<tr>
<td>1.8.3</td>
<td>Dynamic control strategies</td>
<td>55</td>
</tr>
<tr>
<td>1.8.3.1</td>
<td>PD control with gravity compensation</td>
<td>55</td>
</tr>
<tr>
<td>1.8.3.2</td>
<td>Augmented PD control</td>
<td>56</td>
</tr>
<tr>
<td>1.8.3.3</td>
<td>Computed torque control</td>
<td>57</td>
</tr>
</tbody>
</table>
1.8.3.4 PD control with computed feedforward .......................... 58
1.8.3.5 Dynamic adaptive control ...................................... 59
1.8.3.6 Control with time-varying feedback gains ....................... 60
1.8.3.7 Sliding mode control ............................................. 61
1.8.3.8 Other dynamic control approaches ............................. 63

1.9 How can we improve the performance of PKMs from a control point of view? 64
1.10 Objectives of the thesis .............................................. 65
1.11 Main contributions of the thesis ..................................... 65
1.12 Conclusion .............................................................. 67

2 Description and modeling of PKM prototypes .......................... 69
2.1 Introduction .............................................................. 69
2.2 Non-redundant parallel robot prototypes ............................. 70
  2.2.1 Delta robot: a 3-DOF non-redundant PKM ....................... 70
     2.2.1.1 Description of Delta PKM .................................. 70
     2.2.1.2 Kinematics of Delta PKM ................................. 71
     2.2.1.3 Differential Kinematics of Delta PKM ................. 75
     2.2.1.4 Dynamics of Delta PKM .................................. 77
  2.2.2 VELOCE robot: a 4-DOF non-redundant PKM .................... 80
     2.2.2.1 Description of VELOCE PKM .............................. 80
     2.2.2.2 Kinematics of VELOCE PKM .............................. 81
     2.2.2.3 Differential Kinematics of VELOCE PKM .............. 84
     2.2.2.4 Dynamics of VELOCE PKM ............................... 85
  2.3 Redundant parallel robot prototype ................................. 87
  2.3.1 SPIDER4 robot: a 5-DOF redundant PKM ....................... 88
     2.3.1.1 Description of SPIDER4 PKM ............................ 88
     2.3.1.2 Kinematics of SPIDER4 PKM ............................ 88
     2.3.1.3 Differential Kinematics of SPIDER4 PKM .......... 92
     2.3.1.4 Dynamics of SPIDER4 PKM ............................. 94
  2.4 Conclusion .............................................................. 96

3 Proposed control solutions ............................................. 99
3.5.3 Elimination of antagonistic internal forces ................. 141
3.6 Conclusion .................................................................. 142

4 Real-time experiments and results .................................. 145
4.1 Introduction ............................................................... 145
4.2 Experimental platforms and implementation issues .............. 146
  4.2.1 Experimental testbed of the 3-DOF Delta robot .............. 146
    4.2.1.1 Reference trajectory generation ....................... 147
  4.2.2 Experimental testbed of the 5-DOF SPIDER4 robot .......... 148
    4.2.2.1 Reference trajectory generation ....................... 150
  4.2.3 Experimental testbed of the 4-DOF VELOCE robot .......... 153
    4.2.3.1 Reference trajectory generation ....................... 155
  4.2.4 Performance evaluation criteria ............................... 157
  4.2.5 Tuning of the control gains .................................... 158
4.3 Experimental results of contribution 1: Time-varying feedback RISE control .................................................. 159
  4.3.1 Tuning procedure of the control gains ....................... 159
    4.3.1.1 Tuning of the standard RISE control gains .......... 159
    4.3.1.2 Tuning of the proposed time-varying feedback control gains ................................. 159
  4.3.2 Scenario 1: nominal case ........................................ 160
  4.3.3 Scenario 2: robustness towards payload and speed changes .............................................................. 163
  4.3.4 Performance index versus operating acceleration ........... 167
4.4 Experimental results of contribution 2: Model-based ST-SMC algorithm ...................................................... 168
  4.4.1 Experimental results of contribution 2 on SPIDER4 robot .............................................................. 168
    4.4.1.1 Feedforward PID versus conventional CT-ST-SMC .... 169
    4.4.1.2 Feedforward PID versus proposed feedforward ST-SMC .............................................. 172
  4.4.2 Experimental results of contribution 2 on Delta robot .............................................................. 177
    4.4.2.1 Scenario 1: nominal case .................................. 178
    4.4.2.2 Scenario 2: robustness towards payload and speed changes ........................................... 180
    4.4.2.3 Scenario 3: robustness towards very high accelerations ........................................... 183
4.5 Experimental results of contribution 3: PDFF control with actuator and friction dynamics ............................... 186
  4.5.1 Friction parameters identification results ...................... 187
CONTENTS

4.5.2 Scenario 1: nominal case ........................................... 187
4.5.3 Scenario 2: robustness towards payload and speed changes .... 190
4.6 Conclusion ................................................................. 194

General conclusion ......................................................... 197

Bibliography ................................................................. 201

Appendices ................................................................. 227

A Bounded residual dynamics of robotic manipulators .............. 227

B Property of the vectorial hyperbolic tangent ...................... 231
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Illustrative diagram of the 4-DOF SCARA robotic manipulator.</td>
<td>20</td>
</tr>
<tr>
<td>1.2</td>
<td>Robotic manipulator structures: (a) serial, (b) parallel, and (c) hybrid.</td>
<td>21</td>
</tr>
<tr>
<td>1.3</td>
<td>The spherical parallel mechanism proposed in 1928 by James E. Gwinnett</td>
<td>25</td>
</tr>
<tr>
<td>1.4</td>
<td>The spray painting 5-DOF parallel robot proposed in 1942 by Willard L.V. Pollard</td>
<td>26</td>
</tr>
<tr>
<td>1.5</td>
<td>The original Gough platform in 1954 [Bonev, 2003].</td>
<td>26</td>
</tr>
<tr>
<td>1.7</td>
<td>Schematic view of Stewart platform proposed as a flight simulator [Stewart, 1965].</td>
<td>27</td>
</tr>
<tr>
<td>1.8</td>
<td>The first flight simulator based on a hexapod structure as in the mid of 1960s. [Bonev, 2003].</td>
<td>28</td>
</tr>
<tr>
<td>1.9</td>
<td>Technical drawing of the original Delta robot [Clavel, 1990].</td>
<td>29</td>
</tr>
<tr>
<td>1.10</td>
<td>IRB 360 FlexPicker (ABB technology)</td>
<td>30</td>
</tr>
<tr>
<td>1.11</td>
<td>The high-speed Adept Quattro robot (Adept technology)</td>
<td>30</td>
</tr>
<tr>
<td>1.12</td>
<td>Tapster 2 robot designed by Tapster Robotics.</td>
<td>31</td>
</tr>
<tr>
<td>1.13</td>
<td>Full flight simulator of a Boeing 737.</td>
<td>31</td>
</tr>
<tr>
<td>1.14</td>
<td>MISTRAL hexapod swell simulator at HOPPE Marine.</td>
<td>32</td>
</tr>
</tbody>
</table>
1.15 MISTRAL hexapod with a ship model at the wave basin of Ifremer Boulogne sur Mer, France. ................................................................. 32
1.16 Toyoda HexaM PKM-based machine tool [Toyama et al., 1998]. ................. 33
1.17 CAD view of ARROW robot prototype. ................................................... 33
1.18 Optical micro-positioning system used by TAS during satellite manufacturing. . 34
1.19 Photograph of the instrument installed at Advanced Photon Source station 12ID-D [Ju et al., 2017]. ................................................................. 35
1.20 High vacuum diffractometer at SIRIUS beamline. ..................................... 35
1.21 Hexapod medical robot of high precision positioning. ................................. 35
1.22 The autonomous robot weeder of ecoRobotix depositing seeds in a targeted way. 36
1.23 Overview of general research topics of parallel manipulators. ..................... 38
1.24 Overview of the considerable control challenges of PKMs. ......................... 39
1.25 Illustrative diagram of actuation redundancy. (a) Non-redundant 3-DOF PKM, (b) Redundantly actuated 3-DOF PKM. ................................................ 40
1.26 Definition of the loop constraints for a planar 2-DOF redundantly actuated PKM [Mueller, 2011]. ................................................................. 42
1.27 Classification of the main proposed control schemes for parallel manipulators. 47
1.28 Schematic illustration of the joint space control strategy. ............................ 48
1.29 Schematic illustration of the Cartesian space control strategy. ....................... 48
1.30 Illustration of the nonlinear function. ........................................................ 52
1.31 Phase plane response of the system with SMC [Salim Qureshi et al., 2018]. ...... 53
1.32 Block diagram of the $L_1$ adaptive controller [Bennehar et al., 2015a]. ........ 55
1.33 Block diagram of the computed torque controller in joint space. .................. 57
1.34 Block diagram of the general adaptive feedforward controller in joint space. .... 59

2.1 A schematic view of Delta parallel robot including ①: Fixed-base, ②: Actuator, ③: Rear-arm, ④: Forearm, ⑤: Traveling-plate, ⑥: End-effector ....................... 70
2.2 Distribution of the geometric points and frames on Delta PKM ...................... 72
2.3 Top view of the geometrics of Delta PKM .................................................. 72
2.4 Side view of the geometrics of Delta PKM .................................................. 73
2.5 Illustration of dynamic parameters of Delta parallel robot arms. .................... 77
## List of Figures

2.6 A schematic view of VELOCE parallel robot including 1: Fixed-base, 2: Actuator, 3: Rear-arm, 4: Forearm, 5: Moving platform, 6: End-effector. 81

2.7 Illustrative view of the geometric points and frames considered on VELOCE PKM. 83

2.8 A schematic view of SPIDER4 parallel robot including 1: Fixed-base, 2: Actuator, 3: Rear-arm, 4: Forearm, 5: Nacelle, 6: Serial wrist mechanism, 7: Spindle. 89

2.9 Overall dimensions (in mm) of the whole system, SPIDER4 and the tooling. 89

2.10 View of the geometrics of SPIDER4 PKM from the side of the fixed-base. 91

2.11 View of the geometric parameters of one kinematic chain of SPIDER4 PKM. 91

2.12 Illustrative view of the gravitational force acting on a rear-arm of SPIDER4 PKM. 95

3.1 Plot of the evolution of the proportional gain $K_s(.)$ with respect to its argument $e_2$. 108

3.2 Plot of the evolution of the integral gain $\alpha_2(.)$ with respect to its argument $\int e_2$. 108

3.3 Block diagram of the proposed TV-RISE control scheme applied to parallel manipulators in joint space, $K_s(.)$ and $\alpha_2(.)$ represent the time-varying feedback gains. 114

3.4 Illustrative diagram of the effect of the mechanical actuator dynamics. 131

3.5 Illustrative diagram of Coulomb friction force in terms of the sliding velocity. 132

3.6 Illustrative diagram of viscous friction force in terms of the sliding velocity. 133

3.7 Block diagram of the Least Squares Estimation method applied to identify the friction parameters. 139

4.1 View of the real Delta parallel robot used for real-time experiments. 147

4.2 Point-to-point S-curve profile motion illustration. 148

4.3 3D view of pick-and-place reference trajectory in Cartesian space. 149

4.4 View of the real SPIDER4 parallel robot used for real-time experiments. 150

4.5 Experimental setup of SPIDER4 robot. 151

4.6 G1 linear interpolation motion profile. 152

4.7 G3 circular interpolation motion profile. 153

4.8 3D view of Trajectory I for SPIDER4 robot in Cartesian space. 154

4.9 3D view of Trajectory II for SPIDER4 robot in Cartesian space. 154
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.10</td>
<td>View of the real VELOCE parallel robot used for real-time experiments.</td>
<td>155</td>
</tr>
<tr>
<td>4.11</td>
<td>Experimental setup of VELOCE robot.</td>
<td>155</td>
</tr>
<tr>
<td>4.12</td>
<td>Point-to-point fifth degree polynomial profile motion illustration.</td>
<td>156</td>
</tr>
<tr>
<td>4.13</td>
<td>3D view of the reference trajectory for VELOCE robot in Cartesian space.</td>
<td>157</td>
</tr>
<tr>
<td>4.14</td>
<td>Scenario 1: Evolution of the Cartesian and joint tracking errors of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>161</td>
</tr>
<tr>
<td>4.15</td>
<td>Scenario 1: Evolution of the control input torques of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>162</td>
</tr>
<tr>
<td>4.16</td>
<td>Scenario 1: Evolution of the nonlinear feedback gain ((K_s(.) + 1)) of the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>163</td>
</tr>
<tr>
<td>4.17</td>
<td>Scenario 1: Evolution of the nonlinear feedback gain (((k_s + 1)\alpha_2(\cdot))) of the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>163</td>
</tr>
<tr>
<td>4.18</td>
<td>Scenario 2: Evolution of the Cartesian and joint tracking errors of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>164</td>
</tr>
<tr>
<td>4.19</td>
<td>Scenario 2: Evolution of the control input torques of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>165</td>
</tr>
<tr>
<td>4.20</td>
<td>Scenario 2: Evolution of the nonlinear feedback gain ((K_s(.) + 1)) of the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>166</td>
</tr>
<tr>
<td>4.21</td>
<td>Scenario 2: Evolution of the nonlinear feedback gain ((K_s(.) + 1)) of the proposed time-varying feedback RISE controllers on Delta robot.</td>
<td>166</td>
</tr>
<tr>
<td>4.22</td>
<td>Clustered column chart of RMSE(_C) of the original RISE and the proposed time-varying feedback RISE controllers versus operating acceleration on Delta robot.</td>
<td>168</td>
</tr>
<tr>
<td>4.23</td>
<td>Scenario 1: Evolution of the control input torques of the conventional CT-ST-SMC on SPIDER4 robot.</td>
<td>170</td>
</tr>
<tr>
<td>4.24</td>
<td>Scenario 2: Evolution of the Cartesian tracking errors of the standard PIDFF and the conventional CT-ST-SMC controllers on SPIDER4 robot.</td>
<td>171</td>
</tr>
<tr>
<td>4.25</td>
<td>Scenario 2: Evolution of the control input torques of the standard PIDFF and the conventional CT-ST-SMC controllers on SPIDER4 robot.</td>
<td>172</td>
</tr>
<tr>
<td>4.26</td>
<td>Scenario 1: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot.</td>
<td>173</td>
</tr>
</tbody>
</table>
List of Figures

4.27 Scenario 1: Evolution of the control input torques of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot. 174
4.28 Scenario 2: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot. 176
4.29 Scenario 2: Evolution of the control input torques of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot. 177
4.30 Scenario 1: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot. 179
4.31 Scenario 1: Evolution of the control input torques of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot. 180
4.32 Scenario 2: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot. 181
4.33 Scenario 2: Evolution of the control input torques of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot. 182
4.34 Scenario 3: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot. 183
4.35 Scenario 3: Evolution of the Cartesian tracking trajectories with the standard PIDFF control on Delta robot. 184
4.36 Scenario 3: Evolution of the Cartesian tracking trajectories with the proposed FF-ST-SMC algorithm on Delta robot. 185
4.37 Scenario 3: Evolution of the control input torques of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot. 186
4.38 Validation of the friction parameters identification at low and high operating speeds on VELOCE robot. 188
4.39 Scenario 1: Evolution of the Cartesian tracking errors of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot. 189
4.40 Scenario 1: Evolution of the Cartesian tracking errors of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot within the interval [4.5, 5.5] sec. 190
4.41 Scenario 1: Evolution of the control input torques of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot. 191
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.42</td>
<td>Scenario 2: Evolution of the Cartesian tracking errors of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot.</td>
<td>192</td>
</tr>
<tr>
<td>4.43</td>
<td>Scenario 2: Evolution of the Cartesian tracking errors of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot within the interval [1.5, 2] sec.</td>
<td>193</td>
</tr>
<tr>
<td>4.44</td>
<td>Scenario 2: Evolution of the control input torques of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot.</td>
<td>194</td>
</tr>
<tr>
<td>A.1</td>
<td>Illustrative diagram of the scalar function $S(e)$.</td>
<td>230</td>
</tr>
</tbody>
</table>
List of Tables

1.1 The main dynamic properties of serial and parallel manipulators. .................................. 24
2.1 The main dynamic parameters of Delta parallel robot. ................................................. 80
2.2 The main dynamic parameters of VELOCE parallel robot. ............................................. 87
2.3 The main dynamic parameters of SPIDER4 parallel robot. ............................................. 96
4.1 The control gains of the original RISE and the proposed time-varying feedback RISE controllers. .......................................................... 160
4.2 Scenario 1: Control performance evaluation of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot. ..................................................... 162
4.3 Scenario 2: Control performance evaluation of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot. ..................................................... 167
4.4 The control gains of the conventional CT-ST-SMC, the proposed FF-ST-SMC, and the standard PIDFF controllers used on SPIDER4 robot. ................................................. 169
4.5 Scenario 2: Control performance evaluation of the standard PIDFF and the conventional CT-ST-SMC controllers on SPIDER4 robot. ................................................. 171
4.6 Scenario 1: Control performance evaluation of both controllers of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot. ......................................... 175
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.7</td>
<td>Scenario 2: Control performance evaluation of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot.</td>
</tr>
<tr>
<td>4.8</td>
<td>The control gains of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.</td>
</tr>
<tr>
<td>4.9</td>
<td>Scenario 1: Control performance evaluation of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.</td>
</tr>
<tr>
<td>4.10</td>
<td>Scenario 2: Control performance evaluation of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.</td>
</tr>
<tr>
<td>4.11</td>
<td>Scenario 3: Control performance evaluation of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.</td>
</tr>
<tr>
<td>4.12</td>
<td>The identified friction parameters of VELOCE robot.</td>
</tr>
<tr>
<td>4.13</td>
<td>Scenario 1: Control performance evaluation of the Ex-PDFF I.</td>
</tr>
<tr>
<td>4.14</td>
<td>Scenario 1: Control performance evaluation of the Ex-PDFF II.</td>
</tr>
<tr>
<td>4.15</td>
<td>Scenario 2: Control performance evaluation of the standard PDFF and the Ex-PDFF I controllers on VELOCE robot.</td>
</tr>
<tr>
<td>4.16</td>
<td>Scenario 2: Control performance evaluation of the standard PDFF and the Ex-PDFF II controllers on VELOCE robot.</td>
</tr>
</tbody>
</table>
Acknowledgements

Firstly, I would like to express my sincere gratitude to my supervisors, Ahmed CHEMORI and Maher EL RAIFEI, for their continuous support of my PhD study during the past three years, for their patience, motivation, and immense knowledge.

I would also like to thank my committee members, Edouard LAROCHE, Mohamed DJEMAÏ, Olivier COMPANY, and Hassan SHRAIM, for their presence and for their insightful comments and encouragement.

I would also like to thank Francois PIERROT for accepting to be my thesis director, for giving me the opportunity to join LIRMM and accomplish this thesis. As well as Clovis FRANCIS, my thesis co-director, for giving me the opportunity to establish my thesis as a joint PhD with the Lebanese University.

I would especially like to thank Mohamed Bouri, EPFL, Switzerland, for the wonderful collaborations we have made during this thesis. Thank you for your support, encouragement, and advices, for the nights we were working together, and for all the enjoyable moments we have spent together in Lausanne.

I would also like to thank all my colleagues at LIRMM and friends. Thank you for your support, and for all the fun we have had in the last three years.
ACKNOWLEDGEMENTS

My deep and sincere gratitude to my family for their continuous and unparalleled love, help and support. I am grateful to my brothers and sister for always being there for me as friends. I am forever indebted to my parents for giving me the opportunities and experiences that have made me who I am. They selflessly encouraged me to explore new directions in life and seek my own destiny. This journey would not have been possible if not for them, and I dedicate this milestone to them.

I would also like to thank the girl who was always with me. Thank you for seeing the best in me and never giving up. Thank you for waiting patiently. Thank you for motivating me to be a better person. Thank you for showing me what it truly means to love someone. Thank you for being my best friend. Thank you for being you.

Finally, I would like to thank God for giving me the strength, knowledge, ability and opportunity to undertake this research study and to persevere and complete it satisfactorily. Without his blessings, this achievement would not have been possible.
Karol Olszak, Dariusz Korycki, Bogusław Waryś


general introduction

Automation has gained a wide interest in the last centuries in various fields such as industries, medicine, household life, agriculture, space, etc. It is all about replacing the workspace of human beings by computerized machines that can perceive the environment, take the optimal decision, and execute the desired process. Robotic manipulators have been one of the major automated machines used extensively in several areas. Two main types of robotic manipulators exist nowadays: serial manipulators and parallel manipulators.

Although serial manipulators have been mostly used in the last centuries, the interest about parallel manipulators has increased recently thanks to their special features. In contrast to serial manipulators, parallel ones offer more stiffness, better accuracy, high-speed capabilities, and a higher payload-to-weight ratio. However, it still suffers from some drawbacks such as limited workspace and complex singularities behavior. In fact, parallel manipulators are not replacing serial ones, but they offer various advantages for certain applications that need high accelerations and high accuracy.

Parallel manipulators are known by their high nonlinearities, coupled actuation, uncertainties, and actuation redundancy. All the aforementioned aspects can be considered as sources of errors (if they are not taken into account) that may deteriorate the performance of parallel manipulators. In this context, advanced control schemes capable of compen-
sating for the errors become an important requirement for parallel manipulators. Control design shall guarantee robustness and good performances with the change of the operating conditions.

**Problem formulation**

The control design of parallel manipulators is a key factor in obtaining high dynamic performances. However, control of parallel manipulators is considered a challenging task due to several reasons can be mentioned as follows:

- **Complexity of the dynamics:**
  Parallel manipulators are known with their high nonlinearities which may increase considerably when operating at high accelerations. Moreover, their closed-loop kinematic structure gives rise to coupled dynamics that need careful control synchronization between the actuators.

- **Structured and unstructured uncertainties:**
  Uncertainties are the differences or errors between the formulated dynamic model and the real parallel manipulator. Unstructured uncertainties can emerge from the wear of the parts, geometric errors, modeling simplifications, disturbances, etc. While structured uncertainties appear in the form of inaccurate knowledge about the dynamic parameters or their variation with time (payload, external contact force, etc.)

- **Actuation redundancy:**
  Actuation redundancy is achieved by adding additional actuated kinematic chains to the structure such that the number of actuators become greater than that of the degrees-of-freedom. This may increase the achievable accelerations of the system and enlarge the workspace by eliminating singularities. However, it can generate important internal forces that may even cause damages to the mechanical structure of the parallel manipulator.
Objectives of the thesis

In this thesis, we are looking for the necessary control tools to improve the dynamic performance of parallel manipulators in terms of precision and robustness towards operating condition changes. Two strategies can be considered to achieve this goal: i) designing robust control solutions, and ii) compensating for the errors coming from the motor drivers, the actuators dynamics, the friction in the articulations, etc.

Main contributions of the thesis

The main contributions of this thesis revolve around improving the dynamic performance of parallel manipulators by the proposition of new advanced control schemes being robust towards changes of operating conditions, uncertainties, and external disturbances. In this framework, the following control solutions were proposed:

1. A new time-varying feedback RISE control based on nonlinear feedback gains instead of static ones.

2. A novel model-based super-twisting sliding mode control that incorporates (i) a feedforward dynamic term, (ii) the super-twisting algorithm, and (iii) a feedback stabilizing term.

3. Actuator and friction dynamics formulation integrated within a model-based closed-loop PD control with computed feedforward.

The proposed control solutions have been studied and validated in real-time experiments on several available parallel manipulator prototypes.

Organization of the thesis

The thesis is organized as follows:

Chapter 1 provides the context, problem formulation, and the state of the art of this thesis. The main differences between serial and parallel manipulators are addressed. A historical overview of parallel robots as well as some of their potential applications are included. A survey on the existing control schemes proposed in the literature
and implemented to parallel manipulators is included. The chapter ends up with the main objectives of the thesis as well as the main contributions.

**Chapter 2** is devoted to the description and modeling of the available parallel robot prototypes. The existing prototypes are grouped into two different categories: non-redundant and redundant parallel robots. The mechanical structure description, the kinematics, and the dynamics of each prototype are addressed within this chapter.

**Chapter 3** provides a detailed explanation of the main proposed control solutions in this thesis. The contribution for each adopted control strategy is addressed and explained. The applied control solution for eliminating the internal forces in case of redundantly actuated parallel manipulators is explained at the end of this chapter.

**Chapter 4** includes the presentation and discussion of the obtained experimental results by the proposed control solutions. The results for each experimental test are plotted, commented, and discussed in terms of the dynamic performance of the parallel manipulator. The chapter ends up with a conclusion regarding the proposed control solutions and the obtained results.

Finally, the thesis finishes up with a general conclusion in which a summary of the main contributions of the thesis is invoked as well as some perspectives on the extensions of the proposed controllers.

**Publications of the author**

**Submitted to international journals**


**Published in international conferences**


Published book chapters

CHAPTER

1

Context, problem formulation and state of the art

Contents

1.1 Introduction .................................................. 18
1.2 Robotic manipulators ........................................ 19
1.3 Parallel versus serial manipulators ....................... 20
1.4 A historical overview of parallel robots ................. 24
1.5 Potential applications of parallel manipulators .......... 29
1.6 Control problem formulation ............................... 36
1.7 Dynamic modeling of parallel manipulators .......... 41
1.8 Overview of motion control solutions for parallel manipulators ...... 46
1.9 How can we improve the performance of PKMs from a control point of view? 64
1.10 Objectives of the thesis .................................... 65
1.11 Main contributions of the thesis ......................... 65
1.12 Conclusion ................................................. 67
1.1 Introduction

In this chapter, the concept of a robotic manipulator is introduced distinguishing between two main types of manipulators: serial and parallel. The positive points of parallel manipulators compared to serial ones are highlighted such as more stiffness, higher accuracy, greater payload/weight ratio, and better dynamic performance. The long history of parallel robots and their typical and modern applications in different fields are addressed showing the importance of such mechanisms in industries, medical applications, space, machining, agriculture, etc.

From a control point of view, the problem formulation of this thesis is described. Control of PKMs is considered a challenging task in the literature due to their complex and nonlinear dynamics, abundant uncertainties, parameter variations, and actuation redundancy. Control plays a significant role in fulfilling the requirements of the general targeted tasks. Examples on those tasks are high-speed pick-and-place motion cycles, accurate positioning, and precise surgical treatments. The dynamic performance of parallel manipulators can be evaluated through the tracking precision of the desired trajectory, robustness towards changes in operating conditions (speed, acceleration and parameters variation), and stability insurance in the presence of uncertainties and external disturbances. A good control design should take into account the nonlinear dynamics of PKMs, the abundant uncertainties, and the time-varying parameters.

A general overview of the major control strategies of parallel manipulators proposed in the literature is provided in this chapter. A brief discussion on each controller is carried out enlightening the positive and negative points of each strategy. One can distinguish between two types of control strategies of parallel manipulators: kinematic and dynamic control. Because considering the dynamic model within the control design can greatly enhance the dynamic performance of parallel manipulators, the majority of the existing control schemes are full or partial dynamic control strategies. Some of the controllers compensate for a part of the dynamics while the others compensate for all the modeled dynamics. Parameter-identification techniques exist in two modes, offline and online.

The main objectives of the thesis consist of improving the dynamic performance of
parallel robots via control design by proposing robust control strategies and compensating for the errors coming from actuator and friction dynamics.

1.2 Robotic manipulators

The high demand for improving the quality of products and reducing the number of workers employed in several areas of production is guiding the industry nowadays towards automation. It is all about the use of robotic systems instead of humans, such as mechanical manipulators, equipped with control systems.

The need of robotic manipulation has been extended beyond industries [Chua et al., 2003] towards more areas such as space [Yoshida, 2009], underwater robotics [Sivcev et al., 2018], chemically active environments [Svejda and Goubej, 2012; Goubej and Svejda, 2013], household life [Jain and Kemp, 2010], agriculture [Monta et al., 1995], horticulture [Tillett, 1993], and medicine [Preising et al., 1991; Davies, 2000].

According to the International Federation of Robotics under standard ISO 8373, a robot manipulator can be defined as follows [Robot-ISO]:

"A manipulating industrial robot is an automatically controlled, reprogrammable, multipurpose manipulator programmable in three or more axes, which may be either fixed in place or mobile for use in industrial automation applications".

In other words, a series of rigid bodies called links interconnected through hinges or joints that provide relative motion of two consecutive bodies is called a manipulator [Kelly et al., 2005]. The conventional robotic manipulator is the arm resembling the human hand. To set the robotic arm in motion, the joints are actuated through actuators such as electric, hydraulic or pneumatic motors. The extremity of the robotic arm holds usually the so-called end-effector that is responsible for performing the required task (such as gripper). The end-effector can execute translational and rotational motions in the workspace. These motions are known as Degrees of Freedom (DOFs) of the manipulator representing the position and orientation of the end-effector. The manipulator is accompanied by a control system (computer, PLC, etc.) that directs and regulates the motion of the actuated articulations through the control loops taking feedbacks from the sensors measuring the states
of the manipulator (e.g. position, velocity, etc.) [Sciavicco and Siciliano, 2000]. Figure 1.1 illustrates a schematic diagram of the widely used robotic manipulator in assembly operations named SCARA robot \(^1\) [Spong and Vidyasagar, 2004]. The mechanical design of the SCARA robot allows its end-effector to manipulate within three translational and one rotational axes enabling a circular workspace with a certain depth.

![Illustrative diagram of the 4-DOF SCARA robotic manipulator.](image)

In literature, one can distinguish between two main kinematic structures of robotic manipulators: serial and parallel. On the one hand, serial manipulators are defined as open kinematic chains in which there is only one sequence of links connecting the two ends of the chain (the base and the end-effector). For instance, SCARA robot is one of the typical examples of serial manipulators. On the other hand, when there is more than one sequence of links connecting the base and the end-effector, it is called a closed kinematic chain, characterizing a parallel manipulator. One additional category of manipulators worths to be mentioned, it results from the combination of a serial and a parallel structure leading to a hybrid manipulator as illustrated in Figure 1.2.

### 1.3 Parallel versus serial manipulators

The main difference between a serial and a parallel manipulator appears obviously from the kinematic structure of each one. A serial manipulator features an open kinematic

---

1. "SCARA" refers to Selective Compliance Articulated Robot Arm.
1.3. PARALLEL VERSUS SERIAL MANIPULATORS

Figure 1.2 – Robotic manipulator structures: (a) serial, (b) parallel, and (c) hybrid.

structure while a parallel one is a closed kinematic chain. A generalized definition of parallel manipulators has been given by Merlet in his book "Parallel Robots" as follows [Merlet, 2006]:

\[
\text{A generalized Parallel Kinematic Manipulator (PKM) is a closed-loop kinematic chain mechanism whose end-effector is linked to the base by several independent kinematic chains.}
\]

Thanks to their mechanism, parallel manipulators have several advantages over their serial counterparts. However, they still suffer from some considerable disadvantages. Indeed, parallel manipulators are not replacing serial ones, but they offer various advantages for certain applications [Patel and George, 2012].

The different characteristics of serial and parallel manipulators are compared in terms of various aspects in the following [Patel and George, 2012; Pandilov and Dukovski, 2014].

1.3.1 Singularities

For serial manipulators, a singular configuration is a point in the workspace at which the end-effector may lose one or more degrees of freedom. For instance, the robot may lose the ability of motion in some directions despite the motion of its joints. Numerically, singularity happens when the Jacobian matrix (a transformation matrix that relates the
joint and Cartesian velocities by \( \dot{\mathbf{q}} = J^{-1} \dot{\mathbf{X}} \), where \( \dot{\mathbf{q}}, \dot{\mathbf{X}} \) are the joint and Cartesian velocities respectively) becomes ill-conditioned and may not be invertible resulting in infinite joint rates with a stationary end-effector [Tsai, 1999]. This type of singularities may occur at the boundary of the workspace (Boundary Singularities) when the manipulator is either in a fully stretched-out or a folded-back configuration or when the actuators reach their mechanical limits. It can also occur inside the workspace (Interior Singularities) when two or more joint axes become linearly dependent.

For parallel manipulators, the differential kinematic relationship between the joints and end-effector velocities is expressed as: \( J_q \dot{\mathbf{q}} = J_x \dot{\mathbf{X}} \), where \( J_q, J_x \) are the joint and Cartesian Jacobian matrices respectively (more details about this relationship are addressed in Chapter 2). Thus, one can distinguish among three different types of singularities [Merlet, 2006; Tsai, 1999]:

1. **Inverse kinematic singularities (serial singularities):** Singularity in \( J_q \) which means that the determinant of \( J_q \) becomes zero and it is no longer invertible. This type of singularity is similar to that of the serial manipulators already discussed above.

2. **Direct kinematic singularities (parallel singularities):** Singularity in \( J_x \) which means that the determinant of \( J_x \) becomes zero and it is no longer invertible. This means that the end-effector may have non-zero velocity even though the actuated joints have zero velocities. Unlike a serial manipulator, the end-effector gains one or more uncontrollable degrees of freedom. In this case, the end-effector will not be able to resist forces or moments in some directions leading to harmful behavior.

3. **Combined singularities:** Singularities in both \( J_q \) and \( J_x \) where the end-effector can be in a static position for which the actuators undergo some infinitesimal motions or vice versa.

One can conclude that similar types of singularities can occur in serial and parallel manipulators with two additional types special for parallel robots. Consequently, singularity analysis of parallel manipulators is much more complex to be performed than the case of serial ones.
1.3.2 **Workspace**

In comparison to serial manipulators, parallel manipulators possess less and limited dexterous workspace. This limitation is due to the geometrical and mechanical limits of the design such as the physical constraints of spherical and universal joints. The range of motion of actuators in the case of parallel manipulators is less than the one of serial manipulators due to the design and link interference. The rotational motion capabilities of the end-effector in parallel manipulators are limited compared to serial ones. Moreover, the abundant of singularities in parallel manipulators also limit the workspace to a smaller free-singularity region.

1.3.3 **Payload/weight ratio**

Unlike serial manipulators, the handled payload can be shared by the actuators and all the parallel kinematic chains in the case of parallel manipulators. Hence, the load-carrying capacity of parallel robots is much greater than that of serial ones.

1.3.4 **Stiffness and dynamic performance**

Robot stiffness can be defined as the resistance against the deflections caused by external forces and/or moments exerted on the end-effector [Angeles, 2007]. The overall stiffness of a robotic manipulator is related to several factors such as rigidity of the links, mechanical transmission system, compliance errors, actuators, and controller. Due to the open kinematic structure of serial manipulators, the errors of the actuated joints are accumulated from one joint to another deteriorating its stiffness. Using heavy links in serial manipulators may increase their stiffness but it will surely reduce their dynamic performance. Moreover, as links become lighter and the arms longer, the stiffness will be lower as well as the payload/weight ratio [Klimchik et al., 2013]. In parallel manipulators, the actuators are located at the fixed-base (one actuator for each kinematic chain) and the Cartesian errors are averaged at the end-effector instead of accumulation as in serial manipulators. Thus, we can achieve higher stiffness properties and higher dynamic performance (with lightweight links and low inertia) simultaneously.
Table 1.1 – The main dynamic properties of serial and parallel manipulators.

<table>
<thead>
<tr>
<th>Property</th>
<th>Serial manipulators</th>
<th>Parallel manipulators</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Singularities</strong></td>
<td>Inverse kinematic singularities</td>
<td>Inverse, direct, and combined kinematic singularities</td>
</tr>
<tr>
<td><strong>Workspace</strong></td>
<td>Large</td>
<td>Limited</td>
</tr>
<tr>
<td><strong>Workspace/robot size</strong></td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td><strong>Payload/weight ratio</strong></td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td><strong>Stiffness</strong></td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td><strong>Dynamic performance</strong></td>
<td>Poor</td>
<td>Very high</td>
</tr>
<tr>
<td><strong>Accuracy</strong></td>
<td>Low</td>
<td>High</td>
</tr>
</tbody>
</table>

### 1.3.5 Accuracy

Since parallel manipulators are stiffer than serial ones, then the position accuracy of parallel manipulators is better. One of the main problems of serial manipulators is their cantilever structure making them more sensitive to bending with high payloads. Moreover, serial manipulators can suffer more than parallel ones from vibrations at high-speed motions leading to a low accuracy [Pandilov and Dukovski, 2014].

To this point, the main characteristics of serial and parallel manipulators are compared showing the strength points of parallel manipulators as well as their main drawbacks. Table 1.1 summarizes the main differences between serial and parallel manipulators in terms of dynamic properties.

### 1.4 A historical overview of parallel robots

Theoretical problems of parallel architectures can be originated back to the 17th century to the English architect Sir Christopher Wren. More theoretical studies concerning parallel mechanisms have been done later in the 19th century by Cauchy [Cauchy, 1813], Lebesgue [Lebesgue, 1867] and Bricard [Bricard, 1897] [Merlet, 2006].
Figure 1.3 – The spherical parallel mechanism proposed in 1928 by James E. Gwinnett [Gwinnett, 1928].

Things became more interesting when the first patent of a motion platform based on a spherical parallel mechanism was filed in 1928 by J.E. Gwinnett [Gwinnett, 1928]. His invention, illustrated in Figure 1.3, was supposed to be used as a dynamic cinema, but it was never built because the industry at those days was not ready for such complexity.

After one decade, a 5-DOF automated spray painting machine based on a parallel structure was invented by Willard P.V. Pollard. Willard’s junior filed a patent in 1942 on the spray painting machine that consists of two parts: an electrical control system and a mechanical manipulator [Pollard, 1942]. A license was granted to the DeVilbiss company being as the first industrial robot supplier. Figure 1.4 shows this ingenious invention consisting of three motors responsible for positioning the tool head, while its orientation is controlled by two other motors fixed at the base and connected to the tool head through flexible rotary cables.

In 1947, Gough proposed a 6-DOF hexagonal-based parallel mechanism that allows the positioning and orientation of a moving platform [Bovey, 2003]. This invention responds to problems of aero-landing loads and is capable of testing the tyre wear. Six linear actuators modify the lengths of the connecting links controlling the position and orientation of the moving platform and consequently the wheel as can be seen in Figure 1.5. In 1955, the Gough tyre-testing prototype was built and still used until 2000 (see Figure 1.6).

Based on a similar concept to the Gough platform, Stewart proposed in 1965 a 6-DOF motion platform as a flight simulator [Stewart, 1965]. The Stewart’s paper appeared in the
Figure 1.4 – The spray painting 5-DOF parallel robot proposed in 1942 by Willard L.V. Pollard [Pollard, 1942].

Figure 1.5 – The original Gough platform in 1954 [Bonev, 2003].
1.4. A HISTORICAL OVERVIEW OF PARALLEL ROBOTS

Figure 1.6 – Gough platform prototype, called Universal Rig, exhibited in the British National Museum of Science and Industry in 2000 [Bonev, 2003].

proceedings of the British IMechE, and Dr. Gough was one of the reviewers who recalled the existence of his platform. Figure 1.7 shows the proposed Stewart’s platform or Gough-Stewart platform.

![Diagram of Stewart platform](image1)

Figure 1.7 – Schematic view of Stewart platform proposed as a flight simulator [Stewart, 1965].

However, Stewart is not considered the father of flight simulators even though his pa-
Figure 1.8 – The first flight simulator based on a hexapod structure as in the mid of 1960s. [Bonev, 2003].

...per had a great impact on the development of parallel robots especially those of hexapod architectures. The US engineer, Mr. Cappel, an employer at the Franklin Institute Research Laboratories in Philadelphia, reached out the same octahedral hexapod arrangement as the one proposed by Dr. Gough at the request of Sikorsky Aircraft Division of United Technologies for design and construction of a 6-DOF helicopter flight simulator [Bonev, 2003]. A patent was filed in 1974 and then the first-ever flight simulator based on the octahedral hexapod has been manufactured (see Figure 1.8).

The most successful parallel robot design, the Delta robot, was introduced in 1985 when Professor Clavel at EPFL (École Polytechnique Fédérale de Lausanne) got his ingenious idea of using lightweight parallelograms to build a parallel robot capable of moving the end-effector with fixed orientation and achieving high accelerations (up to 50 G) [Clavel, 1990]. It was a relevant solution for pick-and-place applications such as food packaging in industry at that time. Over the years, Delta robot occupied the market in a terrible way, where more than 10,000 units were in use worldwide [Bonev, 2001]. Figure 1.9 shows the original design of the Delta robot whose end-effector can manipulate in three translational and one rotational degrees of freedom.
1.5 Potential applications of parallel manipulators

The potential characteristics of parallel manipulators such as high precision, high-speed, good dynamic performance, and high payload/weight ratio made them good candidates for several applications [Patel and George, 2012; Pandilov and Dukovski, 2014]. Parallel robots become nowadays a must for some applications such as industrial packaging and motion simulators. Day by day, parallel robots become more popular in various fields such as industrial applications, motion simulators, machining applications, space applications, synchrotrons, medical applications, and agricultural applications.

1.5.1 Industrial applications

In addition to the 6 DOFs Gough's platform hexagonal-based parallel mechanism that was used for tyre-testing as discussed before and illustrated in Figure 1.6, several parallel robot prototypes are used for industrial applications nowadays.

The most popular robots for industrial applications are the Delta robot and the Delta-like robots used for sorting and food packaging production lines. Figure 1.10 shows the
FlexPicker Delta robot of ABB company, the leader in state-of-the-art of high-speed robotic picking and packing technology. This robot of cylindrical workspace can handle a maximum payload between 1 Kg and 8 Kg (depending on the robot’s version).

A 4-DOF Delta-like PKM, Adept Quattro robot, shown in Figure 1.11 is considered the fastest parallel robot in the world. This robot was originally designed at Laboratoire d’Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM) named Par4 [Nabat et al., 2005]. The general structure of Par4 was inspired from Delta robot with one additional kinematic chain and articulated traveling-plate leading to higher stiffness near the workspace boundaries.

![Figure 1.10 – IRB 360 FlexPicker (ABB technology)](image1)
![Figure 1.11 – The high-speed Adept Quattro robot (Adept technology)](image2)

The modern application of the Delta robot in the smartphone development area is automated mobile testing. Manual tests that need to be done by a human operator during the mobile development process may result in some errors and mistakes. Moreover, robotizing such a process is more suitable and rapid for repetitive tasks. TestDevLab, a private company founded in 2011, developed TapsterBot robot (see Figure 1.12) able to tap the smartphone screen and press physical buttons that are not tested by software [Guntis, 2017]. The Delta structure of TapsterBot provides a simple design with high speed and accuracy. The device can be used also for automating mobile applications on a smartphone such as games.
1.5.2 Motion Simulators

Gough-Stewart platform is the most used prototype for this kind of applications where various types of simulations needed (flight simulation, ship simulation, space simulation, car simulation, etc). The platform has 6-DOFs and comprises six pods, six spherical joints and six universal joints capable of handling very large payloads. One of the Boeing flight simulators is shown in Figure 1.13.

Hexapods are also used as swell simulators in the naval industry for material resistance testing or in the oil and gas industry to test the marinization of processes on floating production units (see Figure 1.14). SYMETRIE has realized a wide range of swell simulators
using hexapod technology achieving very real simulation with accurate trajectories [Assima et al., 2015a,b]. Furthermore, SYMTRIE developed hexapods to be used in the study of hydrodynamic effects of swell, the phenomena of sloshing or cavitation for example [Symtrie, 2017b]. The hexapod is attached downwards to a trolley, which moves all along the basin. The mobile platform of the hexapod moves the ship model to apply on it the representative forces of those generated by the swell for testing purposes (see Figure 1.15).

![MISTRAL hexapod swell simulator at HOPPE Marine.](image1)

![MISTRAL hexapod with a ship model at the wave basin of Ifermer Boulogne sur Mer, France.](image2)

**1.5.3 Machining applications**

Machining is the process of a controlled material-removal that makes the desired deformation in the shape and size of raw materials. Parallel robots providing large accuracy, low vibration, high acceleration capabilities thanks to the light moving parts, and high stiffness due to the closed-chains structure are welcomed candidates to be the future machining robots. A machine tool based on a parallel structure is proposed in [Toyama et al., 1998] proceeding towards an improved machining with a large stiffness (see Figure 1.16).

Moreover, Figure 1.17 shows a machining device, ARROW robot, developed at LIRMM
1.5. POTENTIAL APPLICATIONS OF PARALLEL MANIPULATORS

Figure 1.16 – Toyoda HexaM PKM-based machine tool [Toyama et al., 1998].

with the architecture of a parallel manipulator capable of executing five degrees-of-freedom in a large workspace [Shayya, 2015].

Figure 1.17 – CAD view of ARROW robot prototype.

1.5.4 Space applications

Thales Alenia Space (TAS), one of the leading satellite manufacturers in Europe, is using a system, developed by SYMETRIE, composed of two hexapods helping operators during the mirror assembling tasks and meeting the quality requirements of satellite manufacturing [Symetrie, 2017a]. The first hexapod positions the mirror, and the second hexapod measures the mirror position in real-time with extreme accuracy (see Figure 1.18). This optical micro-positioning system resolution reaches till 1 μm.
1.5.5 Synchrotrons

Synchrotrons are special types of particle-accelerators in which the accelerating particle beam travels around a fixed closed-loop path. In [Ju et al., 2017], an instrument that uses a high-resolution angular positioning hexapod to exploits the ongoing revolution in synchrotron sources is described (see Figure 1.19). A high vacuum diffractometer (shown in Figure 1.20) integrating two high precision hexapods was developed by SYMETRIE and used by SIRIUS beamline to improve the studying of semiconductor nanostructures and functional oxides in conventional and grazing incidence conditions [Abiven et al., 2018].

1.5.6 Medical applications

Parallel robots are also used for medical purposes such as spine surgery, surgical robot, and computer-aided surgery. One of the ideas is to use the PI M-850 Hexapod, a 6-DOF parallel-kinematics micro-positioning system, thanks to its high stiffness, load capacity and accuracy [Dalvand and Shirinzadeh, 2012]. The principal goals of this new technology aim at high safety of micro-surgical procedures and feasibility of micro-therapy (see Figure 1.21).
1.5. POTENTIAL APPLICATIONS OF PARALLEL MANIPULATORS

Figure 1.19 – Photograph of the instrument installed at Advanced Photon Source station 12ID-D [Ju et al., 2017].

Figure 1.20 – High vacuum diffractometer at SIRIUS beamline.

Figure 1.21 – Hexapod medical robot of high precision positioning.

1.5.7 Agricultural applications

The ecoRobotix developed recently a mobile parallel robot, based on Delta robot, to be used for rapid plant phenotyping on the field as seen in Figure 1.22 [ecoRobotix, 2011]. The platform operates fully autonomously and can be controlled wirelessly by smartphone, in addition to an entire solar power system. The goal of this invention is to improve modern
agriculture and add value to farmers’ work. Delta robots are used thanks to their high-speed motions and swiftness for which they can reach up to 4000 movements per hour.

Figure 1.22 – The autonomous robot weeder of ecoRobotix depositing seeds in a targeted way.

At last, diverse applications can be fulfilled by parallel manipulators and the near future may hold for us fantastic and amazing usage of PKMs. A long path is still to be covered to exploit all parallel manipulator capabilities and take advantage of them in applications as can as possible.

1.6 Control problem formulation

The increased demand on parallel robots in several fields and applications always motivates the scientific community to study and improve these mechanisms aiming to fit the requirements of each corresponding application. There exist multiple aspects, illustrated in Figure 1.23, from which parallel manipulators can be analysed, synthesized and improved.

Some of the work in the literature address the design and architecture optimization problem seeking to improve parallel manipulator performances [Yang et al., 2004; Krut et al., 2006; Chablat and Wenger, 2007; Pierrot et al., 2009], while others point out to stiffness problem [Dagalakis et al., 1989; Gosselin, 1990] trying to enhance it by compensating the vibration for example [Algermissen et al., 2004]. Modeling of parallel robots have also
excited the research community to improve it, searching for the best model that fits the control requirements. Modeling of PKMs includes the inverse and forward dynamic formulation [Khalil and Guegan, 2004; Briot and Khalil, 2015], in addition to the kinematic relations [Besnard and Khalil, 2001; Bi and Jin, 2011]. Besides, a lot of papers in the literature mention the motion planning problem proposing some techniques to avoid the singularities in the workspace of the parallel robots [Dash et al., 2005; Reveles et al., 2016] or to generate optimal trajectories that allow the parallel manipulators exiting the singularity loci [Pagis et al., 2013]. Last but not least, control of parallel robots have possessed a wide interest from scientific researchers even though it is considered a challenging task, but advanced control schemes can provide PKMs enhanced accuracy and precision even at high-dynamic operating conditions [Paccot et al., 2009; Azar et al., 2017].

In this thesis, we aim at improving the dynamic performance of parallel manipulators in different scenarios, from a control point of view, by proposing robust control solutions and compensating for the errors coming from friction, actuator dynamics, etc.

### 1.6.1 Control challenges of parallel manipulators

Control of PKMs is often considered in the literature as a very challenging task since of their complex nonlinear dynamics inherited from their closed-loop structure, abundant uncertainties, parameters variation, and actuation redundancy [Chemori et al., 2013; Chemori, 2017]. Figure 1.24 summarizes the most considerable control challenges that can reduce the dynamic performance of parallel robots in case they are not compensated.

#### 1.6.1.1 Nonlinear complex dynamics of PKMs

The nonlinear dynamics of parallel manipulators make the control task hard to be accomplished for which the classical linear control approaches may fail to guarantee the stability at critical dynamic operating conditions such as high-speed operations [Khalil, 2002]. Touching high-speed acceleration limits of parallel robots can increase considerably the effect of nonlinearity leading to mechanical vibration issues [Natal et al., 2015]. Thus, the need for advanced nonlinear control strategies arises to fit parallel robot application requirements of simultaneous high speed and high precision.
Figure 1.23 – Overview of general research topics of parallel manipulators.

The linear stability analysis tools based on frequency analysis (Nyquist criterion and displayed graphically with Nyquist plots, Nichols charts, or bode gain and phase plots [Nise, 2004]) are not usable for such nonlinear systems, and thus other complicated and
1.6. CONTROL PROBLEM FORMULATION

Control challenges of parallel robots

![Chart](Figure 1.24 – Overview of the considerable control challenges of PKMs.)

difficult tools are used such as Lyapunov stability analysis [Slotine and Li, 1991; Khalil, 2002].

Furthermore, their closed-loop kinematic structure gives rise to coupled dynamics that need careful control synchronization between the actuators. Any failure in one of the actuators affects the whole structure and may lead to possible damage.

1.6.1.2 Uncertainties in PKMs: structured and unstructured

Uncertainties are simply the differences or errors between the formulated dynamic model and the real plant which is in our case the parallel robot [Qaisar and Mahmood, 2016]. In the literature, two types of uncertainties are defined: structured and unstructured.

On the one hand, uncertainties emerging from wear of the components, geometric errors, modeling simplifications (actuator dynamics, friction, etc.) and sensor noise are considered unstructured. On the other hand, structured uncertainties can appear in the form of inaccurate knowledge of dynamic parameters or parameter variations due to the operating environment (handled payload in case of pick-and-place PKMs, contact force with a workpiece in case of machining PKMs, etc.) [Siciliano and Khatib, 2016; Chemori, 2017].

Taking into account those uncertainties in the control design can produce high dynamic performances, a good level of accuracy, and robustness towards dynamic operating condition changes such as high-speed motions and payload variation.
1.6.1.3 Actuation Redundancy in PKMs

When the number of actuators is greater than the number of DOFs of the end-effector, the parallel manipulator is called redundantly actuated [Merlet, 2006]. The difference between the actuation sources and output DOFs of the end-effector is called the degree of redundancy. Figure 1.25 shows an illustration of actuation redundancy through adding one more kinematic chain with one actuator. The degree of redundancy in the redundant platform of Figure 1.25 is one (four actuators and three DOFs, \((x, y, \text{ and } \theta)\)).

\[ \text{Actuated joint} \quad \text{Passive joint} \]

(a) \hspace{2cm} (b)

Figure 1.25 – Illustrative diagram of actuation redundancy. (a) Non-redundant 3-DOF PKM, (b) Redundantly actuated 3-DOF PKM.

Indeed, the redundant actuator holds several advantages to the parallel manipulator described as follows [Mueller, 2009, 2010]:

- Improves the force transmission property and the manipulator stiffness.
- Increases the payload and acceleration capabilities.
- Reduces power consumption of the individual drives and distributes the load.
- Eliminates singularities and enlarges the workspace.

However, actuation redundancy is featured by generating internal prestress, due to the antagonistic forces of the redundant actuation, without generating end-effector forces
1.7. Dynamic modeling of parallel manipulators

Dynamics of parallel manipulators were investigated a lot in the literature and they are still an open problem presenting a notable complexity owing to the closed-loop structure of PKMs. Dynamic model represents a relation between the actuated joint forces $\Gamma$ and the end-effector position, velocity and acceleration ($X, \dot{X}$ and $\ddot{X}$). There are two types of dynamic models:

- **Direct dynamics**: being given the actuated joint forces, we can determine the position, velocity and acceleration of the end-effector.
  \[ \dot{X} = f(\Gamma, X, \dot{X}) \]  
  \[ \dddot{X} = f(\Gamma, X, \dot{X}, \ddot{X}) \]

- **Inverse dynamics**: being given the position, velocity and acceleration of the end-effector, we can determine the actuated joint forces.
  \[ \Gamma = f(X, \dot{X}, \ddot{X}) \]

Indeed, there is no one general approved procedure to formulate the dynamics of parallel manipulators because of their inherent complexity and kinematic constraints [Taghirad, 2013]. There exist several methods in the literature to derive the dynamic equation for a general parallel manipulator.

Most of these methods use the following procedure to derive the dynamic equation [Merlet, 2006; Taghirad, 2013]:
• First, the kinematic chains (limbs) are disconnected from the moving platform obtaining a tree-structure form of multiple open-loop chains (see Figure 1.26).

• Then, a local dynamic model of each limb is derived using their local generalized coordinates (position vector of the limb joints, the position of the center of mass of the moving platform, and its orientation angle).

• Eventually, the obtained dynamic models are combined to get the whole model of the manipulator in terms of its global generalized coordinates.

![Image of parallel kinematic manipulator with loop constraints](image.png)

Figure 1.26 – Definition of the loop constraints for a planar 2-DOF redundantly actuated PKM [Mueller, 2011].

The main existing methods for developing the dynamic model of parallel manipulators in the literature can be introduced as follows: Newton-Euler method [Dasgupta and Choudhury, 1999], Lagrangian method [Cheng et al., 2003], and Virtual work principle / D’Alembert principle [Wang and Gosselin, 1998; Tsai, 2000].

### 1.7.1 The standard inverse dynamic model

For motion control, the inverse dynamic formulation of a parallel manipulator of n actuators and m DOFs (such that \( n \geq m \)), obtained by Lagrange or virtual work approaches, can be expressed in joint space as follows [Siciliano and Khatib, 2016]:

\[
M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \Gamma(t)
\]  

(1.3)

where
1.7. DYNAMIC MODELING OF PARALLEL MANIPULATORS

- \( q, \dot{q}, \ddot{q} \in \mathbb{R}^n \) are the position, velocity and acceleration vectors of the actuated joints respectively,
- \( M(q) \in \mathbb{R}^{n \times n} \) is the total mass and inertia matrix,
- \( C(q, \dot{q}) \in \mathbb{R}^{n \times n} \) is the centrifugal and Coriolis forces matrix,
- \( G(q) \in \mathbb{R}^n \) is the gravitational forces vector,
- \( \Gamma(t) \in \mathbb{R}^n \) is the input torques vector.

The inverse dynamics can be represented in Cartesian space (space of the end-effector) using the Jacobian transformations: 
\[
\dot{X} = J_m \dot{X}, \quad \ddot{X} = J_m \ddot{X} + J_m \dot{\dot{X}}, \quad \text{and} \quad F = J_m^\top \Gamma,
\]
with \( J_m \in \mathbb{R}^{n \times m} \) being the inverse Jacobian matrix of the manipulator. Then, the inverse Cartesian space model can be written as follows:
\[
M_x(q) \ddot{X} + C_x(q, \dot{q}) \dot{X} + G_x(q) = F(t) \quad (1.4)
\]
where
- \( X, \dot{X}, \ddot{X} \in \mathbb{R}^m \) are the position, velocity and acceleration vectors of the center of mass of the end-effector respectively,
- \( M_x(q) = J_m^\top M(q) J_m \) is the Cartesian mass and inertia matrix,
- \( C_x(q, \dot{q}) = J_m^\top M(q) \dot{J}_m + J_m^\top C(q, \dot{q}) J_m \) is the Cartesian centrifugal and Coriolis forces matrix,
- \( G_x(q) = J_m^\top G(q) \) is the Cartesian gravitational forces vector,
- \( F(t) \in \mathbb{R}^m \) is the input forces vector on the end-effector.

In case of redundant parallel manipulators, the direct Jacobian matrix can be calculated from the inverse one using the Moore-Penrose pseudoinverse matrix which can be used when a system of equations does not have a unique solution or has many solutions.

1.7.2 Properties of the dynamic model

The inverse dynamic model of parallel manipulators (1.3), as common for robotic manipulators, inherits some properties for its dynamic terms that are useful in designing the control schemes. The properties can be addressed as follows [Lewis et al., 2004; Kelly et al., 2005]:

- [Lewis et al., 2004; Kelly et al., 2005]:
1.7.2.1 Property of mass and inertia matrix:

**Property 1.** Mass and inertia matrix and its inverse are symmetric, positive definite, and bounded above and below as follows:

\[
\mu_1 I \leq M(q) \leq \mu_2 I
\]

\[
\frac{1}{\mu_2} I \leq M^{-1}(q) \leq \frac{1}{\mu_1} I
\]

where \(\mu_1\) and \(\mu_2\) are two positive scalars that can be computed. \(\mu_2\) can be function of \(q\) for some cases (for example if using prismatic joints). Likewise, the boundedness inequality can represented as follows:

\[
m_1 \leq \|M(q)\| \leq m_2
\]

where \(\|\cdot\|\) is the second norm of a matrix defined for any matrix \(A \in \mathbb{R}^{n \times m}\) as the square root of the maximum eigenvalue of the matrix \(A^*A\) being \(A^*\) the conjugate transpose of \(A\):

\[
\|A\| = \sqrt{\lambda_{\text{max}}(A^*A)}.
\]

\(m_1\) and \(m_2\) are two positive constants.

Furthermore, there exists a positive constant \(K_M > 0\) such that

\[
\|M(x)z - M(y)z\| \leq K_M \|x - y\| \|z\|
\]

\(\forall \ x, y, z \in \mathbb{R}^n.\)

1.7.2.2 Property of Coriolis and centrifugal matrix:

**Property 2.** Coriolis and centrifugal matrix is bounded as follows

\[
\|C(q, \dot{q})\| \leq K_{C_1} \|\dot{q}\|
\]

where \(K_{C_1}\) is a positive constant and \(\|\cdot\|\) the second norm of a vector or a matrix. Note that \(K_{C_1}\) can be function of \(q\) in some cases.
Moreover, the matrix \( H(q, \dot{q}) = \dot{M}(q) - 2C(q, \dot{q}) \) (or \( H(q, \dot{q}) = \frac{1}{2}\dot{M}(q) - C(q, \dot{q}) \)) is a skew-symmetric matrix holds the following:
\[
x^T H(q, \dot{q}) x = 0
\] (1.9)
\[
\dot{M}(q) = C(q, \dot{q}) + C^T(q, \dot{q})
\]
for any vector \( x \in \mathbb{R}^n \).

Furthermore, there exist positive constants \( K_{C_1} > 0 \) and \( K_{C_2} > 0 \) such that
\[
\|C(x, z)w - C(y, v)w\| \leq K_{C_1}\|z - v\|\|w\| + K_{C_2}\|x - y\|\|w\|\|z\| \quad (1.10)
\]
\[\forall \ x, y, z, w, v \in \mathbb{R}^n.\]

### 1.7.2.3 Property of gravity vector:

**Property 3.** There exists some positive constant \( g_0 \) bounding the gravity vector as follows:
\[
\|G(q)\| \leq g_0
\] (1.11)

Note that \( g_0 \) can be function of \( q \) in some cases.

Furthermore, there exists a positive constant \( K_G > 0 \) such that
\[
\|G(x) - G(y)\| \leq K_G\|x - y\| \quad (1.12)
\]
\[\forall \ x, y \in \mathbb{R}^n.\]

### 1.7.2.4 Linear formulation property of the dynamics:

**Property 4.** A fundamental property of PKMs is very essential for model-based adaptive controllers consists of linearity of the dynamics with respect to the parameters, such as inertia and masses [Ortega and Spong, 1989; Siciliano and Khatib, 2016]. All the constant parameters in the dynamic model are considered coefficients of known functions (linear and
nonlinear) of the generalized coordinates. Consequently, \( (1.3) \) can be written in a linear form as follows:

\[
Y(q, \dot{q}, \ddot{q})\Phi = \Gamma(t)
\]  \hspace{1cm} (1.13)

where \( Y(q, \dot{q}, \ddot{q}) \in \mathbb{R}^{n \times r} \) is the matrix of the known functions called regressor, \( \Phi \in \mathbb{R}^r \) is the vector the parameters.

### 1.8 Overview of motion control solutions for parallel manipulators

The motion control problem of parallel manipulators has been studied a lot in the literature. A vast number of control solutions have been proposed and experimented [Paccot et al., 2009; Azar et al., 2017]. The proposed control solutions of serial manipulators can be extended easily to parallel manipulators due to their similar structure of dynamic models. The proposed control strategies can be classified into two basic categories: kinematic and dynamic.

In kinematic control, the coupled structure of the PKM is decoupled into single independent axes. Then, a decentralized controller is developed for every single axis alone. This type of control is simple to be implemented, but it needs a special care for the synchronization among the actuators. Moreover, the dynamics of the manipulator are not taken into account which may deteriorate the dynamic performance leading sometimes to instability at high-speed motions.

Dynamic control strategies consider, fully or partially, the dynamics of the manipulator in their closed-loop design compensating for the high effect of the nonlinear dynamics. As a result, dynamic control provides much higher performance and robustness towards parameter variations compared to kinematic control [Taghirad, 2013]. A classification tree of the main proposed control schemes in the literature is demonstrated in Figure 1.27. All the listed controllers will be discussed in this section.

The existing control strategies can be implemented in joint space or Cartesian space. Joint space control is developed to allow the tracking of the desired joint trajectories while
Cartesian space control works on the direct tracking problem of the end-effector trajectory. In this section, we will clarify the difference between joint and Cartesian space controls. Then, a survey on the proposed control strategies in the literature will be made.
1.8.1 **Joint space versus Cartesian space control**

In joint space control, the joint coordinates \( q(t) \) track the desired joint trajectories \( q_d(t) \) via a feedback control algorithm. Indeed, the task space desired trajectory represents the motions to be executed by the end-effector, and it is generated in Cartesian space. Then, an inverse kinematic model is used to convert the desired end-effector coordinates to the joint ones as illustrated in Figure 1.28. By measuring the actual joint positions (for example, using encoders), the tracking error is computed, \( e_q = q_d - q \). Then, the necessary active joint torques are produced by the controller aiming at the best minimization of the tracking error.

![Figure 1.28 – Schematic illustration of the joint space control strategy.](image)

For a Cartesian space control, the feedback measurements are done directly on the end-effector pose and orientation, and the controller is developed using the Cartesian tracking error \( e_x = X_d - X \). The output of the controller is the needed force to correct the tracking error of the end-effector. This generated control force is converted through the Jacobian matrix to control torques at the level of the actuated joints as shown below in Figure 1.29.

![Figure 1.29 – Schematic illustration of the Cartesian space control strategy.](image)
1.8. OVERVIEW OF MOTION CONTROL SOLUTIONS FOR PARALLEL MANIPULATORS

Considering the availability of fast and accurate end-effector pose measuring tool, Cartesian space control seems to be more relevant for parallel manipulators due to several issues explained below [Paccot et al., 2009]:

- A joint space control acts to reduce the joint tracking error which is a geometric transformation of the end-effector tracking error (the interesting error to be reduced), and thus it can be affected by geometric errors. Consequently, Cartesian space control is considered to be more accurate since it is directly controlling the end-effector position.

- Since an inverse kinematic model is not used in Cartesian space control, the constraints on kinematic identification could be avoided. A dynamic identification, which is easier than kinematic one, could be enough to achieve the desired control performance.

- Any disturbance that may lead to a change in the end-effector posture can be observed by a Cartesian space control while can not be observed by a joint space control. For example, in the neighborhood of singularities, the end-effector position can be shifted without any change in the joint configuration (parallel singularities) [Chablat and Wenger, 1998; Husty, 1996]. Thus, a Cartesian space control could correct this shifting while joint space control fails to do that.

- The kinematic constraints in a joint space control leads to uncontrolled moves of the end-effector in the case of redundantly actuated PKMs. Cartesian space control can minimize or cancel these internal forces ensuring better performance [Marquet et al., 2001; Pierrot et al., 2014].

As long as the fast and accurate measuring tools of the end-effector pose and orientation are still rare, the Cartesian space control is still implemented occasionally. The joint space control is the most employed strategy nowadays to solve the tracking control problem of parallel manipulators.
1.8.2 Kinematic control strategies

1.8.2.1 PD/PID control

The Proportional-Integral-Derivative (PID) controller is a closed-loop feedback mechanism that takes as input the position and velocity errors and generates the required control signal to correct the error between the desired trajectory and the actual one [Ziegler and Nichols, 1942]. This method has been employed widely in the industrial control systems thanks to its simplicity and easy implementation. The PID control equation for n-actuators parallel manipulator [Chaudhary and Ohri, 2016] is expressed in joint space form as follows:

$$\Gamma(t) = K_p e(t) + K_i \int e(\tau) d\tau + K_d \dot{e}(t)$$

where $K_p, K_d, K_i \in \mathbb{R}^{n \times n}$ are diagonal positive definite matrices representing the proportional, integral, and derivative gains respectively. $\dot{e} \in \mathbb{R}^n$ denotes the joint velocity errors. The proportional term effects the response of the system towards any disturbance or change in the error. The steady-state error and general dynamic performance can be improved by the integration part. The derivation term provides enough damping and reduces the oscillations of the system. High derivative gains can lead to signal noises stimulating the resonance frequency of the robot.

The integral term works on accumulating the error overall the time of operation. In the case of a large change in the error, it will accumulate a significant error during the rise leading to an excess overshooting and control input saturation. For this reason, the integral term is omitted sometimes ($K_i = 0$) [Su et al., 2006] or treated with anti-windup integrator [Kumar and Negi, 2012].

A Cartesian PID control fed by measurements provided by a vision system is proposed and applied to a 2-DOF planar parallel manipulator in [Garrido and Trujano, 2019]. The disturbances in such a system appear as time-varying at the visual level. The stability analysis and the experiments provide reasonable performance in spite of uncertainty on the Jacobian matrix associated with the active joints.
1.8. OVERVIEW OF MOTION CONTROL SOLUTIONS FOR PARALLEL MANIPULATORS

1.8.2.2 Nonlinear PD/PID control

The Nonlinear PD/PID (NPD/NPID) controller shares the same structure of the standard PID controller with the use of nonlinear time-varying feedback gains instead of fixed gains [Jingqing, 1994]. Those nonlinear gains can be a function of the system errors, control input, and other parameters. This allows the online automatic change of the gains while the robot is executing a task. The generated values of the gains can deal with the instant operating conditions to reduce the error as possible. The general form of the control equation of a NPID controller is written as follows:

\[
\Gamma(t) = K_p(\cdot)e(t) + K_i(\cdot)\int e(\tau)d\tau + K_d(\cdot)\dot{e}(t)
\]  

(1.15)

where \(K_p(\cdot), K_d(\cdot), K_i(\cdot) \in \mathbb{R}^{n \times n}\) are diagonal matrices representing the nonlinear feedback gains for each axis. The NPID control implemented on a 6-DOF parallel manipulator in [Duan et al., 2004] shows better dynamic performance than the classical PID control. It guarantees stability when changing the operating conditions as well as it ensures better robustness towards uncertainties and disturbances. Furthermore, in [Ouyang et al., 2002], a NPD control design obtained good trajectory tracking performance for a 2-DOF parallel manipulator compared to a simple PD controller.

For a better understanding of the NPID control, consider the nonlinear function proposed in [Jiang and Gao, 2001] to be used for computation of the nonlinear feedback gains expressed as follows:

\[
y = f(x, \alpha, \delta) = \begin{cases} 
|a|^\alpha \text{sign}(a), & \text{if } |a| > \delta \\
\delta^{\alpha-1}, & \text{if } |a| \leq \delta 
\end{cases}
\]

(1.16)

where \(x\) is the input to this function being \(e, \dot{e}\) or \(\int e\), \(y\) is the output of this function, \(\alpha\) and \(\delta\) are two positive constants (0 < \(\alpha\) ≤ 1). For \(\alpha = 1\), the linear relation appears again as \(y = x\). \(\delta\) is usually chosen as a small positive value to get a linear relation around the origin (see Figure 1.30) avoiding numerical problems that may result because of excessive-high gains in that zone. The nonlinear relation shown in Figure 1.30 gives high gain values for small \(x\) and small gain for large \(x\) resulting in strong robustness towards variations of the operating conditions as well as better performance than conventional PID control. Therefore, the
nonlinear feedback gains of (1.15) are given as follows:

\[
K_p(\cdot) = \text{diag} \{k_{p1} f(\varepsilon_1, \alpha_{p1}, \delta_{p1}), \ldots, k_{pn} f(\varepsilon_n, \alpha_{pn}, \delta_{pn})\}
\]

\[
K_i(\cdot) = \text{diag} \{k_{i1} f(\int \varepsilon_1, \alpha_{i1}, \delta_{i1}), \ldots, k_{in} f(\int \varepsilon_n, \alpha_{in}, \delta_{in})\}
\]

\[
K_d(\cdot) = \text{diag} \{k_{d1} f(\dot{\varepsilon}_1, \alpha_{d1}, \delta_{d1}), \ldots, k_{dn} f(\dot{\varepsilon}_n, \alpha_{dn}, \delta_{dn})\}
\]

(1.17)

where \(k_{pj}, k_{ij}\) and \(k_{dj}\) are positive constant gains and \(j = 1, \ldots, n\) represents the \(j\)th actuator.

### 1.8.2.3 Decentralized sliding mode control

Sliding Mode Control (SMC) is a powerful tool against model uncertainties, unknown plant parameters, and parasitic dynamics [Shtessel et al., 2014]. One of the main advantages of SMC is that it works on reduced-order dynamics. The control objective is to allow the convergence of a sliding variable composed of the system states to zero, and thus converge the system states to the sliding surface in a finite time and remain on it moving the
state trajectory towards zero. The control law which is designed based on the sliding surface features a sign function which compensates the effects of the bounded disturbances. The sliding surface and the control law of the conventional SMC are expressed as follows [Shtessel et al., 2014]:

\[ S = hx_1 + x_2 \]
\[ u = -\text{sign}(S) \]  

(1.18)

where \( S \) is the sliding surface, \( x_1, x_2 \) are the system states, \( h \) is a positive gain, and \( u \) is the control law.

The sign function acts on switching between positive and negative feedback, which means that the system is switching between non-asymptotically stable structure and asymptotically unstable one. Therefore, the process continues until we obtain asymptotic convergent system [Salim Qureshi et al., 2018]. The zigzag behavior of the state trajectory seen in Figure 1.31 is due to the switching function ensuring the mandatory sliding overall the surface reaching the origin.

However, this switching produces a discontinuous control signal which is not adequate with real-time implementation. Several techniques have been investigated in the literature.
to avoid or reduce this hard switching known by chattering phenomena, such as Quasi-Sliding Mode control, integration of sign function, continuous high-order SMC [Shtessel et al., 2014].

Decentralized SMC has been applied to parallel manipulators such that none of the dynamic parts of the manipulator appear in the control equation. In [Begon et al., 1995], a fuzzy SMC approach is implemented on a 6-DOF parallel robot showing reduced tracking error during fast motions compared to PI control. However, it is hard to guarantee the stability conditions in such a fuzzy-based controller since the control input is specified by fuzzy rules. While in [Kumar et al., 2015], a continuous modified twisting controller is designed for the position control of a Stewart platform. The relevance of the proposed controller has been proved by numerical simulation showing the accurate positioning with the presence of matched disturbances.

1.8.2.4 $\mathcal{L}_1$ adaptive control

The $\mathcal{L}_1$ adaptive controller has been implemented experimentally for the first time on a parallel kinematic manipulator in [Bennehar et al., 2015a]. It is known for its decoupled estimation and control loops which enables fast adaptation while guaranteeing the robustness of the closed-loop system [Chengyu and Hovakimyan, 2006a,b]. The control input consisting of two independent terms, a fixed state-feedback term and an adaptive term that compensates partially for the nonlinearities of the system, is given as follows:

$$\tau(t) = A_m r(t) + \tau_{ad}(t)$$

(1.19)

where $A_m \in \mathbb{R}^{n \times n}$ is a Hurwitz matrix characterizing the transient response of the system, $r(t) = \dot{e} + \Lambda e$ is the combined error with $\Lambda \in \mathbb{R}^{n \times n}$ being a positive definite diagonal matrix, and $\tau_{ad}(t) = \hat{\phi}(t) \| r(t) \|_{\infty} + \hat{\sigma}(t)$ is the adapted nonlinear function gathering all the nonlinearities of the system including eventual external disturbances. $\hat{\phi}(t)$ and $\hat{\sigma}(t)$ estimates the nonlinear functions $\phi(t)$ and $\sigma(t)$ that represent all the nonlinearities and disturbances of the system. Figure 1.32 shows the general schema of the $\mathcal{L}_1$ adaptive controller implemented on parallel manipulators. Using projection-based adaptation law, the boundedness of the estimated parameters is ensured as well as the convergence of $r(t)$ to zero [Bennehar et al., 2015a]. Note that the adaptive control signal is treated with a low
pass filter, $C(s)$, before computed with the control law. $C(s)$ is a stable and strictly proper transfer function. The advantages of this control law can be briefly mentioned as follows [Hovakimyan and Cao, 2010]:

- The capability of using high adaptation gains leading to fast estimation with stability guaranteed thanks to its robustness-adaptation decoupling structure.
- Parameters to be estimated can be initialized by zero.
- Parameter excitation is not needed to ensure parameter convergence.
- Partial compensation of the nonlinear dynamics of the system without a priori knowledge about it.

The control scheme was experimentally implemented on a 4-DOFs parallel kinematic manipulator showing better performance than a PD controller in terms of tracking precision thanks to the nonlinearities compensation in the $L_1$ adaptive controller.

### 1.8.3 Dynamic control strategies

#### 1.8.3.1 PD control with gravity compensation

For some parallel manipulators with heavy weights, the system gravity is much greater than the maximal friction even if there is no payload handled. Thus, the positioning and
trajectory tracking accuracies of such manipulators are affected by gravity. The steady-state and dynamic precision may be worsened if the gravity effect is not taken into consideration. A PD control with gravity compensation term can be a good and simple solution assuming that the gravity parameters are known [Yang et al., 2010]. Its control equation in joint space model can be expressed as follows:

$$\Gamma(t) = K_p(\cdot)e(t) + K_d(\cdot)\dot{e}(t) + G(q)$$

(1.20)

In order to avoid the online computation of the gravitational term, which may take more time than a PD control, a desired gravity compensator can be used instead of the exact one. A PD control with desired gravity compensation can be expressed as follows [Kelly, 1997]:

$$\Gamma(t) = K_p(\cdot)e(t) + K_d(\cdot)\dot{e}(t) + G(q_d)$$

(1.21)

In [Niu et al., 2018], a PD control with desired gravity compensation was developed for controlling a dynamic brace based on a parallel-actuated structure. The experimental results show reduced influence of the brace system gravity and better performance than a simple PID controller.

1.8.3.2 Augmented PD control

Unlike PD control with gravity compensation, Augmented PD (APD) control compensates the effects of more dynamics such as inertia and mass matrix, centrifugal and Coriolis forces, and the gravity. As more dynamic parameters are taken into account in the dynamic model, the controller can be improved. The joint space expression of the APD controller is given as follows [Zhang et al., 2007]:

$$\Gamma(t) = M(q)\ddot{q}_d + C(q, \dot{q})\dot{q}_d + G(q) + K_p e(t) + K_d \dot{e}(t)$$

(1.22)

It can be observed clearly that the APD controller compensates the effect of the full nonlinear dynamics relying on the desired and measured trajectories. The last two terms represent the PD controller to ensure global asymptotic tracking. However, an online computation of the nonlinear functions of the dynamic model is required for this controller as well as a priori knowledge of the dynamic parameters.
1.8. OVERVIEW OF MOTION CONTROL SOLUTIONS FOR PARALLEL MANIPULATORS

1.8.3.3 Computed torque control

The main idea of Computed Torque (CT) control is that a nonlinear control strategy leads up to a linear closed-loop dynamic equation of the error. Consider an input signal in acceleration form for the dynamic equation of the parallel manipulator, one can write as follows:

\[ \Gamma(t) = M(q)u + C(q, \dot{q}) + G(q) \]  \hspace{1cm} (1.23)

with \( \ddot{q} = u \). Choosing \( u = \ddot{q}_d + K_p e(t) + K_d \dot{e}(t) \), feedforward acceleration plus PD feedback controller, and substituting in (1.23) give the closed-loop error system equation in the linear form as follows:

\[ \ddot{e} + K_d \dot{e} + K_p e = 0 \]  \hspace{1cm} (1.24)

where \( \dddot{e} \in \mathbb{R}^n \) is the joint acceleration error vector. The stability of the final obtained system can be proven as for conventional linear systems and the feedback gains are chosen to ensure the stability convergence. Figure 1.33 illustrates the general block diagram of the CT torque controller in joint space. The control equation can be written as follows [Asgari and Ardestani, 2015]:

\[ \Gamma(t) = M(q)\left(\ddot{q}_d + K_p e(t) + K_d \dot{e}(t)\right) + C(q, \dot{q}) + G(q) \]  \hspace{1cm} (1.25)

\[ \text{Figure 1.33 – Block diagram of the computed torque controller in joint space.} \]

In [Paccot et al., 2008], a CT control in Cartesian space is implemented and validated on the parallel Orthoglide robot using a fast exteroceptive measure. The controller was
reduced to its simplest expression leading to a better accuracy than the joint space CT control (because of no use of the kinematic model in Cartesian space control).

As shown in (1.23), all the known dynamics are used in the control input relying on the feedback measurements and estimations (position and velocity). This control can provide high dynamic performances, but it needs a good knowledge of the parameters. Moreover, it is computationally heavy leading to some limitations in real-time experiments.

1.8.3.4 PD control with computed feedforward

The idea of the PD control with computed feedforward is to use the full inverse dynamic model (similarly to APD controller) to compensate the effect of nonlinearity but within an offline-computation mode using the desired trajectory. One of the advantages of such controller is its simplicity and easy implementation exactly as a simple PD controller, due to all nonlinear dynamic terms are calculated before execution of the robot. Moreover, this strategy avoids the use of the actual measurement and estimated signals \((q(t), \dot{q}(t), \ddot{q}(t))\) which are often noisy and can reduce the control performance, but use instead the desired trajectory signals \((q_d(t), \dot{q}_d(t), \ddot{q}_d(t))\). There is no need to develop velocity and acceleration observers for such control strategy since all the dynamic computations depend on the desired trajectory. The joint space control equation can be formulated as follows [Santibañez and Kelly, 2001]:

\[
\Gamma(t) = M(q_d)\ddot{q}_d + C(q_d, \dot{q}_d)\dot{q}_d + G(q_d) + K_p e(t) + K_d \dot{e}(t) \tag{1.26}
\]

In [Natal et al., 2012], the computed feedforward term is a combination of Cartesian and joint dynamics (dual-space) enclosed within a Cartesian PID controller. The developed dual-space feedforward PID controller was implemented on a redundantly actuated parallel manipulator and tested for high accelerations. Good tracking performance of the proposed controller was validated for high-speed pick-and-place motions compared to the simple Cartesian PID controller.
1.8. OVERVIEW OF MOTION CONTROL SOLUTIONS FOR PARALLEL MANIPULATORS

1.8.3.5 Dynamic adaptive control

Dynamic adaptive control relies on the online adaptation of the system dynamic parameters providing an estimation of the uncertain, time-varying and unknown parameters. Back to 1989, a tutorial on the existing adaptive control strategies for serial rigid manipulators was reported in [Ortega and Spong, 1989] showing the global convergence of the error using the adaption law in the controller, while ensuring boundedness of the estimated parameter. The parameter adaptation law may be driven by the parameter estimation error or the output error (position error).

- Adaptive FeedForward Control:

For parallel manipulators, the Adaptive FeedForward Control (AFFC) equation in joint space formulation can be written as follows:

\[
\Gamma(t) = Y(q_d, \dot{q}_d, \ddot{q}_d)\hat{\Phi} + \Gamma_{FB}
\]

(1.27)

where \(\hat{\Phi}\) is the parameter estimated vector, \(Y(q_d, \dot{q}_d, \ddot{q}_d)\) is the regressor matrix fed with the desired trajectory, and \(\Gamma_{FB}\) is any feedback controller. The general block diagram of the AFFC algorithm is depicted in Figure 1.34. Mostly, the parameter update law is a function of the output error.

![Figure 1.34 – Block diagram of the general adaptive feedforward controller in joint space.](image-url)

Several AFFC algorithms have been developed for parallel kinematic manipulators in the literature. The experimental results ensured the boundedness and convergence of the
estimated parameters to the real values using the adequate adaptation law. It has been validated that the proposed adaptive control laws has better tracking performances than the conventional controllers and more robustness towards parameters variation.

One can mention from those AFFC strategies: PD with adaptive feedforward control [Honegger et al., 1997], dual-space adaptive feedforward control [Natal et al., 2015], Desired Compensation Adaptation Law [Bennehar et al., 2016], Robust Integral of the Sign of the Error (RISE)-based adaptive feedforward control [Bennehar et al., 2018], augmented $\mathcal{L}_1$ adaptive control with adaptive feedforward [Bennehar et al., 2015b], and adaptive terminal sliding mode control [Bennehar et al., 2017].

- **Other Dynamic Adaptive Control:**

  A nonlinear adaptive controller has been developed in task space for the trajectory tracking of a 2-DOF redundantly actuated parallel manipulator [Shang and Cong, 2010]. Experimental results show that the adaptive dynamic controller is more performant than the APD controller especially with dramatic changes of the dynamics in acceleration and deceleration processes. An additional adaptive friction compensation term enhanced the global performance in both low- and high- speed motions. To estimate the system parameters, the gradient descent algorithm is used thanks to its simplicity and easy implementation in real-time experiments. One more nonlinear adaptive dual-mode controller is proposed in [Natal et al., 2016] for the control of a 2-DOF parallel manipulator. The used adaptation law in dual-mode generates continuous control signals and limits the values of the estimated parameters. Different articular velocity observes have been developed for this controller showing better performances than a simple PD controller.

**1.8.3.6 Control with time-varying feedback gains**

  Control strategies with time-varying feedback gains arise from the advantages of using nonlinear feedback gains instead of fixed ones. Indeed, feedback loops with constant gains may have limited performances for high accelerations as well as limited tuning capabilities. Moreover, they don't take into consideration the dynamic change of operating conditions which makes them more sensitive to these changes. In a similar manner of
the NPD controller introduced hereinbefore, several dynamic control schemes of parallel manipulators have been enhanced in the literature.

An Augmented Nonlinear PD (ANPD) controller was proposed in [Shang et al., 2009] based on the conventional APD controller and the replacement of the linear PD control by nonlinear one. The stability analysis of such controller proved that it guarantees asymptotic convergence of both the tracking error and the error rate. The experimental results on a 2-DOFs redundantly actuated parallel robot show that the ANPD controller may realize higher-speed and higher-accuracy trajectory tracking compared to the conventional APD controller. The same approach was considered to improve the conventional CT controller in [Shang and Cong, 2009]. The developed Nonlinear CT (NCT) controller inherits merits from the CT controller, such as simple structure and clear physical meaning of each control parameter. Also it owns the good performances of the NPD algorithm in elimination of the nonlinear factors such as the modeling error and the nonlinear friction. The superiority of the proposed NCT controller in terms of accuracy and high-speed motion was validated through real-time experiments conducted on a 2-DOF redundantly actuated parallel manipulator.

Furthermore, the DCAL controller was revised in [Bennehar et al., 2014, 2016] by replacing the linear PD control term with a nonlinear one. Experiments conducted on a 3-DOF redundantly actuated PKM shows that the proposed controller outperforms the original one in terms of tracking performance while reducing the control effort.

1.8.3.7 Sliding mode control

As discussed before, SMC approach is a robust control strategy able to guarantee the finite time convergence of the sliding surface to the origin even with presence of disturbances and uncertainties. For the uncertain nature of parallel manipulators, SMC-based algorithms could be good candidates for the motion control problem.

In [Huang et al., 2004], a SMC approach has been proposed based on the full knowledge of the 6-DOF Stewart platform dynamics given that the overall system parameters are subjected to uncertainties. The stability analysis based on the Lyapunov theory confirmed the
finite-time convergence of the sliding surface to the origin, and the experimental results proved the effectiveness of the control design.

Besides, a cascaded-control algorithm based on SMC being in the outer loop was proposed in [Guo et al., 2008] to realize the trajectory tracking control of a hydraulically driven 6-DOFs parallel robot manipulator. The cascaded design was proposed to let the controller takes into account not only the mechanical dynamics but also the hydraulic dynamics of the manipulator. Satisfied position tracking behavior of the proposed controller has been shown through real-time experiments compared to a P controller with feedforward compensation.

An enhanced SMC was proposed in [Kim and Lee, 1998] for the real-time control of the 6-DOFs Stewart platform. The augmented proposed sliding surface and the added perturbation estimator compensated effectively for the nonlinear dynamics which was considered partially unknown. The sign function was treated with a continuous approximation to avoid the resulted chattering from the hard switching. Experimental results confirmed that the proposed SMC allowed to design a simple high-performance tracking control system for the Stewart PKM under high payloads and large disturbance conditions. The same controller was implemented on another 2-DOF parallel manipulator confirming again its effectiveness and good performance [Kim et al., 1998].

Another robust SMC approach with an active disturbance compensation has been proposed in [Singh and Santhakumar, 2015] for the trajectory tracking control of a 3-DOF vertical planar PKM in the presence of parameter uncertainties. Disturbance vector compromises dynamic parameter variations, frictional effects, and other unmodelled phenomena. The efficiency and robustness of the proposed controller were proven by numerical simulations and real-time experiments in the presence of the aforementioned disturbances.

In the previous SMC-based controllers, the dynamics of a PKM were partially or fully included within the closed-loop control assuming that the system parameters are known and subjected to uncertainties. Some of the previous controllers compensated for those accommodated uncertainties by designing disturbance observers. In [Bennehar et al., 2017], the uncertainties resulting from parameter variations were treated by an adaptive dynamic term that updates the values of the parameters depending on the operating con-
1.8. OVERVIEW OF MOTION CONTROL SOLUTIONS FOR PARALLEL MANIPULATORS

ditions. Real-time experiments show that the proposed adaptive terminal SMC is more performant than the standard terminal SMC in terms of precision and robustness towards parameter variations (such as handled payload).

Furthermore, a fuzzy SMC algorithm has been proposed for the trajectory tracking problem of a 4-DOF parallel robot [Qi et al., 2007]. The fuzzy logic system was proposed to replace the constant switching control gain avoiding the hard chattering that results from this term. Numerical simulations demonstrated a great reduction in the chattering with good tracking performance and robustness towards parameter uncertainties and external disturbances. Also in [Xu et al., 2018], a fuzzy SMC approach was designed based on a fuzzy neural network control theory. Numerical simulation results demonstrated the effectiveness of the proposed method.

1.8.3.8 Other dynamic control approaches

Other dynamic control approaches have been proposed in the literature worth to be mentioned. A predictive functional control strategy based on a simplified dynamic model of a 4-DOF parallel robot is proposed for the trajectory tracking problem within complex machining task trajectories [Vivas and Poignet, 2005]. Experimental results have shown that predictive functional control has the best performance compared to other control strategies, such as a classical PID and a CT control.

The robust $H_{\infty}$ controller has been tested experimentally on the 3-DOF Delta parallel robot in [Rachedi et al., 2015]. The control was designed by the mixed sensitivity approach taking into account both the sensitivity function matrix and the complementary sensitivity function. Experimental results show that $H_{\infty}$ controller outperforms the classical PID control at high dynamic operating conditions.

It has been shown in [Mueller, 2009] that in the presence of kinematic uncertainties in redundantly actuated PKMs, the internal prestress becomes a serious problem leading to antagonistic control forces or interference with the environment. The paper proposed to deal with those parasitic feedback forces by the control design, and two amended versions of augmented PD and CT controllers were developed for that purpose.
A compliance error compensation technique has been proposed for over-constrained parallel manipulators in [Klimchik et al., 2013]. The proposition takes into account the effect of the nonlinear stiffness coming from the non-perfect geometric model due to manufacturing errors. A comparison study confirmed that the errors to be compensated are highly dependant on the workpiece location.

1.9 How can we improve the performance of PKMs from a control point of view?

One can conclude from the literature review of control strategies of PKMs that achieving a performant controller depends on two main factors: the controller itself and the dynamic model of the manipulator. Thus, from a control point of view, the answer on how can we improve the dynamic performance of parallel manipulators is two-folded:

1. **Modification of the control strategy**: choosing a robust controller and trying to enhance it aiming at better robustness and accuracy could play an important role in getting better dynamic performances of parallel manipulators. Dynamic errors may be generated from the lack of robustness in the feedback controller against noisy measurements, friction, disturbances, and parameter variations. In particular, when the Cartesian measurements are not available, robust controllers become a valuable need to compensate for the sensors’ errors and the geometric model errors.

2. **Modification of the dynamic model**: improving the dynamic model may lead to high performances thanks to the model-based control strategies that compensate for the structured nonlinearities, as well as for the parameter variations for the case of dynamic adaptive schemes. Some sources of dynamic errors can be mentioned as follows: motor drivers, actuators dynamics, transmission system, and friction in the articulations, etc. The dynamic performance of parallel manipulators can be enhanced by proposing new formulations of the dynamic model, that take into account the aforementioned aspects, and enclosing it in the closed-loop control algorithms.
Following the aforementioned strategies, one can achieve high dynamic performances of parallel manipulators at different operating conditions.

1.10 Objectives of the thesis

The objectives of this thesis are to look for the necessary control tools to improve the dynamic performance of parallel robots in terms of motion speed, precision, and robustness. In this framework, two strategies can be considered in order to achieve the goal of the thesis: i) designing robust control solutions, ii) compensating for the errors coming from actuators dynamics, friction in the articulations, etc.

The proposed control solutions will be validated through real-time experiments on different available PKM prototypes. Real-time experiments will be performed in different operating conditions (nominal case, robustness towards disturbances and uncertainties, change of operating conditions, etc.) to show the effectiveness and robustness of the proposed control solutions in terms of global performances of the parallel robots.

1.11 Main contributions of the thesis

Two general guidelines were adopted in order to improve the dynamic performance of parallel manipulators: i) improve some robust control strategies, ii) improve the dynamic model to be used with dynamic control strategies. The main contributions can be listed as follows:

- Contribution 1: A new time-varying feedback RISE control
  A new time-varying feedback Robust Integral of the Sign of the Error (RISE) control strategy was developed for parallel manipulators. This proposed control takes the advantages of the nonlinear feedback gains and the robustness of the RISE controller. Some static feedback gains in the original RISE controller were replaced by nonlinear feedback ones aiming at more robustness towards disturbances, dynamic changes, and uncertainties. The new proposed controller was studied in the Lyapunov stability sense showing that the tracking error asymptotically converges
to zero with time. The relevance of this proposed controller have been validated experimentally on a parallel manipulator prototype.

- **Contribution 2: A novel model-based super-twisting sliding mode control**
  A novel model-based super-twisting sliding mode control was proposed as an extension of the original second order super-twisting algorithm. The control structure formed of a dynamic feedforward term, a feedback super-twisting control, and a stabilizing feedback term is adequate for parallel manipulator control in real-time framework. This formulation is less sensitive to noise measurements that can deteriorate the performance and stimulate the chattering effect. The stability analysis of the proposed control strategy was included ensuring a local asymptotic convergence of the tracking error and a finite-time convergence of the sliding variable. Experimental results on different parallel manipulator platforms showed an improved dynamic performance and more robustness towards disturbances and dynamic changes.

- **Contribution 3: Actuator and friction dynamics formulation in control of PKMs**
  In the framework of improving the dynamic model, an actuator and friction dynamics formulation was proposed being useful for model-based control strategies. The main idea was to include more dynamics to the enclosed model in the closed-loop control. This can boost up the dynamic performance and compensate for more percentage of existing nonlinearities. A PD control with computed feedforward incorporating the actuator and friction dynamics was suggested in order to test the formulated model. Moreover, the stability analysis of the proposed control has been investigated in the Lyapunov sense showing a global asymptotic convergence. The conducted experiments on a real parallel robot prototype showed effectiveness of the proposed dynamic formulation in terms of precision and robustness towards changes of operating conditions.
1.12 Conclusion

As we have seen, parallel mechanisms are serving a wide range of applications nowadays thanks to their significant advantages compared to their serial counterparts. PKMs, having their actuators at the fixed-base only, exhibit high-speed capabilities thanks to their lightweight moving parts. High stiffness and dynamic performance can be achieved simultaneously by PKMs due to their closed kinematic chains leading to high accuracies. Moreover, higher payload/weight ratios can be handled by parallel manipulators compared to serial ones. However, PKMs still suffer from serious limitations regarding their small workspace, abundant singularities and complex mechanism that need more improvements.

The long history and wide range of applications have been addressed in this chapter. The importance of advanced control strategies to be implemented on PKMs has been shown. Control of PKMs is considered challenging due to their highly nonlinear dynamics that increase considerably at high-speed motions, abundant uncertainties, time-varying parameters, external disturbances, and actuation redundancy.

This chapter provided state of the art on the existing control strategies for parallel manipulators classifying them into two categories: kinematic and dynamic control. Kinematic control deals with each axis of the parallel manipulator independently without considering the dynamics in the controller, while dynamic control relies mainly on a part of the dynamics or the full structured dynamics.

Unlike kinematic control, dynamic control compensates for the abundant nonlinearities enhancing the global dynamic performance of the parallel manipulator, especially at high dynamic operating conditions. One family of the dynamic control approaches, dynamic adaptive controllers, provides an online estimation of the system parameters and feeds those parameters again to the controller.

As a conclusion, one of two options can be followed in order to improve the dynamic performance of parallel manipulators. First, one can develop robust control strategies dealing with disturbances and uncertainties. Second, one can enhance the inverse dy-
The dynamic model by incorporating some sources of error such as motor drivers, actuators dynamics, transmission system, and friction in the articulations, etc.

The objectives of this thesis have been introduced as improving the dynamic performance of PKMs from a control point of view in terms of motion speed, robustness, and precision concerning two aspects: i) enhancing some existing robust control strategies, ii) considering more dynamic terms within the closed-loop control. The accomplished contributions of this thesis were listed at the end of this chapter.

The next chapters of the thesis describe the parallel robot prototypes that will be testbeds for validating the proposed controllers. Then, it explains the proposed control solutions within two classifications of contributions: improved robust control and enhanced dynamic model. Finally, the experimental validation of the proposed control solutions will be demonstrated and discussed.
2
Description and modeling of PKM prototypes

Contents

2.1 Introduction .................................................. 69
2.2 Non-redundant parallel robot prototypes ..................... 70
2.3 Redundant parallel robot prototype .......................... 87
2.4 Conclusion ..................................................... 96

2.1 Introduction

This chapter provides descriptions of the experimental prototypes that are used during this thesis for the validation of the proposed control solutions in a real-time framework. The existing prototypes are grouped into two different categories: non-redundant and redundant parallel manipulators. For each platform, a general description of the mechanical structure is presented, the kinematics are briefly addressed, and the dynamic model is mathematically explained and established.
2.2 Non-redundant parallel robot prototypes

Two platforms are presented in this section: the Delta robot and a Delta-like robot which is called Veloce. Both platforms can perform pick-and-place applications autonomously such as the packaging process for industrial products. As mostly known about pick-and-place applications, end users look for high-speed robots featuring a good level of accuracy which can be satisfied by our two platforms via designing robust and performant controllers.

2.2.1 Delta robot: a 3-DOF non-redundant PKM

2.2.1.1 Description of Delta PKM

Delta robot has been invented by prof. Reymond Clavel at Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland [Clavel, 1990]. It is mainly used for industrial applications that need pick-and-place cycle motions. Its smart design enables it to operate at

Figure 2.1 – A schematic view of Delta parallel robot including 1: Fixed-base, 2: Actuator, 3: Rear-arm, 4: Forearm, 5: Traveling-plate, 6: End-effector
very high accelerations thanks to the minimized mass of the mechanical parts which are
supposed to be in motion. Figure 2.1 shows a kinematic illustration of the Delta robot and
its main components.

The overall structure is composed of three actuators that are integral to a fixed-base,
three kinematic chains and one movable platform that is the traveling-plate. The three
kinematic chains form passive links between the actuators and the moving platform. The
shaft of each motor is connected to an extremity of a rear-arm supposed to make rota-
tion through a revolute joint. The second extremity of the rear-arm is linked to two paral-
lel rods through ball-and-socket passive joints. The parallel rods are then mounted to the
traveling-plate from their other side through the same said joints. The traveling-plate holds
a small end-effector that picks and places objects through an electric magnet. The syn-
chronized control of the three arms allows the traveling-plate to manipulate within three
basic translational DOFs (x, y and z) conserving its parallelism property with respect to the
fixed-base. The robot is considered a non-redundant PKM because the number of actu-
ators is equal to the number of the output DOFs.

2.2.1.2 Kinematics of Delta PKM

Consider the 3-dimensional coordinate vector \( \mathbf{X} = [x, y, z]^{T} \) as a representation of the
pose of the end-effector in the reference frame attached to the fixed base. Another 3-
dimensional coordinate vector \( \mathbf{q} = [q_1, q_2, q_3]^{T} \) represents the formed angles by the ac-
tuated joints.

The distribution of the needed geometric points is shown in Figure 2.2. Let \( O \) be
the center of the circle passing through all the actuated joints that are represented by
points \( A_i \) for \( i = 1, 2, 3 \) (see also Figure 2.3). The basic reference frame attached to \( O \) is
\( R = \{O, e_x, e_y, e_z\} \), where \( e_x, e_y, e_z \) are the corresponding unit vectors.

Let \( B_i \) and \( C_i \) be two virtual points located at the midpoints of each two ball-and-socket
joints connecting one rear-arm to one forearm and one forearm to the traveling-plate re-
spectively. This consideration can be done because the orientation of the traveling-plate
never changes when it moves in the workspace of the robot [Krut et al., 2006]. An auxiliary
frame \( R_i = \{A_i, u_i, v_i, z_i\} \) is attached to each actuated joint such that \( \mathbf{z}_i = e_z \) and \( \mathbf{u}_i \) is
normal to the circle crossing the actuated joints at $A_i$ directed away from the circle. $q_i$ is defined as an angle between $\vec{u}_i$ and $\overrightarrow{A_iB_i}$ starting from $\vec{u}_i$ in the vertical plane $(u_iz_i)$ as shown in Figure 2.2.

Let us consider a small circle passing through all $C_i$’s of center $O_t$, and attach to $O_t$ a local frame of the traveling-plate $R_t = \{O_t, x_t, y_t, z_t\}$.

Let $E = [x, y, z]^T$ be the pose of the end-effector with respect to the reference frame $R$. Then, a translation $T_h$, where $h$ is length of the end-effector, is performed to obtain the position vector $O_t = [x, y, z + h]^T$ (see Figure 2.4). Point $C_i$ is lying always in the horizontal plane $(x_1y_1)$ with a fixed angle $\alpha_i = (i-1)\frac{2\pi}{3}$ for $i = 1, 2, 3$ from $\vec{x}_t$ as shown clearly in Figure 2.3. Then $C_i$ is given as follows:

$$C_i = O_t + [r_t \cos(\alpha_i), r_t \sin(\alpha_i), 0]^T$$ \hspace{1cm} (2.1)

where $r_t$ is the radius of the circumscribed circle of the traveling-plate. The actuated joints $A_i$ are represented in the Cartesian reference frame $R$ as follows (see Figure 2.3):

$$A_i = [r_b \cos(\alpha_i), r_b \sin(\alpha_i), 0]^T$$ \hspace{1cm} (2.2)

---

Figure 2.2 – Distribution of the geometric points and frames on Delta PKM

Figure 2.3 – Top view of the geometrics of Delta PKM
where $r_b$ is the radius of the circumscribed circle of the fixed-base. In order to get the position vector of $B_i$ in the reference frame $R_i$, let us first introduce the position vector $R^1B_i$ with respect to the auxiliary frame $R_i$ as follows:

$$R^1B_i = [u_{B_i}, v_{B_i}, z_{B_i}]^T = [L_i \cos(q_i), 0, L_i \sin(q_i)]^T$$  \hspace{1cm} (2.3)

As common for all parallel robots, their arms are rigid enough to have always fixed lengths. Then, the rigidity of the rear-arms and forearms of Delta robot gives the following equalities respectively:

$$\|A_iB_i\| = L_i$$  \hspace{1cm} (2.4)

$$\|B_iC_i\| = l_i$$  \hspace{1cm} (2.5)

**Inverse kinematic Model of Delta PKM**

The idea of the Inverse Kinematic Model (IKM) is to find the corresponding joint position vector $q$ for a given Cartesian position vector $X$. Developing (2.4) and (2.5) in the
auxiliary frame $R_i$ leads to a system of two equations as follows:

$$u_{B_i}^2 + z_{B_i}^2 = L_i^2 \quad (2.6)$$

$$(u_{C_i} - u_{B_i})^2 + v_{C_i}^2 + (z_{C_i} - z_{B_i})^2 = l_i^2 \quad (2.7)$$

On one hand, the motion of a rear-arm is described by a circle of center $A_i$ and radius $L_i$ represented by equation (2.6). On the other hand, (2.7) describes the motion of a forearm as a sphere of center $C_i$ and radius $l_i$. Solving the aforementioned two equations for $u_{B_i}$ and $z_{B_i}$ in the frame $R_i$ gives the intersection point $B_i$ between the circle of each rear-arm and the sphere of each forearm.

Using (2.6) and (2.7), one can find the coordinates of $B_i$ as function of the point $C_i$, $u_{B_i} = f_1(R_iC_i)$ and $z_{B_i} = f_2(R_iC_i)$, respecting the accessible geometric workspace of the robot.

Indeed, the frame $R_i$ is obtained after performing a rotation on the reference frame $R$ about its $z$-axis by angle $\alpha_i$ and then a translation $T_{rb}$. $C_i$ is previously given in the reference frame $R$ as function of the Cartesian position vector of the end-effector in (2.1). Then, the coordinates of $C_i$ in the frame $R_i$ can be computed using the transformation matrix as follows:

$$\begin{pmatrix} R_iC_i \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\alpha_i) & -\sin(\alpha_i) & 0 & T_{rb}\cos(\alpha_i) \\ \sin(\alpha_i) & \cos(\alpha_i) & 0 & T_{rb}\sin(\alpha_i) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C_i \\ 1 \end{pmatrix} \quad (2.8)$$

Thus, the coordinates of the four actuated joints representing the inverse kinematic solution can be obtained as follows:

$$q_i = \text{atan2}(z_{B_i}, u_{B_i}) \quad (2.9)$$

**Forward kinematic Model of Delta PKM**

The Forward Kinematic Model (FKM) provides the Cartesian vector position of the end-effector $X$ given the joint vector position $q$. 
2.2. NON-REDUNDANT PARALLEL ROBOT PROTOTYPES

For each kinematic chain, the coordinates of \( B_i \) in the frame \( R_i (R_i B_i) \) can be calculated from the given \( q_i \) using (2.3). The coordinates of \( B_i \) in the reference frame \( R \) are deduced based on the transformation matrix used in (2.8) as follows:

\[
\begin{pmatrix}
B_i \\
1
\end{pmatrix} = \begin{pmatrix}
\cos(\alpha_i) & -\sin(\alpha_i) & 0 & r_b \cos(\alpha_i) \\
\sin(\alpha_i) & \cos(\alpha_i) & 0 & r_b \sin(\alpha_i) \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}^{-1}
\begin{pmatrix}
R_i B_i \\
1
\end{pmatrix}
\]  

\[ B_i = A_i + [L \cos(q_i) \cos(\alpha_i), L \cos(q_i) \sin(\alpha_i), L \sin(q_i)]^T \]

\[ B_i = A_i + A_i B_i \]

Now, developing (2.5) in the reference frame \( R \) leads to the following system of equations:

\[
(x_{C_i} - x_{B_i})^2 + (y_{C_i} - y_{B_i})^2 + (z_{C_i} - z_{B_i})^2 = l_i^2 \quad \forall \quad i = 1, 2, 3
\]  

(2.11)

Inserting (2.1) in (2.11) leads to another system of three equations and three unknowns, \( x_t, y_t \) and \( z_t \), the Cartesian coordinates of the moving platform of center \( O_t \), as follows:

\[
(x_t + r_t \cos(\alpha_i) - x_{B_i})^2 + (y_t + r_t \sin(\alpha_i) - y_{B_i})^2 + (z_t - z_{B_i})^2 = l_i^2 \quad \forall \quad i = 1, 2, 3
\]  

(2.12)

The numerical solution of (2.12) respecting the mechanical constraints of the robot gives the coordinates of \( O_t \) in the reference frame \( R \) which is the intersection point of three virtual spheres of center \( B'_i \) and radius \( |B'_i O_t| = l_i \) (see Figure 2.4). Therefore, the forward kinematic solution is deduced as follows:

\[
x = x_t \quad y = y_t \quad z = z_t - h
\]  

(2.13)

2.2.1.3 Differential Kinematics of Delta PKM

The differential kinematic model of parallel robots provides a relation between the Cartesian velocity vector \( \dot{X} \) and the joint velocity vector \( \dot{q} \) using the Jacobian matrix \( J(q, X) \). Then, one can formulate the Jacobian matrix by differentiating with respect to time the kinematic relationship between \( X \) and \( q \) in (2.11).
Applying the time derivative to (2.11) in the reference frame \( R \) leads us to the following equality:

\[
(x_{Ci} - x_{Bi}) \dot{x}_{Ci} + (y_{Ci} - y_{Bi}) \dot{y}_{Ci} + (z_{Ci} - z_{Bi}) \dot{z}_{Ci}
= (x_{Ci} - x_{Bi}) \dot{x}_{Bi} + (y_{Ci} - y_{Bi}) \dot{y}_{Bi} + (z_{Ci} - z_{Bi}) \dot{z}_{Bi}
\]

(2.14)

From (2.1), one can conclude that point \( C_i \) and the end-effector \( E \) have the same Cartesian velocity which means \( \dot{X}_{C_i} = \dot{X} \). Moreover, the Cartesian velocity of \( B_i \) can be derived from (2.10) as follows:

\[
\dot{X}_{B_i} = t_i \dot{q}_i
\]

(2.15)

where \( t_i \) is the tangent vector at point \( B_i \) to the circle of the rear-arm, shown in Figure 2.4, given as follows:

\[
t_i = [-L \sin(q_i) \cos(\alpha_i), -L \sin(q_i) \sin(\alpha_i), L \cos(q_i)]^T
\]

(2.16)

Therefore, (2.14) can be arranged and rewritten in the form below:

\[
J_x \dot{X} = J_q \dot{q}
\]

(2.17)

where \( J_q \) and \( J_x \) are given as follows:

\[
J_q = \text{diag}\{t_1^T B_1 C_1, t_2^T B_2 C_2, t_3^T B_3 C_3\}
\]

(2.18)

\[
J_x = [B_1 C_1^T, B_2 C_2^T, B_3 C_3^T]^T
\]

(2.19)

Finally, the Jacobian matrix is computed as follows:

\[
J = J_x^{-1} J_q
\]

(2.20)

It is worth to say that in the case of non-redundant parallel manipulators, such as Delta robot, the inverse of \( J_x \) always exists as long as the robot follows trajectories away from singularities. The differential kinematic model of Delta robot is then given by the following equations:

\[
\dot{X} = J \dot{q}
\]

(2.21)

\[
\dot{q} = J_m \dot{X} = J^{-1} \dot{X}
\]

(2.22)
2.2. NON-REDUNDANT PARALLEL ROBOT PROTOTYPES

2.2.1.4 Dynamics of Delta PKM

The dynamic model of Delta robot is established in this section, as in [Bennehar et al., 2018], based on the virtual work principle described in [Codourey, 1998]. As common for Delta-like PKMs, two assumptions are considered for a simplification purpose as follows:

Assumption 1. Both dry and viscous frictions in all passive and active joints are neglected.

Assumption 2. The forearms’ mass is split up into two point-masses, the first one is added to the mass of the rear-arms while the second is considered with the mass of the traveling-plate (see Figure 2.5).

Looking for the dynamics of the traveling-plate, one can define two kinds of forces acting on it: the gravitational force $G_{tp} \in \mathbb{R}^3$ and the inertial force $F_{tp} \in \mathbb{R}^3$.

Back to Assumption 2, the total mass of the traveling-plate including the half-masses of the forearms can be calculated as follows:

$$m_{tp} = m_p + \frac{3}{2}m_f$$  \hspace{1cm} (2.23)

where $m_p$ is the own mass of the traveling-plate and $m_f$ is the mass of each forearm as shown in Figure 2.5.
CHAPTER 2. DESCRIPTION AND MODELING OF PKM PROTOTYPES

Then, the gravitational force acting on the traveling-plate can be expressed as follows:

\[ G_{tp} = -M_{tp} G \]  \hspace{1cm} (2.24)

where \( M_{tp} \in \mathbb{R}^{3 \times 3} \) is the diagonal mass matrix of the traveling-plate (\( M_{tp} = \text{diag}(m_{tp}, m_{tp}, m_{tp}) \)). \( G \in \mathbb{R}^3 \) is the gravity vector (\( G = [0 \ 0 \ g]^T \), being \( g = 9.81 \text{ m/s}^2 \) the gravity acceleration).

The inertial force acting on the traveling-plate arising from its acceleration is defined as follows:

\[ F_{tp} = M_{tp} \ddot{X} \]  \hspace{1cm} (2.25)

with \( \ddot{X} \in \mathbb{R}^3 \) denoting its acceleration vector.

The contributions of the gravitational and inertial forces to the actuator torques are evaluated using the Jacobian matrix as follows:

\[ \Gamma_{G_{tp}} = J^T G_{tp} \]  \hspace{1cm} (2.26)

\[ \Gamma_{F_{tp}} = J^T F_{tp} \]  \hspace{1cm} (2.27)

Besides, the dynamics from the actuators side includes the contributions of forces acting on the rear-arms. Here, we name three contributing torques: (i) the actuators input torque \( \Gamma \in \mathbb{R}^3 \), (ii) the effect of the rear-arms gravitational forces \( \Gamma_{G_{\text{arm}}} \in \mathbb{R}^3 \) and (iii) the inertial contribution due to rear-arms acceleration \( \Gamma_{\text{arm}} \in \mathbb{R}^3 \).

In order to quantify the contribution of the rear-arms gravitational forces, let us define the following diagonal matrix taking into consideration the statement of Assumption 2.

\[ M_{r} = \text{diag}(m_{\text{req}}, m_{\text{req}}, m_{\text{req}}) \]  \hspace{1cm} (2.28)

with

\[ m_{\text{req}} = m_{r} l_{rg} + L \frac{m_{r}}{2} \]  \hspace{1cm} (2.29)

where \( m_{r} \) is the mass of each rear-arm, \( l_{rg} \) is the distance from the axis of rotation of each rear-arm to its center of gravity, while \( L \) is the complete length of each rear-arm as
2.2. NON-REDUNDANT PARALLEL ROBOT PROTOTYPES

illustrated in Figure 2.5. Then, the torque produced by the gravitational forces of the rear-arms is given by:

\[ \Gamma_{G_{\text{arm}}} = -g M_{r} \cos(q) \]  \hspace{1cm} (2.30)

where \( \cos(q) \triangleq [\cos(q_1), \cos(q_2), \cos(q_3)]^T \).

The inertial contribution of the operating acceleration of the rear-arms can be defined as follows:

\[ \Gamma_{\text{arm}} = I_{\text{arm}} \ddot{q} \]  \hspace{1cm} (2.31)

where \( I_{\text{arm}} \in \mathbb{R}^{3 \times 3} \) is a diagonal inertia matrix including the inertia of the actuators, the rear-arms and the half-masses of the forearms with respect to the actuators’ rotation axes. \( \ddot{q} \in \mathbb{R}^3 \) is the acceleration vector in joint space.

After applying the virtual work principle, stating that the contribution of all non-inertial forces must be equal to the contribution of all inertial forces, one can formulate the inverse dynamic model as follows:

\[ \Gamma = I_{\text{arm}} \ddot{q} + J^T M_{tp} \ddot{X} + \Gamma_{G_{tp}} + \Gamma_{G_{\text{arm}}} \]  \hspace{1cm} (2.32)

By computing the first time derivative of (2.21), we obtain the relation between joint and Cartesian accelerations, expressed as follows:

\[ \ddot{X} = J \ddot{q} + \dot{J} \dot{q} \]  \hspace{1cm} (2.33)

where \( \dot{J} \) is the time derivative of \( J \).

Now, substituting (2.33) in (2.32) and rearranging the terms give the inverse dynamic model of Delta parallel robot in the joint space as follows:

\[ \Gamma(t) = M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) \]  \hspace{1cm} (2.34)

where \( M(q) = I_{\text{arm}} + J^T M_{tp} J \) is the total mass and inertia matrix of the robot, \( C(q, \dot{q}) = J^T M_{tp} \dot{J} \) is the Coriolis and centrifugal forces matrix, \( G(q) = -\Gamma_{G_{tp}} - \Gamma_{G_{\text{arm}}} \) is the gravitational forces vector, and \( \Gamma(t) \) is the control input vector. The main dynamic parameters of delta parallel robot are summarized in Table 2.1.
### Table 2.1 – The main dynamic parameters of Delta parallel robot.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$</td>
<td>Rear-arm length</td>
<td>240 mm</td>
</tr>
<tr>
<td>$l$</td>
<td>Forearm length</td>
<td>480 mm</td>
</tr>
<tr>
<td>$m_r$</td>
<td>Rear-arm mass</td>
<td>0.22 kg</td>
</tr>
<tr>
<td>$m_f$</td>
<td>Forearm mass</td>
<td>0.084 kg</td>
</tr>
<tr>
<td>$m_p$</td>
<td>Own traveling-plate mass</td>
<td>0.305 kg</td>
</tr>
<tr>
<td>$I_{\text{act}}$</td>
<td>Actuator inertia</td>
<td>$1.82 \times 10^{-3}$ kg.m$^2$</td>
</tr>
</tbody>
</table>

#### 2.2.2 VELOCE robot: a 4-DOF non-redundant PKM

#### 2.2.2.1 Description of VELOCE PKM

VELOCE robot, designed and fabricated at LIRMM, is a Delta-like parallel robot featuring one additional kinematic chain and one additional rotational degree-of-freedom. A CAD view of the fabricated VELOCE robot is illustrated in Figure 2.6. It is a non-redundant fully parallel manipulator having four identical kinematic chains where each one is considered a series of an actuator, a rear-arm, and a forearm. Each forearm comprises two parallel rods connected from one extremity to a rear-arm and from the other extremity to a traveling-plate through ball-and-socket passive joints the same as in Delta robot. Thus, the moving-platform maneuvers in three translation DOFs (x, y and z) and one rotational DOF ($\theta_z$) around z-axis perpendicular to the fixed-base in the main reference frame preserving without any inclination or orientation.

The innovative feature of the VELOCE robot lies mainly in its moving-platform which is made of two traveling-plates guided in translation relatively along a screw (along z-axis) holding by its end the end-effector as shown in Figure 2.6. Each traveling-plate is connected to two opposite side kinematic chains. This configuration transforms the relative distance between the two traveling-plates into the rotation of the end-effector [Company et al., 2013].
2.2. NON-REDUNDANT PARALLEL ROBOT PROTOTYPES

Figure 2.6 – A schematic view of VELOCE parallel robot including 1: Fixed-base, 2: Actuator, 3: Rear-arm, 4: Forearm, 5: Moving platform, 6: End-effector.

2.2.2.2 Kinematics of VELOCE PKM

The translational and rotational DOFs of VELOCE robot are represented in the 4-dimensional Cartesian coordinate vector $X = [x, y, z, \theta_2]^T$. The joint space position vector representing the values of actuated angles is given by the 4-dimensional vector $q = [q_1, q_2, q_3, q_4]^T$.

Considering similar notation and distribution of the geometrical points of Delta robot (Section 2.2.1.2), one can express with respect to the base frame $R$ the following parameters.

The angle of distribution of the actuators starting from $\vec{e}_x$ with anticlockwise direction:

$$\alpha_i = (i - 1) \frac{\pi}{2} \quad (2.35)$$

Vector $\vec{u}_i$ of the auxiliary frame $R_i$:

$$\vec{u}_i = [\cos(\alpha_i), \sin(\alpha_i), 0]^T \quad (2.36)$$
Point $A_i$ representing the position of each actuator:

$$A_i = r_b u_i$$  \hspace{1cm} (2.37)

Point $B_i$ where the joint between a rear-arm and a forearm takes place:

$$B_i = A_i + L [\cos(\alpha_i)\cos(q_i), \sin(\alpha_i)\cos(q_i), \sin(q_i)]^T$$  \hspace{1cm} (2.38)

Point $C_i$ where the joint between a forearm and one of the traveling-plates takes place:

$$C_i = E + r_t u_i + (h + \frac{p_i}{\pi} \theta_z) e_z$$  \hspace{1cm} (2.39)

where $r_b$ is the radius of the circle passing through the four actuators of center $O$, $L$ is the length of the rear-arm, $E = [x, y, z]^T$ is the position vector of the end-effector in reference $R$, $r_t$ is the radius of the circle circumscribed of a traveling-plate, $h$ is the geometric distance shown in Figure 2.7, $p_i = 0$ for $i = \{1, 3\}$ and $p_i = p$ for $i = \{2, 4\}$ with $p$ being the pitch of the helical joint which is the axial distance between the crests of adjacent threads of the screw. Considering a double start screw used in our prototype, the linear distance covered in one full round is two pitches ($2p/2\pi$).

**Inverse kinematic Model of VELOCE PKM**

Following the same manner used to calculate the IKM of Delta PKM in Section 2.2.1.2, one can compute the position vector of point $B_i$ in the auxiliary reference $R_i$, the intersection of the circle of the rear-arm and the sphere of the forearm, knowing the Cartesian coordinates of the end-effector.

Developing the equalities of rigidity of the rear-arms and forearms ,(2.4) and (2.5), in the auxiliary reference $R_i$ provides a relation between $^R_i B_i$ and $^R_i C_i$. Using the transformation matrix in (2.8), one can deduce the position vector $C_i$ in the reference $R$.

Thus, the inverse kinematic solution representing the four actuated joint angles is given as follows:

$$q_i = \arctan2(z_{B_i}, u_{B_i})$$  \hspace{1cm} (2.40)
2.2. NON-REDUNDANT PARALLEL ROBOT PROTOTYPES

![Diagram of a parallel robot prototype with geometric points and frames.]

Figure 2.7 – Illustrative view of the geometric points and frames considered on VELOCE PKM.

**Forward kinematic Model of VELOCE PKM**

Given the joint position vector $q_i$, one can obtain the position vector $B_i$ in the reference frame $R$ using (2.38) with $\alpha_i$ introduced in (2.35) and $i = 1,\ldots,4$.

Back to the equality that expresses the rigidity property of the forearms in Delta-like parallel robots (2.5), developing this equality in the reference frame $R$ leads to a system of four equations in terms of $C_i$ as follows:

$$
(x_{C_i} - x_{B_i})^2 + (y_{C_i} - y_{B_i})^2 + (z_{C_i} - z_{B_i})^2 = l_i^2 \quad \forall \quad i = 1,\ldots,4
$$

(2.41)

Substituting (2.39) in (2.41) leads to a new system of equations in terms of the Cartesian
position vector \( \mathbf{X} \) as follows:

\[
\begin{align*}
(x + r_i - x_{B_1})^2 + (y - y_{B_1})^2 + (z + h - z_{B_1})^2 &= l_1^2 \\
(x - x_{B_2})^2 + (y + r_i - y_{B_2})^2 + (z + h + \frac{p}{\pi} \theta_z - z_{B_2})^2 &= l_2^2 \\
(x - r_i - x_{B_3})^2 + (y - y_{B_3})^2 + (z + h - z_{B_3})^2 &= l_3^2 \\
(x - x_{B_4})^2 + (y - r_i - y_{B_4})^2 + (z + h + \frac{p}{\pi} \theta_z - z_{B_4})^2 &= l_4^2
\end{align*}
\]

where \( l_1 = l_2 = l_3 = l_4 = l \) are the lengths of the forearms. Solving the final obtained system (2.42) that consists of four equations and four unknowns \( x, y, z \) and \( \theta_z \) provides the forward kinematic solution of VELOCE robot which is represented by the intersection of four spheres of equations shown in (2.42).

### 2.2.2.3 Differential Kinematics of VELOCE PKM

Differentiating with respect to time the kinematic relationship of VELOCE parallel robot (2.41) gives us the differential kinematic relation between the Cartesian velocity vector \( \dot{\mathbf{X}} \) and the joint velocity vector \( \dot{\mathbf{q}} \).

To proceed in developing the Jacobian matrix, we address the time derivatives of the position vectors \( C_i \) and \( B_i \) in (2.39) and (2.38) respectively as follows:

\[
\begin{align*}
\dot{x}_{C_i} &= \dot{x} \\
\dot{y}_{C_i} &= \dot{y} \\
\dot{z}_{C_i} &= \dot{z} + \frac{p}{\pi} \dot{\theta}_z
\end{align*}
\]

\[
\dot{X}_{B_i} = t_i \ddot{q}_i \equiv \begin{cases} 
\dot{x}_{B_i} = (-L \cos(\alpha_i) \sin(q_i)) \ddot{q}_i \\
\dot{y}_{B_i} = (-L \sin(\alpha_i) \sin(q_i)) \ddot{q}_i \\
\dot{z}_{B_i} = (L \cos(q_i)) \ddot{q}_i
\end{cases}
\]

where \( t_i \) is the tangent vector at point \( B_i \) to the circle of the rear-arm.
2.2. NON-REDUNDANT PARALLEL ROBOT PROTOTYPES

Substituting (2.43) and (2.44) in the time derivative of (2.41) yields up to the differential kinematic relation as follows:

\[ \dot{X} = Jq \]  

(2.45)

\[ J = J_x^{-1} J_q \]  

(2.46)

\[ J_x = \begin{pmatrix} B_1 C_1^T (z_{C_1} - z_{B_1}) \frac{p_1}{\pi} \\ B_2 C_2^T (z_{C_2} - z_{B_2}) \frac{p_2}{\pi} \\ B_3 C_3^T (z_{C_3} - z_{B_3}) \frac{p_3}{\pi} \\ B_4 C_4^T (z_{C_4} - z_{B_4}) \frac{p_4}{\pi} \end{pmatrix} \]  

(2.47)

\[ J_q = \text{diag}\{t_1^T B_1 C_1, t_2^T B_2 C_2, t_3^T B_3 C_3, t_4^T B_4 C_4\} \]  

(2.48)

Similar to Delta PKM, VELOCE PKM is a non-redundant prototype that has \( J_x \) always invertible as long as the followed trajectory is free of singularities.

2.2.2.4 Dynamics of VELOCE PKM

The dynamic model of VELOCE robot and a Delta one have a lot of similarities, except few differences in VELOCE robot coming from the fourth kinematic chain and the additional rotational motion.

Considering Assumptions 1 and 2 of Delta-like PKMs, for simplification purposes, the dynamics of VELOCE robot can be classified according to the working space, either dynamics of Cartesian space or dynamics of joint space.

Regarding the dynamics of Cartesian space, it covers the forces acting on the traveling-plate such as the gravitational force \( G_{tp} \in \mathbb{R}^3 \) and the inertial force \( F_{tp} \in \mathbb{R}^4 \) expressed as follows:

\[ G_{tp} = -M_{tp} G \]  

(2.49)

\[ F_{tp} = M_{tp} \ddot{X} \]  

(2.50)
where $M_{tp} \in \mathbb{R}^{4 \times 4}$ is the total mass matrix of the moving platform including the half-masses of the forearms, $G = [0, 0, g, 0]^T$ is the gravity vector with $g = 9.81 \text{m/s}^2$ being the gravity acceleration, and $\ddot{X} \in \mathbb{R}^4$ represents the Cartesian acceleration vector. The contributions of the aforementioned forces to the actuator torques are computed using the Jacobian matrix as follows:

$$\Gamma_{G_{tp}} = J^T G_{tp} \quad (2.51)$$

$$\Gamma_{F_{tp}} = J^T F_{tp} \quad (2.52)$$

In the joint space, the dynamics include the actuator input torques $\Gamma \in \mathbb{R}^4$, the effect of the rear-arm gravitational forces $\Gamma_{G_{arm}} \in \mathbb{R}^4$ and the inertial contribution due to the rear-arm accelerations $\Gamma_{arm} \in \mathbb{R}^4$. The torque contribution coming from the gravitational forces of the rear-arms is given as follows:

$$\Gamma_{G_{arm}} = - g M_r \cos(q) \quad (2.53)$$

$$M_r = \text{diag}(m_{r_{eq}}, m_{r_{eq}}, m_{r_{eq}}, m_{r_{eq}}) \quad (2.54)$$

$$m_{r_{eq}} = m_r l_{rG} + L \frac{m_r}{2} \quad (2.55)$$

where $m_r$ is the mass of each rear-arm, $l_{rG}$ is the distance from the axis of rotation of each rear-arm to its center of gravity, $L$ is the complete length of each rear-arm as illustrated in Figure 2.5, and $\cos(q) \triangleq [\cos(q_1), \cos(q_2), \cos(q_3), \cos(q_4)]^T$.

Moreover, the torque contribution of the inertial forces coming from the acceleration of the rear-arms is calculated as follows:

$$\Gamma_{arm} = I_{arm} \ddot{q} \quad (2.56)$$

where $I_{arm} \in \mathbb{R}^{3 \times 3}$ is a diagonal inertia matrix including the inertia of the actuators, the inertia of the rear-arms and the inertia of the half-masses of the forearms with respect to the actuators’ rotation axes. $\ddot{q} \in \mathbb{R}^3$ is the acceleration vector in joint space.
Similarly to dynamics of Delta robot in Section 2.2.1.4, and after applying the virtual work principle, the inverse dynamic model of VELOCE PKM in joint space is given as follows:

\[ \Gamma(t) = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) \]  

(2.57)

where \( M(q) = I_{\text{arm}} + J^T M_{\text{tp}} J \) is the total mass and inertia matrix of the robot, \( C(q, \dot{q}) = J^T M_{\text{tp}} J \) is the Coriolis and centrifugal forces matrix, \( G(q) = -\Gamma_{G_{\text{tp}}} - \Gamma_{G_{\text{arm}}} \) is the gravitational forces vector, and \( \Gamma(t) \) is the control input vector. The main dynamic parameters of VELOCE parallel robot are summarized in Table 2.2.

### 2.3 Redundant parallel robot prototype

In this section, the SPIDER4 parallel robot is presented. This platform can run controlled machining processes of material-removal that make a desired deformation in the shape and size of raw materials. Machining devices of parallel structures are considered good mechanical solutions for the machining operations thanks to the high accuracy and stiffness provided by their closed kinematic chains.
2.3.1 SPIDER4 robot: a 5-DOF redundant PKM

2.3.1.1 Description of SPIDER4 PKM

SPIDER4 robot is a Delta-like redundantly actuated parallel manipulator designed and fabricated at LIRMM within the cooperation with TECNALIA, the research and innovation organisation located at Spain. The intention behind this platform is to perform machining operations with high dynamic performance, high precision and dexterity. It is accompanied with a tooling (table) on which the parts to be machined are fixed.

Figure 2.8 shows SPIDER4 PKM structure consists of a fixed-base holding four high torque actuators each linked to a rear-arm through a revolute joint. Two parallel rods forming a forearm are connected to each rear-arm as well as to the traveling-plate by the means of universal joints. The traveling-plate (also referred as the nacelle) is allowed to move within three translational axes \( x, y \) and \( z \) thanks to the parallel kinematic structure. Additional independent serial wrist mechanism (two motors) is attached to the nacelle offering two more rotational movements for the machining spindle around the axes of the motors \( M_1 \) and \( M_2 \) as illustrated in Figure 2.8. Thus, SPIDER4 robot is a 5-DOF redundant parallel manipulator with a degree of redundancy equal to one. It is worth to mention that the overall dimensions of SPIDER4 with the tooling are 4600 mm in length, 2500 mm in width and 2400 mm in height as illustrated in Figure 2.9.

In this thesis, we are concerned only with the parallel structure of SPIDER4 proposing control solutions for the trajectory tracking problem of the nacelle in the workspace. All the modeling coming in the sequel is based only on the parallel structure of the SPIDER4 robot.

2.3.1.2 Kinematics of SPIDER4 PKM

The kinematic model of SPIDER4 robot describes the relation between the actuated joint angles and the Cartesian position of the nacelle. As previously mentioned, in our modeling and control developments, we are concerned only with parallel structure and not aware of the spindle positioning.

Consider \( X = [x, y, z]^T \) as the Cartesian position vector of the nacelle center and \( q = \)
2.3. REDUNDANT PARALLEL ROBOT PROTOTYPE

Figure 2.8 – A schematic view of SPIDER4 parallel robot including ①: Fixed-base, ②: Actuator, ③: Rear-arm, ④: Forearm, ⑤: Nacelle, ⑥: Serial wrist mechanism, ⑦: Spindle.

Figure 2.9 – Overall dimensions (in mm) of the whole system, SPIDER4 and the tooling.
As the joint position vector representing the configuration of the actuated joints.

Let us define the reference frame $R = \{ O, e_x, e_y, e_z \}$ attached to center of the fixed-base and the auxiliary frame $R_i = \{ A_i, u_i, v_i, z_i \}$ attached to the point $A_i$ representing one of the actuators for $i = 1,..,4$ as illustrated in Figures 2.10 and 2.11.

The position vector of each actuator can be given in the reference frame $R$ as follows:

$$\mathbf{A}_i = r_b \mathbf{u}_i$$

(2.58)

where $\mathbf{u}_i = [\cos(\alpha_i), \sin(\alpha_i), 0]^T$, $\alpha_i = \frac{\pi}{4} - \frac{2i}{4} \pi$ and $r_b$ is the radius of the circle circumscribed of the actuator points $A_i$ (see Figure 2.10).

Then, the position vector of the point $B_i$ where the joint between a rear-arm and a forearm takes place can be given as follows (see Figure 2.11):

$$\mathbf{B}_i = \mathbf{A}_i + L[\cos(\alpha_i) \cos(q_i), \sin(\alpha_i) \cos(q_i), \sin(q_i)]^T$$

(2.59)

where $L$ is the length of a rear-arm.

The position vector of the point $C_i$ where the joint between a forearm and the nacelle takes place can be given in the reference frame $R$ as follows:

$$\mathbf{C}_i = \mathbf{X} + r_t \mathbf{u}_i$$

(2.60)

where $r_t$ is the length of a rear-arm.

**Inverse kinematic Model of SPIDER4 PKM**

Starting from a known Cartesian position vector $\mathbf{X}$, one can compute the joint position vector $\mathbf{q}$ using the inverse kinematic model. Similarly to Delta PKM, calculating the coordinates of point $B_i$ in the auxiliary frame $R_i$ leads to a solution of the inverse kinematic problem as follows:

$$q_i = \text{atan2}(z_{B_i}, u_{B_i})$$

(2.61)

$z_{B_i} \text{ and } u_{B_i}$ can be computed by developing the rigidity equalities of the rear-arms and the forearms, (2.4) and (2.5) respectively, in the auxiliary frame $R_i$. 

2.3. **REDUNDANT PARALLEL ROBOT PROTOTYPE**

Figure 2.10 – View of the geometrics of SPIDER4 PKM from the side of the fixed-base.

Two equations similar to (2.6) and (2.7) will be obtained describing respectively the

Figure 2.11 – View of the geometric parameters of one kinematic chain of SPIDER4 PKM.
motion of a rear-arm (circle of center $A_i$ and radius $L_i$) and forearm (sphere of center $C_i$ and radius $l_i$).

Solving the two obtained equations in terms of $z_{B_i}$ and $u_{B_i}$ give the coordinates of the point $B_i$ as an intersection of the rear-arm circle and the forearm sphere. Therefore, the inverse kinematic model is concluded by (2.61).

**Forward kinematic Model of SPIDER4 PKM**

The forward kinematic model of SPIDER4 PKM provides the Cartesian position vector of the nacelle starting from a known configuration of the four actuated joint angles.

One can develop the equality of rigidity of the forearm (2.5) in the reference frame $R$ obtaining the following system of equations:

$$(x_{C_i} - x_{B_i})^2 + (y_{C_i} - y_{B_i})^2 + (z_{C_i} - z_{B_i})^2 = l_i^2 \quad \forall \quad i = 1,..,4$$  \hspace{1cm} (2.62)

Substituting the position vector of the point $C_i$ (2.60) in (2.62) leads to a new system of equations as follows:

$$(x + r_t \cos(\alpha_i) - x_{B_i})^2 + (y + r_t \sin(\alpha_i) - y_{B_i})^2 + (z - z_{B_i})^2 = l_i^2 \quad \forall \quad i = 1,..,4$$  \hspace{1cm} (2.63)

For more simplification, following a technique of change of variables, (2.63) can be re-written as follows:

$$(x - x'_{B_i})^2 + (y - y'_{B_i})^2 + (z - z'_{B_i})^2 = l_i^2 \quad \forall \quad i = 1,..,4$$  \hspace{1cm} (2.64)

where $B'_i = B_i - r_t [\cos(\alpha_i), \sin(\alpha_i), 0]^T$ is the translation of point $B_i$ along the $u_i$-axis with coefficient $-r_t$. Therefore, the forward kinematic solution represents the intersection of four spheres of centers $B'_i$ and radii $l_i$.

**2.3.1.3 Differential Kinematics of SPIDER4 PKM**

Applying the time derivative to the kinematic relationship of SPIDER4 PKM (2.62) leads to a Jacobian matrix formulation, and thus establishing the differential kinematic model.
From (2.60), it is clear that the Cartesian velocity vector $\dot{X}_{Ci}$ and that of the center of the nacelle $\dot{X}$ are equal. Then, after differentiating with respect to time (2.62) and getting use of the velocity of the nacelle, one can establish the equation below:

$$\begin{bmatrix} B_1C_1^T, B_2C_2^T, B_3C_3^T, B_4C_4^T \end{bmatrix}^T \dot{X} = \begin{bmatrix} B_1C_1^T, B_2C_2^T, B_3C_3^T, B_4C_4^T \end{bmatrix}^T \dot{X}_{Bi} \quad (2.65)$$

The velocity vector $\dot{X}_{Bi}$ can be derived from (2.59) as follows:

$$\dot{X}_{Bi} = t_i \dot{q}_i \quad (2.66)$$

where $t_i$ is the tangent vector at point $B_i$ to the circle of the rear-arm given as follows:

$$t_i = [-L\sin(q_i)\cos(\alpha_i), -L\sin(q_i)\sin(\alpha_i), L\cos(q_i)]^T \quad (2.67)$$

Finally, the differential kinematic relationship between the Cartesian velocity vector and the joint velocity vector is formulated as follows:

$$\dot{X} = J \dot{q} \quad (2.68)$$

where $J$ is the Jacobian matrix given as follows:

$$J = J_x^+J_q \quad (2.69)$$

with $(.)^+$ denotes the pseudoinverse of a non-diagonal matrix. $J_x$ and $J_q$ can be stated as follows:

$$J_x = \begin{bmatrix} B_1C_1^T, B_2C_2^T, B_3C_3^T, B_4C_4^T \end{bmatrix}^T \quad (2.70)$$

$$J_q = \text{diag} \{t_1^TB_1C_1, t_2^TB_2C_2, t_3^TB_3C_3, t_4^TB_4C_4 \} \quad (2.71)$$

It is worth to note that using the pseudoinverse technique for solving redundancy looks good since it generates the minimum norm joint velocities, but still, the kinematic singularities are not avoided [Siciliano, 1990]. The pseudoinverse of the Jacobian matrix exists as long as our robot is far from singular postures within its operational workspace, which means that the Jacobian matrix doesn’t lose its rank.
2.3.1.4 Dynamics of SPIDER4 PKM

Considering the same Assumptions 1 and 2 of Delta robot, the inverse dynamic model of SPIDER4 PKM can be established based on the virtual work principle.

On the one hand, the traveling-plate dynamics can be described by Newton-Euler formulation as follows:

\[ M_{tp}(\ddot{X} - G) = 0 \]  
(2.72)

where \( M_{tp} = \text{diag}(m_{tp}, m_{tp}, m_{tp}) \) is the total mass matrix including the mass of the nacelle, the payload lifted by the nacelle (the two motors and spindle), and the half-masses of the forearms, \( \ddot{X} \in \mathbb{R}^3 \) is the Cartesian acceleration vector and \( G = [0, g, 0]^T \) represents the gravity vector with \( g = 9.81 \text{ m/s}^2 \) being the gravity acceleration. Then, the gravitational force acting on the traveling-plate can be given as follows:

\[ G_{tp} = -M_{tp}G \]  
(2.73)

while the inertial force arising from the acceleration of the traveling-plate is stated as follows:

\[ F_{tp} = M_{tp}\ddot{X} \]  
(2.74)

The above-mentioned forces are converted into torque contributions at the joint side using the Jacobian matrix as follows:

\[ \Gamma_{G_{tp}} = J^T G_{tp} \]  
(2.75)

\[ \Gamma_{F_{tp}} = J^T F_{tp} \]  
(2.76)

On the other hand, the dynamics of the rear-arms from the joint side comprise the torque generated from the actuators \( \Gamma \in \mathbb{R}^4 \), the torque contribution of the gravitational force acting on the rear-arms \( \Gamma_{G_{\text{arm}}} \in \mathbb{R}^4 \), and the inertial contribution due to the rear-arms’ acceleration \( \Gamma_{\text{arm}} \in \mathbb{R}^4 \).

For the case of SPIDER4 PKM, it is clear that the gravitational force acting on a rear-arm is not in the same plane of its rotational motion as shown in Figure 2.12. This is due to the horizontal orientation of SPIDER4 PKM and its inclination around z-axis with an angle \( \alpha \).
Then, with a simple geometric computation, one can compute the torque contribution of the gravitational effects as follows:

$$\Gamma_{G_{\text{arm}}} = -gM_{r\alpha}\cos(q)$$  \hspace{1cm} (2.77)

with

$$M_{r\alpha} = \begin{pmatrix}
    m_{\text{req}} \sin(\alpha_1) & 0 & 0 & 0 \\
    0 & m_{\text{req}} \sin(\alpha_2) & 0 & 0 \\
    0 & 0 & m_{\text{req}} \sin(\alpha_3) & 0 \\
    0 & 0 & 0 & m_{\text{req}} \sin(\alpha_4)
\end{pmatrix}$$  \hspace{1cm} (2.78)

$$m_{\text{req}} = m_r l_{rG} + L \frac{m_f}{2}$$  \hspace{1cm} (2.79)

where $m_r$ is the mass of each rear-arm, $l_{rG}$ is the distance from the axis of rotation of each rear-arm to its center of gravity, $L$ is the complete length of each rear-arm as illustrated in Figures 2.11 and 2.12, and $\cos(q) \triangleq [\cos(q_1), \cos(q_2), \cos(q_3), \cos(q_4)]^T$. 
Table 2.3 – The main dynamic parameters of SPIDER4 parallel robot.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$</td>
<td>Rear-arm length</td>
<td>535 mm</td>
</tr>
<tr>
<td>$l$</td>
<td>Forearm length</td>
<td>1100 mm</td>
</tr>
<tr>
<td>$m_r$</td>
<td>Rear-arm mass</td>
<td>17.6 kg</td>
</tr>
<tr>
<td>$m_f$</td>
<td>Forearm mass</td>
<td>4.64 kg</td>
</tr>
<tr>
<td>$m_{tp}$</td>
<td>Total traveling-plate mass</td>
<td>51.54 kg</td>
</tr>
<tr>
<td>$I_{arm}$</td>
<td>Rear-arm inertia</td>
<td>1.69 kg.m$^2$</td>
</tr>
<tr>
<td>$I_{act}$</td>
<td>Actuator inertia</td>
<td>$2.23 \times 10^{-3}$ kg.m$^2$</td>
</tr>
</tbody>
</table>

The torque contribution of the inertial force acting on a rear-arm can be defined as follows:

$$\Gamma_{\text{arm}} = I_{\text{arm}} \ddot{q}$$  \hspace{1cm} (2.80)

where $I_{\text{arm}} \in \mathbb{R}^{4 \times 4}$ is a diagonal inertia matrix including the inertia of the actuators, the rear-arms and the half-masses of the forearms with respect to the actuators’ rotation axes. $\ddot{q} \in \mathbb{R}^4$ is the acceleration vector in joint space.

Finally, the inverse dynamic model of SPIDER4 PKM can be formulated using the virtual work principle as follows:

$$\Gamma(t) = M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q)$$  \hspace{1cm} (2.81)

where $M(q) = I_{\text{arm}} + J^T M_{\text{tp}} J$ is the total mass and inertia matrix of the robot, $C(q, \dot{q}) = J^T M_{\text{tp}} J$ is the Coriolis and centrifugal forces matrix, $G(q) = -\Gamma_{\text{G}_{\text{tp}}} - \Gamma_{\text{G}_{\text{arm}}}$ is the gravitational forces vector, and $\Gamma(t)$ is the control input vector. The main dynamic parameters of SPIDER4 parallel robot are summarized in Table 2.3.

### 2.4 Conclusion

In this chapter, we have introduced the three parallel manipulator prototypes that will be used for the experimental validation of the proposed control schemes. The experimental platforms can be listed as follows: Delta PKM at EPFL, Switzerland, VELOCE and SPI-
DER4 PKMs at LIRMM. The presented prototypes are all Delta-like parallel robots redundantly and non-redundantly actuated for which we can verify the performance and applicability of our control schemes in both cases.

The general mechanical structure of each parallel manipulator has been described. The kinematic modeling has been presented as well as the Jacobian matrices of all robots were established. Using the virtual work principle and relying on some assumptions (Assumptions 1 and 2), the inverse dynamics of all the parallel manipulators were formulated. The main characteristics, geometric parameters and dynamic parameters of each PKM were addressed in this chapter.

The established dynamic models will be used in the design of some control approaches within the next chapter.
## 3.1 Introduction

Control of parallel manipulators gained a wide interest in the last decades with the hope of achieving an adequate control design fit with the desired performances. The increasing fields of the parallel robot applications require high dynamic performances, high accuracy at low- and high-speed motions, and robustness against abundant uncertainties and nonlinearities. The control task of parallel manipulators is considered complicated and challenging due to the complexity of dynamics, uncertainties, parameters variation, and actuation redundancy.
Nonlinearity effect may increase considerably in parallel manipulators especially at high-speed motions leading to bad performance or loss of stability in some cases. The closed-loop structure of PKMs induce complex structure and coupled dynamics need careful synchronization between the actuators. Moreover, uncertainties can exist in parallel manipulators in two forms: i) unstructured uncertainties emerging from model simplifications, wear of the parts, measurement noise, geometric-uncertainties, etc., ii) structured uncertainties that appear as parameters variation and inexact knowledge of the dynamic parameters.

In the presence of all those uncertainties, actuation redundancy (in case of redundantly actuated manipulators) may lead to some antagonistic forces that appear as generated internal forces, called prestress. These forces can deteriorate the performance of the parallel manipulator and they should be taken into account by the control design.

Therefore, the need of advanced control strategies robust against uncertainties, changing nonlinearities, and disturbances arises. Enhancing some robust control strategies and improving the dynamic model of PKMs can lead to a better dynamic performance in terms of high-speed motions, precision, and robustness. This chapter provides a detailed explanation of the main proposed control solutions in this thesis. The contribution for each adopted control strategy is addressed and explained.

The main contributions can be listed briefly as follows:

1. A new time-varying feedback Robust Integral of the Sign of the Error (RISE) control strategy was developed for parallel manipulators. Some static feedback gains in the original RISE controller were replaced by nonlinear feedback ones aiming at more robustness towards disturbances, dynamic changes, and uncertainties.

2. A novel model-based super-twisting sliding mode control was proposed as an extension of the original second order super-twisting algorithm. The control structure formed of a dynamic feedforward term and a feedback super-twisting control can be more adequate for parallel manipulator control in real-time framework compared to the conventional super-twisting algorithm.

3. In the framework of improving the dynamic model, an actuator and friction dynam-
ics formulation was proposed being useful for model-based control strategies. A PD control with computed feedforward incorporating the actuator and friction dynamics was suggested in order to test the formulated model. Incorporating more dynamics can boost up the dynamic performance and compensate for more percentage of existing nonlinearities.

Furthermore, in case of redundantly actuated parallel manipulators, the adopted solution to avoid the effect of the internal prestress is explained at the end of this chapter.

### 3.2 Contribution 1: A new time-varying feedback RISE control

This study focuses on the development of a new class of the Robust Integral of the Sign of the Error (RISE) control law adequate for parallel manipulator systems. A revisit for the original RISE is done by altering some static feedback gains into time-varying nonlinear ones depending on the system states. The proposed controller takes advantage of both RISE control robustness towards uncertainties and the special behavior of nonlinear feedback gains towards time-varying parameters. A Lyapunov-based stability analysis is included to prove the semiglobal asymptotic tracking of the proposed new controller.

#### 3.2.1 Background on the original RISE controller

Consider the second order MIMO nonlinear dynamical systems represented as:

\[ M(x, \dot{x}) \ddot{x} + F(x, \dot{x}) = u \]  

(3.1)

where \( x(t), \dot{x}(t) \in \mathbb{R}^n \) denote the system states: position and velocity respectively, and \( \ddot{x}(t) \in \mathbb{R}^n \) denotes the acceleration, with "n" actuators. Note that \( x(t) \) and \( \dot{x}(t) \) are assumed to be measurable states. \( u(t) \in \mathbb{R}^n \) represents the control input. \( M(.,.) \in \mathbb{R}^{n \times n} \) and \( F(.) \in \mathbb{R}^n \) are uncertain nonlinear functions. In most of the real-world robotic systems, the mathematical model in (3.1) is poorly known and usually formulated with some simplifications, non-modelled phenomena and disturbances.
Let the output tracking error be defined as follows:

\[ e_1 = x_d - x \]  

(3.2)

where \( x_d(t) \in \mathbb{R}^n \) is the desired trajectory. In order to achieve an asymptotic tracking of a reference trajectory \( x_d(t) \) (\( e_1 \to 0 \) as \( t \to \infty \)), the system and the desired signal should have the assumed properties below.

**Property 5.** The matrix \( M(.) \in \mathbb{R}^{n \times n} \) is a symmetric positive-definite matrix and satisfies \( \forall \xi(t) \in \mathbb{R}^n \) the following inequality:

\[ m||\xi||^2 \leq \xi^T M(.) \xi \leq m(x)||\xi||^2 \]  

(3.3)

with \( m \in \mathbb{R} \) is a positive constant, and \( m(x) \in \mathbb{R} \) is a positive non-decreasing function. Notice that \( ||\cdot|| \) stands for the classical Euclidean norm.

**Property 6.** If \( x(t) \) and \( \dot{x}(t) \in \mathcal{L}_\infty \) (measurable and bounded), then \( F(.) \) is bounded. Moreover, the first and second partial derivatives of the elements of \( M(.) \) and \( F(.) \) with respect to \( x \) and \( \dot{x} \) exist and are also bounded.

**Property 7.** The chosen reference trajectory \( x_d(t) \in \mathbb{R}^n \) is differentiable till the 4th order, and its derivatives are bounded.

\[ x_d^{(i)}(t) \in \mathcal{L}_\infty \quad \text{for} \quad i = 0, 1, ..., 4 \]  

(3.4)

To develop the closed-loop error system equation, we need to introduce the auxiliary errors \( e_2(t), r(t) \in \mathbb{R}^n \) as follows:

\[ e_2 = \dot{e}_1 + \alpha_1 e_1 \]  

(3.5a)

\[ r = \dot{e}_2 + \alpha_2 e_2 \]  

(3.5b)

where \( \alpha_1, \alpha_2 \) are positive constant design gains added to increase the flexibility of tuning.

After differentiating (3.5b) with respect to time, multiplying both sides of the obtained equation by \( M(x, \dot{x}) \), then using the system dynamics \( (3.1) \), we get the equation below:

\[ M(.) \dot{r} = M(.) (\ddot{x}_d + \alpha_1 \ddot{e}_1 + \alpha_2 \dot{e}_2) + \dot{M(.)} \ddot{x} + \dot{F(.)} - \dot{u} \]  

(3.6)
3.2. CONTRIBUTION 1: A NEW TIME-VARYING FEEDBACK RISE CONTROL

By adding and subtracting the two terms \( \frac{1}{2} \dot{M}(.) r \) and \( e_2 \) for the right-hand side of the above-obtained equation (3.6), it can be rewritten as follows:

\[
M(.) r = -\frac{1}{2} \dot{M}(.) r - e_2 - u + N(.) \tag{3.7}
\]

where \( N(.) \) is defined as:

\[
N(.) \equiv N(x, \dot{x}, \ddot{x}, t) = M(.) (\ddot{x}_d + \alpha_1 \dot{e}_1 + \alpha_2 \dot{e}_2) + \dot{M}(.) (\dot{x} + \frac{1}{2} r) + e_2 + \ddot{f}(.) \tag{3.8}
\]

Based on the stability analysis introduced in [Xian et al., 2004], RISE control law that can achieve the control objective is defined as follows:

\[
u(t) = (k_s + 1) e_2(0) + \int_0^t (k_s + 1) \alpha_2 e_2(\sigma) d\sigma + \int_0^t \beta \text{sgn}(e_2(\sigma)) d\sigma \tag{3.9}
\]

where \( k_s \) and \( \beta \) are two positive constant gains, \( \text{sgn}(.) \) is the standard signum function. The integral of signum can hold smooth bounded disturbances for a sufficient condition on the feedback gain. The second term of (3.9) is used to ensure a zero input signal at time \( t_0 = 0 \).

Computing the first time derivative of (3.9) and substituting in (3.7) leads to the following closed-loop error system equation:

\[
M(.) \dot{r} = -\frac{1}{2} \dot{M}(.) r - e_2 - (k_s + 1) r - \beta \text{sgn}(e_2) + N(.) \tag{3.10}
\]

Let’s now consider the auxiliary function defined by: \( N_d(t) = N(x_d, \dot{x}_d, \ddot{x}_d, t) \). Then, one can add and subtract \( N_d(t) \) to the right-hand side of (3.10) obtaining the following:

\[
M(.) \dot{r} = -\frac{1}{2} \dot{M}(.) r - e_2 - (k_s + 1) r - \beta \text{sgn}(e_2) + \ddot{N} + N_d \tag{3.11}
\]

where

\[
\ddot{N}(x, \dot{x}, \ddot{x}, t) = N(x, \dot{x}, \ddot{x}, t) - N_d(t) \tag{3.12}
\]

Thanks to properties 5 and 6 of the nonlinear functions \( M(.) \) and \( F(.) \), and property 7 required in the desired trajectory, one can deduce that functions \( N_d(t) \) and \( \ddot{N}_d(t) \in \mathcal{L}_\infty \) (i.e. exist and bounded).
Since $N(.)$ is continuous, one can show that $\tilde{N}(.)$ can be upper bounded as follows:

$$||\tilde{N}|| \leq \rho(||z||)||z||$$

(3.13)

where $z(t) = [e_1 \quad e_2 \quad r]^T$, and $\rho(.) : \mathbb{R} \geq 0 \rightarrow \mathbb{R} \geq 0$ is a globally invertible nondecreasing function. For the proof of (3.13), the reader can refer to Lemma 1 in the appendix of [Patre, 2009].

Referring to [Xian et al., 2004], it can be verified that the control law of (3.9) ensures that all the closed-loop system states are bounded and converge to zero

$$e_1^{(i)}(t) \rightarrow 0 \quad \text{as} \quad t \rightarrow \infty, \quad \text{for} \quad i = 0, 1, 2$$

(3.14)

as long as the control gain $k_s$ is chosen large enough relative to the initial conditions of the system, $\alpha_1, \alpha_2 > 1/2$, and $\beta$ satisfies the following condition:

$$\beta > ||N_d(t)||_{L_\infty} + \frac{1}{\alpha_2} ||\dot{N}_d(t)||_{L_\infty}$$

(3.15)

where $||.||_{L_\infty}$ is the $L_\infty$ norm [Khalil, 2002].

### 3.2.2 Applications of RISE control

RISE feedback law is a continuous control solution dealing with Multi-Input-Multi-Output (MIMO) high-order nonlinear systems. This non-model-based control strategy can guarantee a semi-global asymptotic tracking under limited assumptions on the system uncertainties and time-varying parameters. RISE controller has been applied in different real-time applications thanks to the robustness and disturbances rejection provided by its feedback closed-loop architecture.

It has been proved experimentally in [Feemster, 2014] the high efficiency of RISE controller for disturbance rejection, compared to some classical controllers, in a directed energy platform experiencing jitter to promote beam regulation on a target. In [Fischer et al., 2014a; Chemori et al., 2016], the uncertainties and external disturbances accommodated by Autonomous Underwater Vehicles (AUVs) were treated using RISE feedback control in real operating conditions showing a satisfying efficiency and a good performance of such
RISE control law. Furthermore, a saturated RISE feedback control was designed and experimented on a two-link robot manipulator in [Fischer et al., 2014b] taking the advantage of high gain control strategies while guaranteeing saturation limits are not surpassed.

Besides, model-based adaptive control laws with RISE feedback have been proposed and applied to different platforms such as hydraulic load simulator [Luo et al., 2017] and a parallel robot Delta [Bennehar et al., 2018]. It has been proved experimentally the stability and high performances of such RISE-based control schemes.

### 3.2.3 Proposed time-varying feedback RISE control

#### 3.2.3.1 Motivation

Because of the powerful robustness and performance acquired by RISE and RISE-based control strategies, the idea of improving such controller arises. Although RISE control law has shown satisfying performances with the highly nonlinear and uncertain systems, the linear part of the RISE controller can be more sensitive to disturbances and measurement noise and less performant at high dynamic operating conditions [Bennehar et al., 2014]. Moreover, the fixed feedback gains of RISE control limits the tuning capabilities of the controller. For parallel manipulators, the dynamic performance is affected by the position in the workspace (trajectory), operating acceleration, payload handled, and other uncertainties. RISE controller can show good behavior when operating at nominal conditions, but it may come out with weak performances at high dynamic operating conditions.

Indeed, conventional linear control has been used in a wide range of industrial applications providing a good performance. However, its good performance is limited to a small range operation and around the nominal steady state only. At critical operating conditions (for example: high-speed, high-precision applications), linear control may degrade the performance and even lead to instability while nonlinear control can handle the variation in the nonlinear dynamics preserving the stability and the good performance [Slotine and Li, 1991; Khalil, 2002].

One of the most studied concepts in the area of nonlinear control is utilizing nonlinear functions as feedback gains able to adapt itself with the variation of the system states,
control inputs or other variables. A typical example of such nonlinear control is the NLPI

discussed before in section 1.8.2.2 that was proposed to enhance the adaptability and rob-


tude of the simple PID regulator [Jingqing, 1994]. The notion of NPID control was

extended to several fixed-gain controllers of parallel manipulators as shown already in

sections 1.8.2.2 and 1.8.3.6. All the nonlinear extended controllers show better dynamic

performances compared to the fixed-gain controllers in terms of tracking precision and

robustness towards uncertainties, disturbances, and varied parameters.

Motivated by the advantages of using nonlinear feedback gains instead of the fixed ones

and the significant performance of RISE feedback law for different applications, the Time-

Varying feedback RISE (TV-RISE) control is proposed as a new control methodology for

robotics. The proposition works on enhancing the RISE control law by replacing the fixed

feedback gains with time-varying ones that depend on the system states: position error,

velocity error, and the integral of the position error. We look to increase the robustness of

RISE regulator towards disturbances, uncertainties and variation of system nonlinearities

depending on the operating point. The TV-RISE controller can be more adequate for the

control problem of PKMs known with their high nonlinearities, uncertainties, and varied

performance with the dynamic operating conditions.

3.2.3.2 Control design

The original controller in (3.9) can be split up into two parts: a linear feedback part

based on the measured combined error $e_2$, and a nonlinear signum function. The linear

part consists of proportional and integral actions on the combined error, which is similar to

a PI controller but taking as input the combined error instead of the position error. These

two linear control actions may lead up to poor performances when dealing with highly

nonlinear systems at critical dynamic operating conditions. They have considerable sen-

sitivity to disturbances and limited tuning capabilities.

We propose to replace the proportional and the integral static feedback gains by non-
3.2. CONTRIBUTION 1: A NEW TIME-VARYING FEEDBACK RISE CONTROL

linear time-varying ones. The proposed TV-RISE controller is given as follows:

\[ u(t) = (K_s(\cdot) + 1)e_2(t) - (K_s(t_0) + 1)e_2(0) \]

\[ + \int_0^t (k_{s0} + 1)\alpha_2(\cdot)e_2(\sigma)d\sigma + \int_0^t \beta sgn(e_2(\sigma))d\sigma \]

(3.16)

with \( K_s(\cdot) \) and \( \alpha_2(\cdot) \) are two nonlinear feedback functions designed as suggested in [Shang et al., 2009]:

\[ K_s(\cdot) \equiv K_s(e_2, \epsilon_1, \delta_1) = \begin{cases} 
    k_{s0} |e_2|^{\epsilon_1 - 1}, & |e_2| > \delta_1 \\
    k_{s0} \delta_1^{\epsilon_1 - 1}, & |e_2| \leq \delta_1 
\end{cases} \]

(3.17a)

\[ \alpha_2(\cdot) \equiv \alpha_2(e_2, \epsilon_2, \delta_2) = \begin{cases} 
    \alpha_{20} |\int_0^t e_2|^{\epsilon_2 - 1}, & |\int_0^t e_2| > \delta_2 \\
    \alpha_{20} \delta_2^{\epsilon_2 - 1}, & |\int_0^t e_2| \leq \delta_2 
\end{cases} \]

(3.17b)

where \( k_{s0}, \alpha_{20}, \epsilon_1, \delta_1, \epsilon_2, \delta_2 \) are positive design parameters need to be chosen carefully. Indeed, to meet the desired performance, \( \epsilon_1 \) and \( \epsilon_2 \) are chosen within the intervals \([0.5, 1]\) and \([1, 1.5]\) respectively.

On the one hand, the selection of \( \epsilon_1 \) within the interval \([0.5, 1]\) can reduce the proportional gain \( K_s(\cdot) \) at high combined error values and increase it at small ones (see Figure 3.1). As long as the combined error remains within the small interval \([-\delta_1, +\delta_1]\) around zero, the proportional gain remains constant as a maximum saturated value. Notice that the combined error gives knowledge about both position and velocity errors. Thus, such variation of the gain could result in a rapid transition of the closed-loop system states and favorable damping.

On the other hand, the nonlinear feedback gain \( \alpha_2(\cdot) \) varies as function of the integral of the combined error (see Figure 3.2), which means that it is more concerned with the steady state combined errors (i.e. errors that persist with time). The choice of \( \epsilon_2 \) within the interval \([1, 1.5]\) gives large integral gain for the large steady state combined errors, and small integral gain for the small steady state combined errors as illustrated in Figure 3.2. As long as this error remains within the small interval \([-\delta_2, +\delta_2]\) around zero, the integral gain remains as a minimum constant value. This variation may accelerate the tracking
Choosing $\epsilon_1$ and $\epsilon_2$ in their corresponding intervals leads to globally bounded nonlin-
3.2. CONTRIBUTION 1: A NEW TIME-VARYING FEEDBACK RISE CONTROL

ear functions as follows (bounds can be realized from Figures 3.1 and 3.2):

\[ 0 < K_{sm} \triangleq k_{s0} \| e_2 \|_\infty^{-1} \leq K_s(\cdot) \leq k_{s0} \delta_1^{-1} \triangleq K_{sM} \]  
(3.18a)

\[ 0 < \alpha_{2m} \triangleq \alpha_{20} \delta_2^{-1} \leq \alpha_2(\cdot) \leq \alpha_{20} \| e_2 \|_\infty^{-1} \triangleq \alpha_{2M} \]  
(3.18b)

where \( \| \cdot \|_\infty \) indicates the infinity-norm.

Using the above introduced time-varying feedback gains in the standard equation of RISE controller may enhance the global tracking performance of such controller and may improve its robustness towards changes in system parameters. It is worth to confirm that the structure of the nonlinear functions is simple enough to be implemented in real-time experiments.

3.2.3.3 Closed-loop error dynamics:

In order to analyse the stability of the proposed TV-RISE controller, we need to establish its related closed-loop error equation based on the nonlinear MIMO system (3.1).

Let us first define the auxiliary error \( r(t) \) which is synthesized now using the nonlinear function \( \alpha_2(\cdot) \) as follows:

\[ r = \dot{e}_2 + \alpha_2(\cdot)e_2 \]  
(3.19)

Following the same previous procedure: differentiating \( r(t) \), multiplying both sides by \( M(\cdot) \), getting use of the system dynamics (3.1), and arranging the elements of the obtained equation leads to:

\[ M(\cdot)\dot{r} = -\frac{1}{2} \ddot{M}(\cdot)r - e_2 - \dot{u} + N(\cdot) \]  
(3.20)

where \( N(\cdot) \) is a new auxiliary function defined as follows:

\[ N(\cdot) \equiv N(x, \dot{x}, \ddot{x}, t) = M(\cdot)(\dddot{x}_d + \alpha_1 \dddot{e}_1 + \alpha_2(\cdot)e_2 + \dot{\alpha}_2(\cdot)e_2) \]

\[ + \ddot{M}(\cdot)(\ddot{x} + \frac{1}{2}r) + e_2 + \dddot{f}(\cdot) \]  
(3.21)

The equation of the closed-loop error system is then derived by differentiating the control law of TV-RISE controller (3.16) with respect to time and substituting it in (3.20). Intro-
ducing the supplementary function $\tilde{N}(., t)$ as in (3.12) allows the closed-loop error equation to be as follows:

$$M(.)\dot{r} = -\frac{1}{2}M(.)r - e_2 - K_s(.)e_2 - (K_s(.) + 1)e_2 - (k_{s0} + 1)\alpha_2(.)e_2$$

$$- \beta \text{sgn}(e_2) + \tilde{N} + N_d$$

(3.22)

Since $\alpha_2(.)$ is continuous, the upper bound of $\|\tilde{N}\|$ in (3.13) still exist.

### 3.2.3.4 Stability analysis

**Theorem 3.2.1.** The control law proposed in (3.16) applied to the second-order nonlinear MIMO system whose dynamic model is governed by (3.1) ensures that all the system signals are bounded and converge asymptotically to zero with time going to infinity, knowing that the design control gains are chosen such that

$$\beta > \|N_d(t)\|_{\infty} + \left(\frac{1}{\alpha_2M}\right)\|\tilde{N}_d(t)\|_{\infty}$$

with $\alpha_1 > 1/2$, $\epsilon_1 \in [0.5, 1]$, $\epsilon_2 \in [1, 1.5]$, and the bounds $K_sM$, $\alpha_2m$ in (3.18a) and (3.18b) are chosen large enough.

**Proof.** Let us first consider the function $L(t) \in \mathbb{R}$ defined as follows:

$$L(t) = r(N_d(t) - \beta \text{sgn}(e_2))$$

(3.23)

With the use of Lemma 1 in [Xian et al., 2004], we can conclude that if $\beta$ is chosen satisfying the following condition:

$$\beta > \|N_d(t)\|_{\infty} + \left(\frac{1}{\alpha_2M}\right)\|\tilde{N}_d(t)\|_{\infty}$$

(3.24)

then the following inequality holds:

$$\int_0^t L(\tau)d\tau \leq \beta |e_2(0)| - e_2(0) N_d(0)$$

(3.25)

Then, an additional function $P(t) \in \mathbb{R}$ needs to be defined as follows:

$$P(t) = \beta |e_2(0)| - e_2(0) N_d(0) - \int_0^t L(\tau)d\tau$$

(3.26)
3.2. **CONTRIBUTION 1: A NEW TIME-VARYING FEEDBACK RISE CONTROL**

Knowing that $P(t) \geq 0$, $\forall \ t \geq 0$ is ensured from (3.24) and (3.25).

We now introduce a continuous differentiable definite positive function $V : \mathbb{R}^{3n+1} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ as follows:

$$V(y, t) = \frac{1}{2} y^T e_1 + \frac{1}{2} r^T M(.) r + P$$  \hspace{1cm} (3.27)

where $y = [z^T \sqrt{P}]^T$ and $z(t)$ is defined previously. In view of the characteristics of the matrix $M(.)$ stated by Property 5 and its bounds in (3.3), $V(y, t)$ can be bounded as follows:

$$\lambda_1 \|y\|^2 \leq V(y, t) \leq \lambda_2 (\|y\|) \|y\|^2$$  \hspace{1cm} (3.28)

being $\lambda_1 = (1/2) \min\{1, m\}$ and $\lambda_2 = (1/2) \max\{\pi(\|y\|), 1\}$.

Applying the time derivative of (3.27), and using equations (3.22), (3.23) and (3.26) leads to:

$$\dot{V} = e_1^T e_2 - \alpha_1 e_1^T e_1 - r^T e_2 - \dot{K}_s(.) r^T e_2 - (K_s(.) + 1) r^T r$$

$$+ (K_s(.) + 1) \alpha_2 r^T e_2 - (k_s0 + 1) \alpha_2 e_2^T e_2 + r^T \tilde{N}$$  \hspace{1cm} (3.29)

where $\dot{K}_s(.)$ is the time derivative of the nonlinear function $K_s(.)$. Now, we need to find an upper bound for $\dot{V}$ in (3.29).

Using the conventional inequality for any two vectors, $a$ and $b$ namely $a^T b \leq (\|a\|^2 + \|b\|^2)/2$, one can write:

$$\dot{V} \leq \frac{1}{2} \|e_1\|^2 + \frac{1}{2} \|e_2\|^2 - \alpha_1 \|e_1\|^2 - \frac{1}{2} \|r\|^2 - \|e_2\|^2 - \frac{|K_{\text{sm}}|^2 \|r\|^2}{2}$$

$$- \frac{(K_{\text{sm}} + 1) \|e_2\|^2}{2} + \frac{(K_{\text{sm}} + 1) \alpha_2 \|e_2\|^2}{2}$$

$$+ \frac{(K_{\text{sm}} + 1) \alpha_2 \|e_2\|^2}{2}$$  \hspace{1cm} (3.30)

where $K_{\text{sm}}$ is a lower bound for $K_s(.)$. After developing and re-arranging (3.30) we obtain:

$$\dot{V} \leq -\zeta_1 \|e_1\|^2 - \zeta_2 \|e_2\|^2 - \zeta_3 \|r\|^2 - \mu \|r\|^2 + \|r\| \rho (\|z\|) \|z\|$$  \hspace{1cm} (3.31)

where $\zeta_1, \zeta_2, \zeta_3$ and $\mu$ are constants to be chosen positive defined as follows:

$$\zeta_1 = \alpha_1 - \frac{1}{2}$$  \hspace{1cm} (3.32a)
\[ \zeta_2 = \frac{1}{2} \left( 1 - K_{sM} \alpha_{2m} + k_{s0} \alpha_{2m} \right) \]  
(3.32b)

\[ \zeta_3 = \frac{1}{2} \left( 3 + 2K_{sm} - K_{sM} \alpha_{2m} - \alpha_{2m} \right) \]  
(3.32c)

\[ \mu = \frac{1}{2} (k_{s0} + 1) \alpha_{2m} \]  
(3.32d)

From (3.32a), \( \alpha_1 \) should satisfy the condition \( \alpha_1 > 1/2 \). Equation (3.31) can be rewritten as follows:

\[ \dot{V} \leq -\lambda_3 \| z \|^2 - \left( \mu \| r \|^2 - \| r \| \rho (\| z \|) \| z \| \right) \]  
(3.33)

being \( \lambda_3 = \min \{ \zeta_1, \zeta_2, \zeta_3 \} \). Using the mathematical remarkable square identities \( (a - b)^2 = a^2 - 2ab + b^2 \), (3.33) can be rewritten as follows:

\[ \dot{V} \leq -\left( \lambda_3 - \frac{\rho^2(\| z \|)}{4\mu} \right) \| z \|^2 \triangleq -c \| z \|^2 \]  
(3.34)

where \( c \) is some positive constant, which implies that the following inequality holds:

\[ \lambda_3 > \frac{1}{4\mu} \rho^2(\| z \|) \]  
(3.35)

Let us define the region \( \mathcal{D} \) using inequality (3.35) as follows:

\[ \mathcal{D} = \left\{ y \in \mathbb{R}^3 \times \mathbb{R}_{\geq 0} \mid \| y \| < \rho^{-1} \left( 2 \sqrt{\lambda_3 \mu} \right) \right\} \]  
(3.36)

We know that \( V(y, t) \in \mathcal{L}_\infty \) is a continuously differentiable function such that \( W_1(y) \leq V(y, t) \leq W_2(y) \) (see equation (3.28)) and \( \dot{V}(y, t) \leq -W(y) \) (from equation (3.34)). Hence \( e_1, e_2, r \in \mathcal{L}_\infty \).

\( W_1(y), W_2(y) \) are continuous positive-definite functions \( \forall \ t \geq 0 \) and \( \forall \ y \in \mathcal{D} \), and \( W(y) \) is uniformly continuous positive-semidefinite function.

Given that the initial conditions \( y(0) \in \mathcal{S} \), a subset of \( \mathcal{D} \) defined as follows:

\[ \mathcal{S} = \left\{ y \in \mathcal{D} \mid W_2(y) < \lambda_1 \left( \rho^{-1} \left( 2 \sqrt{\lambda_3 \mu} \right) \right)^2 \right\} \]  
(3.37)
then we can conclude, using Lemma 2 of [Xian et al., 2004], that \( \|z(t)\|^2 \to 0 \) as \( t \to \infty \), \( \forall y(0) \in S \). This means that all the closed-loop system states \((e_1, e_2, r)\) asymptotically converge to zero with time.

\[
e_{1}^{(i)}(t) \to 0 \quad \text{as} \quad t \to \infty \quad \forall y(0) \in S
\]  

(3.38)

and here the proof is concluded.

\[\square\]

### 3.2.3.5 Application of the proposed controller to PKMs

For an adequate control design and implementation, we re-define the position error in (3.2) for parallel manipulators of \( n \) actuators as a difference between the desired joint angle \( q_d \in \mathbb{R}^n \) and the actual measured one \( q \in \mathbb{R}^n \) as follows:

\[
e_1 = q_d - q
\]  

(3.39)

The measurement of the actual angle position is performed by means of encoders integrated in the motors, and the position in Cartesian space is computed using the forward kinematics of the robot as common for most parallel robots.

The dynamic model of parallel manipulators (1.3) is considered as a second order non-linear MIMO system with a structure similar to the system equation (3.1).

Consequently, the mass and inertia matrix \( M(q) \) is a symmetric positive-definite matrix satisfying the boundedness condition introduced in Property 5. The dynamics of a parallel manipulator satisfy Property 6 such that \( q(t) \) and \( \dot{q}(t) \) are measurable and are bounded giving that \( C(q, \dot{q}) \) and \( G(q) \) are bounded. Then, the first and second partial derivatives of \( M(q) \) with respect to \( q \) and those of \( C(q, \dot{q}) \), \( G(q) \) with respect to \( q, \dot{q} \) exist and bounded. Also, the chosen desired trajectory \( q_d(t) \) satisfies the property of differentiability and boundedness reported in Property 7.

Therefore, parallel robot dynamics fit the design of RISE-based controllers and it is possible to implement the proposed control schemes in real-time experiments. The proposed TV-RISE control architecture is summarized in the block diagram depicted in Figure 3.3.
3.3 Contribution 2: A novel model-based super-twisting sliding mode control

This study focuses on the development of a new model-based Super-Twisting Sliding Mode Control (ST-SMC) for parallel manipulators taking into account their highly nonlinear dynamics. The proposed controller relies on the robust second order ST-SMC that provides a continuous control signal and a finite-time convergence of the sliding variable and its derivative. A feedforward dynamic term is enclosed within the proposed control strategy aiming at high dynamic performances in terms of nonlinearities compensation and accuracy. The designed controller has the advantage of relying on the desired trajectories instead of the measured ones for which it can be less sensitive to noise measurements (reduced chattering), more computationally efficient, and more robust. The stability analysis of the proposed controller is established using a Lyapunov function candidate. It shows a
finite-time convergence of the sliding variable and a local asymptotic convergence of the tracking error.

### 3.3.1 Background on the super-twisting sliding mode control

Consider the Single-Input-Single-Output (SISO) second order nonlinear uncertain system below:

\[ \ddot{x}_1 = f(x_1, \dot{x}_1, t) + g(x_1, \dot{x}_1, t) \nu \]  

(3.40)

where \( x_1, \dot{x}_1 \) are the system states with \( x_1 \) being the output, \( \nu \) is the scalar control signal, \( f(.) \) represents the unknown bounded uncertainties and perturbations, such that \(|f(.)| \leq L\) with \( L \) being a positive constant, and \( g(.) \neq 0 \) is the known nonlinearity. Assuming that \( g(.) \) is positive and invertible for all \( t \), the state-variable presentation of (3.40) can be written as follows:

\[
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= u + f(x, t)
\end{aligned}
\]  

(3.41)

where \( x = [x_1, x_2]^T \) is the state vector and \( u \) is a control input such that \( \nu = g^{-1}(x, t)u \).

#### 3.3.1.1 Main concept of the sliding mode control

The control objective is to develop a control signal \( u(x_1, x_2) \) that drives the state variables to zero as time goes to infinity in the presence of uncertainties and perturbations \( f(x, t) \) [Shtessel et al., 2014]. A linear state-feedback control law can achieve the asymptotic stability if and only if \( f(x, t) \equiv 0 \), such that \( u \) is given as follows:

\[ u = -k_1 x_1 - k_2 x_2, \quad k_1, k_2 > 0 \]  

(3.42)

Indeed, the state variables converge to a bounded domain around zero depending on the chosen control gains and the system perturbations.

Then, Sliding Mode Control (SMC) algorithm is proposed to attain the asymptotic convergence of the state variables in the presence of the unknown system perturbations [Shtessel et al., 2014]. It is all about inserting a nonlinear discontinuous term into the controller responsible for rejecting the disturbances, driving the state variables to a sliding surface in
a finite time, and restricting them on the surface thereafter in the presence of the bounded disturbances. First, a new variable in the state space is defined representing the sliding surface:

\[ s = x_2 + cx_1 \]  \hspace{1cm} (3.43)

where \( c \) is a positive constant. The above sliding surface results with the desired compensated dynamics, \( \dot{x}_1 + cx_1 = 0 \), that leads to the asymptotic convergence of \( x_1, x_2 \rightarrow 0 \) without any effect of the disturbance \( f(x, t) \). Thus, it is clear that we need to drive the sliding variable \( s \) to zero by the control \( u(x_1, x_2) \) in finite time so that we can obtain the asymptotic convergence \( \lim_{t \rightarrow \infty} x_1, x_2 = 0 \). Applying some Lyapunov function techniques (\( V = \frac{1}{2}s^2 \)) to the sliding surface dynamics, the required first-order SMC signal and the finite time of the reaching phase to the sliding surface can be derived respectively as follows [Shtessel et al., 2014]:

\[ u = -cx_2 - \rho \text{sign}(s) \]  \hspace{1cm} (3.44)

\[ t_r \leq \frac{2V^{1/2}(s_0)}{\alpha} \]  \hspace{1cm} (3.45)

where \( s_0 \) is the sliding variable value at time \( t = 0 \), the control gain \( \rho = L + \frac{\alpha}{\sqrt{2}} \) and \( \alpha \) is a positive constant related to the reaching time. The introduced signum function works on compensating the bounded disturbances and achieving the asymptotic convergence of the state variables in the presence of perturbations and uncertainties. However, its high-frequency switching nature leads to a finite amplitude and frequency switching control signal, zigzag behavior, due to the discrete-time nature of the control implementation which is known as chattering. This oscillation in the control signal is undesirable for practical implementations being harmful to the actuators, the mechanical parts, and the control accuracy.

Mainly, the advantages of the first-order SMC are:

- robustness due to theoretical exact compensation of the bounded matched disturbances without being affected by such disturbances.
- reduced order of sliding equations.
- finite-time convergence of the sliding surface.
3.3. **CONTRIBUTION 2: A NOVEL MODEL-BASED SUPER-TWISTING SLIDING MODE CONTROL**

and the disadvantages are:

- chattering.
- asymptotic convergence of the state variables and not finite-time convergence.
- relative degree one of the sliding surface such that higher order derivatives are required for the sliding surface design.

Several solutions were proposed in the literature to produce a smooth/continuous control signal and reduce the chattering such that Quasi-Sliding Mode and Asymptotic Sliding Mode [Bartoszewicz, 1998; Lee et al., 1999; Christopher and Spurgeon, 1998]. However, the price to be paid for obtaining a smooth control signal can be less robustness and accuracy (Quasi-Sliding Mode) or asymptotic convergence of both sliding surface and state variables (Asymptotic Sliding Mode).

Furthermore, second-order SMC algorithms can achieve finite-time convergence of the sliding variable and its derivative. It can ensure quadratic precision of the convergence with respect to the sliding output as well as the sliding surface is no longer needed \( s = x_1 \) (Twisting and Terminal controllers) [Zhihong et al., 1994; Emel’yanov et al., 1996; Yu and Man, 1996]. Moreover, the sliding dynamics are reduced to the order \( r - 2 \) for the systems with relative degree \( r \). The relative degree of a system describes how the control input enters the system. It is equal to the number of times we have to differentiate the output of a system before the input appears explicitly. Nevertheless, for the systems of relative degree two, the controller still produces a discontinuous control signal and chattering phenomenon persists.

Moreover, the second-order Super-Twisting SMC (ST-SMC) algorithm has been proposed and developed, resulting in an exact finite-time convergence of the sliding variable and its derivative, a high accurate asymptotic convergence of the variable states, and a continuous control signal [Levant, 1993].
3.3.1.2 Super-twisting sliding mode control

The ST-SMC algorithm that achieves the asymptotic stability of system (3.41) and finite-time convergence of sliding surface (3.43) is given as follows [Levant, 1993]:

\[
\begin{align*}
    u &= -k_1 |s|^{1/2} \text{sign}(s) + w \\
    \dot{w} &= -k_2 \text{sign}(s)
\end{align*}
\]  

(3.46)

where \( k_1, k_2 \) are positive control gains. Applying the time derivative to the sliding surface, the sliding variable dynamics can be written as follows:

\[
\dot{s} = -k_1 |s|^{1/2} \text{sign}(s) + w + f(x, t)
\]  

(3.47)

The Lyapunov candidate that proves the asymptotic stability is given as follows [Moreno and Osorio, 2008]:

\[
V = \frac{1}{2} \xi^T P \xi
\]  

(3.48)

where \( \xi = [|s|^{1/2} \text{sign}(s) \quad w]^T \) and

\[
P = \begin{pmatrix}
    4k_2 + k_1^2 & -k_1 \\
    -k_1 & 2
\end{pmatrix}
\]

is chosen to be a positive definite matrix.

Following a similar manner in [Moreno and Osorio, 2008], the expression of the derivative of the Lyapunov function can be derived as follows:

\[
\dot{V} = -\frac{1}{2|s|^{1/2}} \xi^T Q \xi + \frac{f(x, t)}{|s|^{1/2}} F^T \xi
\]  

(3.49)

where \( F = [2k_2 + k_1^2 \quad -k_1]^T \) and

\[
Q = k_1 \begin{pmatrix}
    2k_2 + k_1^2 & -k_1 \\
    -k_1 & 1
\end{pmatrix}
\]

Knowing that the bounded perturbation satisfies \( f(x, t) \leq \epsilon |s|^{1/2} \), \( \epsilon \) being a positive constant, it can be shown that

\[
\dot{V} \leq -\frac{1}{|s|^{1/2}} \xi^T Q \xi
\]  

(3.50)
3.3. CONTRIBUTION 2: A NOVEL MODEL-BASED SUPER-TWISTING SLIDING MODE CONTROL

with

\[
\tilde{Q} = \frac{k_1}{2} \begin{pmatrix}
2k_2 + k_1^2 - \left(\frac{4k_2}{k_1^2} + k_1\right)e & -k_1 + 2e \\
-k_1 + 2e & 1
\end{pmatrix}
\]

The global asymptotic stability is achieved when \( \dot{V} \) is negative definite, which means \( \tilde{Q} > 0 \). Thus, the control gains should satisfy the following conditions:

\[
k_1 > 2\varepsilon \\
k_2 > k_1 \frac{5k_1 + 4\varepsilon^2}{2(k_1 - 2\varepsilon)}
\]

It can be shown also that the states converge to zero in finite-time \( t_r \) as in (3.45) with

\[
\alpha = \frac{\lambda_{\max}^{-1/2}(P)\lambda_{\min}^{1/2}(Q)}{\lambda_{\min}(P)},
\]

such that \( \lambda_{\min}(\cdot), \lambda_{\max}(\cdot) \) are the minimum and maximum eigen values of a matrix respectively.

3.3.2 Applications of the super-twisting sliding mode control

An application of the ST-SMC algorithm to motion control systems was illustrated by numerical simulations to an under-actuated robotic system in [Rivera et al., 2011] ensuring the facilitation of the motion control design and elimination of the chattering phenomenon at the outputs. In [Derafa et al., 2012], the ST-SMC technique has been designed and implemented for the attitude tracking problem of a quadrotor. The implemented control law has the general formula of a computed torque approach based on the super-twisting algorithm which is able to ensure robustness with respect to bounded external disturbances. The experimental results show the good performance of the proposed controller in terms of stabilization and tracking accuracy. Another version of the model-based ST-SMC algorithm has been implemented to a mobile robot in [Solea and Cernega, 2015] based on a continuous sliding surface (integrated error). Simulation and experimental results show better performances of the proposed controller in terms of eliminating the chattering and reducing the tracking errors compared to conventional SMCs.

Furthermore, several variable-gain ST-SMC versions have been proposed for different experimental prototypes (robotic arm [Mobayen et al., 2017], mass-spring-damper [Gonzalez et al., 2012], seesaw module [Oliveira et al., 2018] and space robot [Zhao et al., 2018])
allowing to compensate for a larger class of perturbations (by estimating the maximum bound of the perturbations) than the conventional ST-SMC and to further reduce the chattering effect of the classical first-order SMCs.

3.3.3 Proposed model-based super-twisting sliding mode control

3.3.3.1 Motivation

Recalling the nonlinear dynamical system of the parallel manipulators, one can write:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \Gamma_d = \Gamma$$  (3.52)

where $\Gamma_d \in \mathbb{R}^n$ represents the vector of the external disturbances, uncertainties, and non-modeled phenomena. Assuming that $\Gamma_d$ is bounded, the conventional ST-SMC algorithm that can be designed for such type of models can be expressed as follows [Jeong et al., 2018]:

$$\Gamma = M(q)(\dot{\bar{r}} + \Gamma_{\text{ST-SMC}})$$  (3.53)

where $\bar{r} = \ddot{q}_d - \lambda \dot{e}$ with $\lambda$ being a positive control gain such that the tracking error is defined as $e = q - q_d$, $\Gamma_{\text{ST-SMC}}$ is the control structure given in (3.46), and $s = \dot{e} + \lambda e$ being the sliding surface. One of the main drawbacks of this control structure is the lack of some parts of the model dynamics (including only the inertia matrix) which may decrease the dynamic performance of a parallel robot. Incorporating the structured nonlinearities within the closed-loop control is very essential for parallel manipulators known of high nonlinear dynamics that increase considerably when operating at high dynamic conditions (high-speed motions, payload handling, etc.).

Other implementations of the ST-SMC algorithm can be explained as considering the final control input equal to the original ST-SMC given in (3.46) without any consideration of the dynamics [Rivera et al., 2011]. This decentralized implementation is insufficient for compensating the nonlinearities, enhancing the dynamic performance, and increasing the robustness towards uncertainties. The aforementioned control approaches depend only on the high values of the control design gains, $k_1$ and $k_2$, to achieve the desired tracking performance.
3.3. CONTRIBUTION 2: A NOVEL MODEL-BASED SUPER-TWISTING SLIDING MODE CONTROL

The control design of the ST-SMC approach taking into account the nonlinear dynamics within computed torque formulation can be expressed as follows [Derafa et al., 2012; Mobayen et al., 2017]:

\[ \Gamma = M(q)(\ddot{r} + \Gamma_{ST-SMC}) + C(q, \dot{q})\dot{q} + G(q) \]  

(3.54)

The above computed torque control based on the ST-SMC algorithm needs well and precise knowledge of the dynamic model to obtain good tracking performances. Relying on the measured signals to compute the dynamic model in an online form can make the controller more sensitive to noise measurements decreasing the global performance and increasing the chattering. Further, it has been shown in [Khalil and Dombre, 2004] that computed torque control is unable to cope well with modeling errors. Moreover, for computationally heavy dynamic model, this controller may face significant limitations in real-time implementations.

The implementations of the variable-gain ST-SMC strategies consider all or part of the dynamic nonlinearities of the system as perturbations. This provides an online adaptation of the maximum bound of those perturbations to be compensated by the robustness term (sign function). However, in the presence of structured nonlinearities as the dynamic model of parallel manipulators, it can be better to enhance the control by a nonlinear compensation term based on the dynamic model.

Avoiding all the above issues, we propose to replace the computed torque with a feedforward term having at the end a super-twisting feedforward sliding mode control approach. The feedforward dynamic term relying on the desired trajectories instead of the measured ones can be much more efficient in the computation cost since it can be computed offline and stored to be used within the control. Also it is insensitive to the sensor measurement noises providing better performance and less chattered signal.

However, the experimental work done in [Cheng et al., 2003] has proven that an augmented PD control provides better tracking accuracy than a computed torque controller especially at high-speed motions. Thus a PD with computed feedforward which is exactly an augmented PD fed with the desired trajectory can perform better than computed torque control. Besides, the superiority in tracking performance of a PD control with computed
feedforward among a simple PD, a computed torque, and an augmented PD controllers has been proved on an experimental robotic arm in [Reyes and Kelly, 2001].

To deal with the parametric uncertainties existing in parallel robots (for example: variation of the handled payload in pick-and-place applications), a dynamic adaptive ST-SMC controller is proposed as an extension to the feedforward ST-SMC. The feedforward ST-SMC approach is considered as a preferable formulation to introduce the adaptive control which offer an online adaptation of the dynamic parameters that may vary while operating the robot.

The proposed controllers takes the advantages of the standard ST-SMC algorithm such as robustness towards disturbances, accurate convergence with the presence of external disturbances, and continuous control output, as well as the advantages of the feedforward dynamic term such as compensating for the structured nonlinearities, insensitivity towards measurement noises, computation-efficiency, and coping parametric uncertainties provided by dynamic adaptation algorithms.

3.3.3.2 Control design

This section provides a step-by-step derivation of the proposed feedforward ST-SMC algorithm. The standard sliding surface for a super-twisting SMC algorithm can be given as follows:

\[ s = \dot{e} + \Lambda e \]  

with \( e = q_d - q \) being the tracking error and \( \Lambda \) being a positive definite diagonal matrix of control gains for each axis.

Combining the defined sliding surface (3.55) and the dynamic system (3.52) leads to the equation below:

\[ M(q)(\ddot{q}_d - \dot{s} + \Lambda \dot{e}) + C(q, \dot{q})\dot{q} + G(q) + \Gamma_d = \Gamma \]  

Let us define an auxiliary reference velocity trajectory \( \dot{r} = \dot{q}_d + \Lambda \dot{e} \) shifted from the actual desired one by \( \Lambda \dot{e} \). Then, (3.56) can be re-written in the form below:

\[ M(q)\dot{r} - M(q)\dot{s} + C(q, \dot{q})\dot{q} + G(q) + \Gamma_d = \Gamma \]
3.3. **CONTRIBUTION 2: A NOVEL MODEL-BASED SUPER-TWISTING SLIDING MODE CONTROL**

where $\ddot{r}$ is the corresponding shifted desired acceleration. The sliding surface dynamics can be obtained from (3.57) as follows:

$$\dot{s} = M^{-1}(q)\left(-\Gamma + M(q)\ddot{r} + C(q, \dot{q})\dot{q} + G(q) + \Gamma_d\right)$$  \hspace{1cm} (3.58)

Thus, the control input $\Gamma$ can be chosen in a way having an exact compensation for the nonlinearities of the dynamic model as well as theoretical compensation for the disturbance term. The conventional model-based super-twisting SMC control is defined as follows:

$$\Gamma = M(q)\left(\ddot{r} + K_2|s|^{\frac{1}{2}}\text{sign}(s) + w\right) + C(q, \dot{q})\dot{q} + G(q)$$

$$\dot{w} = K_3\text{sign}(s)$$ \hspace{1cm} (3.59)

where $K_2, K_3$ are two positive definite diagonal matrices. Note that the control law in (3.59) is in the form of computed torque control based on the super-twisting algorithm.

For highly nonlinear dynamic systems (especially those of large mass and inertia parameters), this control law may be sensitive to the measurements noise decreasing the global performance and increasing the chattering effect. Chattering could be augmented with this control law knowing that the conventional super-twisting algorithm reduces excessively the effect of chattering. As a result, this control can deteriorate the dynamic performance of the parallel manipulator in terms of precision and robustness.

To avoid all the aforementioned problems, a feedforward super-twisting SMC algorithm is proposed that can compensate for the abundant nonlinearities as well as take the advantages of the standard super-twisting algorithm. It has been shown experimentally the benefits of the feedforward-based controllers in terms of computational cost, robustness towards noises and disturbances, and nonlinearities compensation [Bennehar et al., 2017; Natal et al., 2015].

The proposed feedforward ST-SMC algorithm comprises three main parts: the feedforward term, the super-twisting algorithm, and a feedback term added to insure the stability of the system. The control equation of the proposed control law is given as follows:

$$\Gamma = M(q_d)\ddot{q}_d + C(q_d, \dot{q}_d)\dot{q}_d + G(q_d) + K_1s + K_2|s|^{\frac{1}{2}}\text{sign}(s) + w$$

$$\dot{w} = K_3\text{sign}(s)$$ \hspace{1cm} (3.60)

where $K_1$ is a positive definite diagonal matrix of the feedback control gains.
3.3.3.3 Stability analysis

Theorem 3.3.1. Assuming that the desired velocity is upper bounded, the joint position and velocity tracking errors \( e = q_d - q, \dot{e} \) of a robotic manipulator of dynamic model (3.52) with bounded disturbances follow a local asymptotic convergence under the feedforward super-twisting sliding mode control given by (3.60), with the proper choice of \( \Lambda, K_1, K_2 \) and \( K_3 \).

Moreover, the local asymptotic stability is achieved in a finite time of maximum value \( T = \frac{2V^2(s_0)}{\gamma} \), where \( s_0 = \dot{e}_0 + \Lambda e_0 \) is the initial value of the sliding variable, \( \gamma \) is a positive constant depending on the control gains \( \Lambda, K_1, K_2, K_3 \) and the disturbance’s upper bound, and \( V(s) \) is a positive radially unbounded function given in (3.63).

Proof. Considering the nonlinear dynamical system (3.52) of bounded disturbances \( \Gamma_d \), the sliding variable (3.55), and the control equation of the proposed feedforward ST-SMC (3.60), one can establish the sliding variable dynamics as follows:

\[
M \dot{s} = -K_1 s - K_2 |s|^{1/2} \text{sign}(s) - w - h(q, \dot{q}) - C \dot{e} + \Lambda M \dot{e} + \Gamma_d \tag{3.61}
\]

with \( h(q, \dot{q}) \) being the residual dynamics expressed as follows:

\[
h \equiv h(q, \dot{q}) = \left[ M(q_d) - M(q) \right] \ddot{q}_d + \left[ C(q_d, \dot{q}_d) - C(q, \dot{q}) \right] \dot{q}_d + \left[ G(q_d) - G(q) \right] \tag{3.62}
\]

Without loss of generality, the scalar notation will be considered in the coming part of the stability analysis for simplicity purposes. For system (3.61), the following Lyapunov function is proposed to prove its stability:

\[
V(s) = \frac{1}{2} \xi^T P \xi \tag{3.63}
\]

with \( \xi = [|s|^{1/2} \text{sign}(s), w]^T \) and

\[
P = \begin{pmatrix}
4K_3 + K_2^2 & K_2 \\
K_2 & 2
\end{pmatrix} \tag{3.64}
\]

chosen to be a positive definite matrix. One can notice that \( V(s) \) is continuously differentiable everywhere, except on \( s = 0 \). However, from (3.61), the state trajectories of the system just cross \( s = 0 \) and cannot stay on it, except when the origin \( (s = 0) \) has been reached.
Thus, \( V(s) \) is differentiable for almost every \( t \) and one can apply Lyapunov’s theorem for the points where \( V(s) \) is differentiable [Moreno and Osorio, 2008].

The solutions of the discontinuous differential equation (3.61) are interpreted as the ones of the differential inclusion \( \dot{s} \in f(s) \). \( \text{sign}(s) \) assigns the interval \([-1,1]\) for \( s = 0 \). Then, since \( 0 \in f(0) = [-1,1] \), it follows that \( s = 0 \) is an equilibrium point [Moreno and Osorio, 2008]. The time derivative of \( V(s) \) along the solutions of the system leads to the following:

\[
\dot{V} = -\frac{1}{2M|s|^2} \xi^T Q \xi + \frac{1}{2M|s|^2} f(s,t) F^T \xi \tag{3.65}
\]

where \( F = [K_2^2 + 4K_3, K_2]^T \), \( f(s,t) = \left( \Gamma_d - K_1 s - (C - \Lambda M) \dot{e} - h \right) \), and

\[
Q = \begin{pmatrix}
K_2^3 + 4K_2K_3 - 2K_2K_3M & K_2^2 + 2K_3 - 2K_3M \\
K_2^2 + 2K_3 - 2K_3M & K_2
\end{pmatrix} \tag{3.66}
\]

\( M \) is positive (or positive definite matrix in non-scalar case) and \( K_2, K_3 \) are chosen such that \( Q \) becomes positive definite matrix. The perturbation term \( |f(s,t)| \) can be upper bounded as follows:

\[
|f(s,t)| \leq K_1 |s| + |C| |\dot{e}| + \Lambda |M| |\dot{e}| + |h| + |\Gamma_d| \tag{3.67}
\]

According to [Kelly et al., 2005], the Euclidean norm of the residual dynamics of a robotic manipulator (\(|h(q, \dot{q})|\)) can be upper bounded by the following (for vector form):

\[
|h(q, \dot{q})| \leq k_{h_1} |\dot{e}| + k_{h_2} |e| \tag{3.68}
\]

where \( k_{h_1}, k_{h_2} \) are two positive constants. Considering the scalar case, \(|h| \leq k_{h_1} |\dot{e}| + k_{h_2} |e|\), Properties 1 and 2 of mass matrix and Coriolis and centrifugal matrix in Chapter 1, Section 1.7.2, and assuming that the disturbance function is globally bounded by \(|\Gamma_d| \leq \epsilon |s|\), such that \( \epsilon > 0 \), the inequality (3.67) can be developed to be as follows:

\[
|f(s,t)| \leq (K_1 + \epsilon) |s| + (k_C |q_d| M + \Lambda k_M + k_{h_1}) |\dot{e}| + k_{h_2} |e| + k_C |\dot{e}|^2 \tag{3.69}
\]
where $k_C, k_M$ are two positive constants. $|q_d|_M$ is the upper bound of the desired velocity trajectory. For small values of $|\dot{e}|$ (when $s$ is around the origin), the linear term $(k_C|q_d|_M + \Lambda k_M + k_{h1})|\dot{e}|$ dominates the quadratic term $k_C|\dot{e}|^2$. Then, using the facts that $|e| \leq \frac{1}{\Lambda}|s|$ and $|\dot{e}| \leq |s|$, the term $|f(s, t)|$ can be locally upper bounded as follows:

$$|f(s, t)| \leq \mu |s|$$  \quad (3.70)

with $\mu = K_1 + e + k_C|q_d|_M + \Lambda k_M + k_{h1} + \frac{1}{\Lambda} k_{h2}$. Moreover, if $s$ is around the origin, it implies that $|s| \leq |s|^\frac{1}{2}$. Thus, (3.70) can be expressed as follows:

$$|f(s, t)| \leq \mu |s|^\frac{1}{2} \quad (3.71)$$

Making use of (3.71), $\dot{V}(s)$ in (3.65) can be locally upper bounded as follows:

$$\dot{V} \leq -\frac{1}{2M|s|^2} \xi^T \tilde{Q} \xi + \frac{\mu}{2M|s|^2} |s|^\frac{1}{2} |F^T \xi|$$  \quad (3.72)

leading to

$$\dot{V} \leq -\frac{1}{2M|s|^2} \xi^T \tilde{Q} \xi$$  \quad (3.73)

where

$$\tilde{Q} = \begin{pmatrix}
K_2^2 + 4K_2K_3 - 2K_2K_3M - \mu(K_2^2 + 4K_3) & K_2^2 + 2K_3 - 2K_3M - \frac{\mu}{2}K_2 \\
K_2^2 + 2K_3 - 2K_3M - \frac{\mu}{2}K_2 & K_2
\end{pmatrix}$$  \quad (3.74)

with $\lambda_{\min}(\tilde{Q})$ is the minimum eigen value of $\tilde{Q}$ and $||\xi||$ is the Euclidean norm of vector $\xi$. $\dot{V}$ is negative definite if $\tilde{Q}$ is a positive definite matrix. Then, following Lyapunov’s direct method, if $\Lambda, K_1, K_2, K_3$ are selected such that $\tilde{Q} > 0$, the origin $s = 0$ is an equilibrium point that is locally asymptotically stable.
3.4. CONTRIBUTION 3: ACTUATOR AND FRICTION DYNAMICS FORMULATION IN CONTROL OF PKMS

Furthermore, $V(s)$ is positive definite and radially unbounded by the following [Moreno and Osorio, 2008]:

$$
\frac{1}{2}\lambda_{\min}(P) ||\xi||^2 \leq V(s) \leq \frac{1}{2}\lambda_{\max}(P) ||\xi||^2
$$

(3.75)

where $||\xi||^2 = |s|^2 + \omega^2$ is the square of the Euclidean norm of $\xi$. $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ are the minimum and maximum eigen values respectively of any matrix $A$. Making use of (3.75), (3.73), and the fact that

$$
|s|^2 \leq ||\xi|| \leq \sqrt{2} V^{\frac{1}{2}} \frac{\lambda_{1/2}(P)}{\lambda_{\min}(P)}
$$

(3.76)

it can be shown that $\dot{V}$ is upper bounded by

$$
\dot{V} \leq -\gamma V^{\frac{1}{2}}
$$

(3.77)

where

$$
\gamma = \frac{\lambda_{1/2}(P) \lambda_{\min}(Q)}{\sqrt{2M \lambda_{\max}(P)}}
$$

(3.78)

The differential equation $\nu(t) = -\gamma \nu^{1/2}(t)$ for $\nu(0) = \nu_0 \geq 0$ has a solution expressed as: $\nu(t) = (\nu_0^{\frac{1}{2}} - \frac{\gamma}{2} t)^2$. Then, following the comparison principle [Khalil, 2002] that says $V(t) \leq \nu(t)$ when $V(s_0) \leq \nu_0$, one can conclude that $V(s(t))$, and therefore $s(t)$, converges to zero in finite time at most after $T = \frac{2V^{\frac{1}{2}}(s_0)}{\gamma}$. To this end, the proof is concluded.

\[\square\]

3.4 Contribution 3: Actuator and friction dynamics formulation in control of PKMs

This contribution deals with a new dynamic formulation of parallel manipulators incorporating the actuator and friction dynamics to be utilized in control. A model-based controller, PD with computed feedforward, is proposed taking into consideration the formulated dynamics. The motivation behind this contribution is to enhance the control performance by compensating the unfavorable nonlinearities abundant extensively in PKMs.
Those nonlinearities may increase considerably when operating at high-speed motions. The proposed feedforward dynamic part relies on the reference trajectories instead of the measured ones, improving the control performance and the computational efforts. This section covers the stability of the proposed control based on a Lyapunov function candidate showing a global asymptotic convergence of the tracking error.

3.4.1 Motivation

In robotic manipulators, the accomplishment of any operation task requires the execution of a specific motion prescribed to the manipulator’s end-effector. The motion is driven by joint actuators fed with the suitable control signals that are delivered by the controller based on the desired trajectory and the measurements. The correct regulation of the controller needs an accurate analysis of the robot mechanical structure, actuators and sensors.

The actuator and friction dynamics play a significant role in enhancing the tracking performance of the robot manipulator especially in the case of parallel robots. The mechanical components of the actuator (inertia and damping coefficients) have high impact on the dynamic performance of the parallel manipulator particularly when operating at high-speed motions. Moreover, friction dynamics were investigated a lot in the literature showing complexity and difficulty in estimating its parameters. For this reason, it is mostly ignored while deriving the dynamic model unless some works where it was taken into account. However, it is featured with uncertainties and time-varying nature being highly effective on the dynamic performance of the parallel manipulator especially for low-speed and high-precision applications.

The mechanical model of the actuator dynamics including the inertia and the damping coefficients has been incorporated within an adaptive tracking control of a serial robotic manipulator in [Cheah et al., 2006]. Experimental results proved the ability of the end-effector to track a desired trajectory with the uncertain parameters. The good performance of the controller has been illustrated and verified.

In [Grotjahn et al., 2004; Shang et al., 2008], nonlinear friction models were incorporated to dynamic control strategies implemented to parallel manipulators experimentally.
The efficiency and capability of the proposed algorithms have been proved and validated. Moreover, the tracking accuracy of the parallel manipulators has been improved obviously with the nonlinear friction compensation.

A robust nonlinear control equipped with a friction estimator has been implemented to a 6-DOF parallel manipulator in the Cartesian space in [Kim et al., 2005]. Real-time experiments were investigated showing better tracking performances with the friction observer under the uncertain friction property. A nonlinear friction model has been designed and included within an augmented PD control employed on a 2-DOF parallel manipulator in [Zhang et al., 2007]. The experimental results show that, with forward dynamic compensation, the augmented PD controller can improve the tracking performance of the parallel manipulator over the simple PD controller.

In [Shang et al., 2010], the friction dynamics of the actuated joints have been enclosed within dynamic controller for a 2-DOF parallel manipulator. The weighted least square method was applied to estimate the friction parameters. The dynamic control experiments based on the identified model with the estimated parameters were implemented to the parallel manipulator. The tracking accuracy of the identified model show better results compared to the so-called nominal model.

### 3.4.2 Proposed compensation technique

#### 3.4.2.1 Actuator mechanical dynamics

Thanks to the proven effectiveness of the dynamic controllers incorporating the actuator dynamics, we propose to extend the classical dynamic model of parallel robots in (1.3) by a full mechanical model of the actuator. The intend behind this proposition is to improve the inverse dynamic model to be used within dynamic control strategies of parallel manipulators. Thus, better dynamic performance can be expected in terms of high-speed motions, precision and robustness towards uncertainties and nonlinearities.

Robotic manipulators can be driven in general by electric actuators or hydraulic ones. In this thesis, we are concerned only with electric actuators knowing that all our experimental prototypes are electrically actuated. Mostly, the used electric actuators in robotic
manipulators are permanent-magnet DC motors controlled by current mode amplifiers. The dynamic equation describing the rotational motion of such motors can be written as follows [Lewis et al., 2004]:

\[
J \ddot{q}_m + B \dot{q}_m = \Gamma_m - R_G \Gamma
\] (3.79)

where \(\Gamma_m \in \mathbb{R}^n\) is the actual requested torque from the motors by the controller, \(\Gamma \in \mathbb{R}^n\) is the output torque of the motors at the level of the jointed links to the shafts, \(J \in \mathbb{R}^{n \times n}\) is a diagonal matrix representing the total inertia of the actuators and the linked load to the rotors (rear arm), \(B \in \mathbb{R}^{n \times n}\) is a diagonal matrix denoted to the damping coefficient in the rotors of the actuators, \(\dot{q}_m, \ddot{q}_m \in \mathbb{R}^n\) are the angular velocities and accelerations of the motor shafts respectively, and \(R_G \in \mathbb{R}^{n \times n}\) is a diagonal matrix representing the ratios of the gears equipped with motors.

Indeed, the requested torque from the motor \(\Gamma_m\) by the drive controller is not the same as the output torque \(\Gamma\). The torque vector \(\Gamma\) is defined as the desired torque needed to manipulate the mechanical structure of the robot at the level of the rear-arms. The internal actuator dynamics appear as a dynamic load in addition to the dynamics of the robot as illustrated in Figure 3.4. The torque and angle relations of the gear reduction ration can be expressed as follows:

\[
\Gamma_C = R_G^{-1} \Gamma_m
\] (3.80)

\[
q = R_G q_m
\] (3.81)

where \(\Gamma_C\) is the control input vector including the actuator dynamics and \(q\) is the actuated joint angle at the level of the linked load. In the case of direct drive actuators where no gearbox is used, the gear ratio is subjected to one (\(R_G = I\), where \(I\) is the identity matrix).

Using the classical dynamics of parallel manipulators in (1.3), the actuator dynamics in (3.79), and the relations (3.80), (3.81), one can reformulate the extended dynamic model with actuator dynamics as follows:

\[
\left( J_0 + M(q) \right) \ddot{q} + \left( B_0 + C(q, \dot{q}) \right) \dot{q} + G(q) = \Gamma_C
\] (3.82)
3.4. CONTRIBUTION 3: ACTUATOR AND FRICTION DYNAMICS FORMULATION IN CONTROL OF PKMS

Figure 3.4 – Illustrative diagram of the effect of the mechanical actuator dynamics.

where $J_0$ and $B_0$ are the diagonal matrices of the inertia and damping coefficients respectively including the gear reduction ratio as follows:

$$
J_0 = \text{diag}(j_1/r_1^2, ..., j_n/r_n^2) \\
B_0 = \text{diag}(b_1/r_1^2, ..., b_n/r_n^2)
$$

(3.83)

with $j_i$, $b_i$, and $r_i$ being the inertia, damping coefficient, and gear ratio for the single actuator knowing that $i = 1, ..., n$.

The inertia and damping coefficients are mostly given by the motor manuals and there is no need to perform an identification technique.

3.4.2.2 Actuated joints friction

In this section, we propose to incorporate the friction dynamics of the actuated joints within the closed-loop control design aiming at better dynamic performance and tracking accuracy. Knowing that most of the parallel manipulator structures (as well as our prototypes) possess self-lubricating passive joints, that make their friction parameters much smaller than those of the actuated joints, the friction of the passive joints is ignored in this proposition.

The main effective types of friction are respectively the Coulomb and viscous frictions [Olsson et al., 1998]. The active joint friction dynamics can be expressed in terms of the angular velocity $\dot{q}$ as follows [Zhang et al., 2007]:

$$
F(\dot{q}) = F_c \text{sign}(\dot{q}) + F_v \dot{q} + D
$$

(3.84)
where $F_c, F_v \in \mathbb{R}^{n \times n}$ are the diagonal matrices of the corresponding Coulomb and viscous friction coefficients respectively and $D \in \mathbb{R}^n$ is the vector representing the zero-drift coefficients of the control board.

The Coulomb friction occurred when two objects undergoes relative motion and slides against each other. It works on converting the kinetic energy into thermal energy or heat. The Coulomb friction force is proportional to load, opposes the direction of motion, and is independent of contact area. Figure 3.5 shows an illustration of the behavior of the Coulomb friction compensating force with respect to the sliding velocity.

![Coulomb friction compensating force vs Sliding velocity](image)

**Figure 3.5** – Illustrative diagram of Coulomb friction force in terms of the sliding velocity.

The viscous friction results from the relative motion between the fluid and the moving boundaries / plates. It acts against the motion of any solid body through the fluid. It occurs in the cases when there is an oil between the contact surfaces, which reduces friction coefficient $F_v$. Note that this coefficient is lower than Coulomb friction coefficient $F_c$. The dynamic relation between the viscous friction force and the sliding velocity is illustrated in Figure 3.6.

The zero-drift coefficient is a gradual change in the scale zero of a measuring instrument due to factors such as time, line voltage, or ambient temperature effects. Drift is an indication of the loss of perfect repeatability.
Combining the friction model in (3.84) to the last obtained dynamic model of parallel manipulators in (3.82), one can obtain a joint space dynamic formulation with the actuator and friction parameters as follows:

\[
(J_0 + M(q))\ddot{q} + (B_0 + C(q, \dot{q}))\dot{q} + G(q) + F(\dot{q}) = \Gamma
\]

\[(3.85)\]

3.4.2.3 Proposed control solution

In order to test the efficiency of the formulated model including the actuator and friction dynamics in control, a PD controller with computed feedforward is proposed. The main advantage of such controller is that the heavy computational dynamic part can be computed offline based on the desired trajectory. Thus, the controller becomes simple and easy for implementation as well as its computational effort is reduced. The control equation including the actuator and friction dynamics is given in joint space as follows:

\[
\Gamma = K_p e + K_d \dot{e} + (J_0 + M(q_d))\ddot{q}_d + (B_0 + C(q_d, \dot{q}_d))\dot{q}_d + G(q_d) + F(\dot{q}_d)
\]

\[(3.86)\]

where \(K_p\) and \(K_d\) are two diagonal positive definite matrices representing the proportional and derivative gains respectively. The tracking error being considered in joint space as \(e = q_d - q\).
CHAPTER 3. PROPOSED CONTROL SOLUTIONS

For practical reasons and control stability proof, signum function of the Coulomb friction dynamics is approximated by a hyperbolic tangent one. Indeed, the singularity at zero velocity of the direct Coulomb friction modeling can lead to a non-smooth discontinuous control force as well as to computational burden [Duan and Singh, 2006; Pennestri et al., 2016]. A hyperbolic tangent model ensures that acceleration is continuous and smooth, so the jerk of the system is also continuous. Moreover, it can guarantee an asymptotic stability and a smooth control signal simultaneously [Cai and Song, 1993; Song et al., 1998].

Then, the control equation can be re-formulated as follows:

$$\Gamma = K_p e + K_d \dot{e} + M'(q_d) \ddot{q}_d + C'(q_d, \dot{q}_d) \dot{q}_d + G(q_d) + F_c \tanh(\dot{q}_d) + D$$  \hspace{1cm} (3.87)

with $M'(q_d) = J_0 + M(q_d)$, $C'(q_d, \dot{q}_d) = B_0 + F_v + C(q_d, \dot{q}_d)$, and $\tanh(\dot{q}_d) = [\tanh(\dot{q}_{1d}), ..., \tanh(\dot{q}_{n_d})]^T$ is the vectorial hyperbolic tangent of $\dot{q}_d$.

Substituting the control law (3.87) in the dynamic model (3.85) gives the closed loop system equation as follows:

$$M'\ddot{e} = -K_p e - K_d \dot{e} - C'\dot{e} - h$$  \hspace{1cm} (3.88)

where $M' \equiv M'(q)$, $C' \equiv C'(q, \dot{q})$ and $h$ being the residual dynamics expressed as follows:

$$h \equiv h(q, \dot{q}) = \left[ M'(q_d) - M'(q) \right] \dot{q}_d + \left[ C'(q_d, \dot{q}_d) - C'(q, \dot{q}) \right] \dot{q}_d + \left[ G(q_d) - G(q) \right] + F_c \left[ \tanh(\dot{q}_d) - \tanh(\dot{q}) \right]$$  \hspace{1cm} (3.89)

Note that the boundedness properties of the inertia matrix and Coriolis and centrifugal matrix addressed in section 1.7.1 (Property 1 and Property 2) are still valid even after adding the actuator and friction dynamics.

### 3.4.2.4 Stability analysis

**Theorem 3.4.1.** Assuming that the desired velocity and acceleration are upper bounded, the joint position and velocity tracking errors of a robotic manipulator of dynamic model (3.85), incorporating actuator parameters, viscous friction, and Coulomb friction, follow a global asymptotic convergence under the PD control with computed feedforward given by (3.87), with the proper choice of $K_p$ and $K_d$.  


3.4. CONTRIBUTION 3: ACTUATOR AND FRICTION DYNAMICS FORMULATION IN CONTROL OF PKMS

Proof. In order to study the stability analysis of the proposed controller, consider the Lyapunov function candidate below [Kelly et al., 2005]:

\[ V(t, e, \dot{e}) = \frac{1}{2} \dot{e}^T M'(q) \dot{e} + \frac{1}{2} e^T K_p e + \gamma \tanh(e)^T M'(q) \dot{e} \]  \hspace{1cm} (3.90)

where \( \tanh(e) \in \mathbb{R}^n \) is a vectorial hyperbolic tangent function of the position error and \( \gamma \) is a positive constant. The positivity and boundedness of the suggested Lyapunov function were proven in [Kelly et al., 2005].

Using Property 2 (Section 1.7.2 of Chapter 1) which provides the skew-symmetry feature of \( \frac{1}{2} \dot{M}' - C' \) (equations (1.9)), the time derivative of the Lyapunov function along the trajectories of the closed-loop system gives the following:

\[ \dot{V}(t, e, \dot{e}) = -\dot{e}^T K_d \dot{e} + \gamma \dot{e}^T \text{Sech}^2(e)^T M'(q) \dot{e} - \gamma \tanh(e)^T K_p e \\
- \gamma \tanh(e)^T K_d \dot{e} - \gamma \tanh(e)^T C'(q, \dot{q})^T \dot{e} \\
- \dot{\dot{e}}^T h(q, \dot{q}) - \gamma \tanh(e)^T h(q, \dot{q}) \]  \hspace{1cm} (3.91)

where the squared hyperbolic secant \( \text{Sech}^2(.) \) is the derivative of the hyperbolic tangent \( \tanh(.) \). Now, an upper bound is needed for the derivative of the Lyapunov function in terms of the system states \( e, \dot{e} \) to establish the stability analysis.

One can bound \( \dot{V} \) from the upper side as follows:

\[ \dot{V}(t, e, \dot{e}) \leq -\dot{e}^T K_d \dot{e} + \gamma \dot{e}^T \text{Sech}^2(e)^T M'(q) \dot{e} - \gamma \tanh(e)^T K_p e \\
+ \gamma |\tanh(e)^T K_d \dot{e}| + \gamma |\tanh(e)^T C'(q, \dot{q})^T \dot{e}| \\
+ |\dot{\dot{e}}^T h(q, \dot{q})| + \gamma |\tanh(e)^T h(q, \dot{q})| \]  \hspace{1cm} (3.92)

The vectorial hyperbolic tangent and secant functions can be bounded trivially for any vector \( x \in \mathbb{R}^n \) as follows:

\[ ||\tanh(x)|| \leq ||x|| \quad \text{and} \quad ||\tanh(x)|| \leq \sqrt{n} \]  \hspace{1cm} (3.93)

\[ ||\text{Sech}^2(x)|| \leq \sqrt{n} \]  \hspace{1cm} (3.94)
CHAPTER 3. PROPOSED CONTROL SOLUTIONS

Using the two facts $|x^Ty| \leq ||x|| ||y||$ and $|x^TAy| \leq ||x|| ||A|| ||y||$ for any vectors $x, y \in \mathbb{R}^n$ and matrix $A \in \mathbb{R}^{n \times n}$, the bounds of $\tanh(.)$ and $\text{Sech}(.)$ in (3.93) and (3.94) respectively, and boundedness property of the Coriolis and centrifugal matrix (1.8), one can develop (3.92) to the form below:

$$
\dot{V}(t, e, \dot{e}) \leq -\lambda_{\min}\{K_d\}||\dot{e}||^2 + \gamma \lambda_{\max}\{M'\}||\dot{e}||^2 - \gamma \lambda_{\min}\{K_p\}||\tanh(e)||^2
$$

$$
+ \gamma \lambda_{\max}\{K_d\}||\dot{e}|| ||\tanh(e)||
$$

$$
+ \gamma k_1 ||\dot{e}|| ||q_d|| ||\tanh(e)|| + \gamma \sqrt{n} k_{C_1} ||\dot{e}||^2
$$

$$
+ ||\dot{e}|| ||h(q, \dot{q})|| + \gamma ||\tanh(e)|| ||h(q, \dot{q})||
$$

(3.95)

where $\lambda_{\min}\{A\}, \lambda_{\max}\{A\}$ are the minimum and maximum eigen values respectively for any matrix $A$.

There exist two positive constants $k_{h_1}, k_{h_2} > 0$ such that the norm on the residual dynamics $||h(q, \dot{q})||$ can be upper bounded by the following (see proof in Appendix A):

$$
||h(q, \dot{q})|| \leq k_{h_1} ||\dot{e}|| + k_{h_2} ||\tanh(e)||
$$

(3.96)

With the use of (3.89), $\dot{V}$ can be then upper bounded as follows:

$$
\dot{V}(t, e, \dot{e}) \leq -c_1 ||\dot{e}||^2 - c_2 ||\tanh(e)||^2 + c_3 ||\dot{e}|| ||\tanh(e)||
$$

(3.97)

where

$$
c_1 = \lambda_{\min}\{K_d\} - \gamma \lambda_{\max}\{M'\} - \gamma \sqrt{n} k_{C_1} - k_{h_1}
$$

$$
c_2 = \gamma \lambda_{\min}\{K_p\} - \gamma k_{h_2}
$$

(3.98)

$$
c_3 = \gamma \lambda_{\max}\{K_d\} + \gamma k_{C_1} ||q_d||_M + k_{h_2} + \gamma k_{h_1}
$$

where $||q_d||_M > 0$ is an upper bound of the desired velocity. Then, (3.97) can be re-written in the form below:

$$
\dot{V}(t, e, \dot{e}) \leq -\left(||\dot{e}|| ||\tanh(e)||\right) \left(\begin{array}{cc} c_1 & -c_3/2 \\ -c_3/2 & c_2 \end{array}\right) \left(||\dot{e}|| ||\tanh(e)||\right)
$$

(3.99)

$$
\dot{V}(t, e, \dot{e}) \leq -z^TQz
$$
3.4. CONTRIBUTION 3: ACTUATOR AND FRICTION DYNAMICS FORMULATION IN CONTROL OF PKMS

In order to have $\dot{V}(t, e, \dot{e})$ globally negative definite, matrix $Q$ should be positive definite. A positive definite matrix must have strictly positive determinant ($\det(Q) > 0$) and diagonal components ($Q_{ii} > 0$) according to Sylvester’s theorem. Thus, the control gain matrices $K_p, K_d$ need to be chosen in a way that satisfies the following inequalities:

\begin{align}
  c_1 &> 0 \\
  c_2 &> 0 \\
  c_2^2 &< 4c_1c_2
\end{align}

(3.100)

Following direct Lyapunov theorem of global asymptotic stability [Kelly et al., 2005], and having a globally positive definite Lyapunov function $V(t, e, \dot{e}) > 0$ such that its time derivative is globally negative definite $\dot{V}(t, e, \dot{e}) < 0$, the global asymptotic stability of the closed-loop system (3.88) is verified as follows:

\[ \|z\| \to 0 \quad \text{as} \quad t \to \infty \]  
(3.101)

leading to

\[ \|e\|, \|\dot{e}\| \to 0 \quad \text{as} \quad t \to \infty \]  
(3.102)

and the proof is concluded.

3.4.3 Friction parameters identification

The identification technique applied in this study is based on the method of Least Squares Estimation which is a procedure to determine the best fit line to a given data. The proof uses simple calculus and linear algebra. This identification goes in an offline mode based on the desired trajectory and some real data (position, velocity, acceleration, and input). To identify the parameters of the friction model in (3.84), a sequence of steps is performed using the experimental testbed and the Matlab/Simulink environment.

1. Apply on the experimental parallel robot prototype a simple PD controller fed with a desired reference trajectory in a nominal scenario (used also in the PD control with computed feedforward (3.86) experiments).
2. Get out the generated control input $\Gamma_{PD}$ and the measured signal $q$.

3. Estimate from the measured angles $q$ the angular velocity and acceleration $\dot{q}, \ddot{q}$.

4. Compute using the available data the dynamic model including the actuator dynamics (3.82): $H = \left( J_0 + M(q) \right) \ddot{q} + \left( B_0 + C(q, \dot{q}) \right) \dot{q} + G(q)$.

5. Evaluate the friction dynamic model using the following equation:

$$Y \equiv F(\dot{q}) = H - \Gamma_{PD} \quad (3.103)$$

6. Over $N$ sample times, for each $i$ the actuator, one can formulate:

$$\begin{bmatrix} Y_{i,1} \\ Y_{i,2} \\ \vdots \\ Y_{i,N} \end{bmatrix} = \begin{bmatrix} \dot{q}_{i,1} \\ \dot{q}_{i,2} \\ \vdots \\ \dot{q}_{i,N} \end{bmatrix} \begin{bmatrix} \tanh(\dot{q}_{i,1}) & 1 \\ \tanh(\dot{q}_{i,2}) & 1 \\ \vdots & \vdots \\ \tanh(\dot{q}_{i,N}) & 1 \end{bmatrix} \begin{bmatrix} f_{v_i} \\ f_{c_i} \\ d_i \end{bmatrix} \quad (3.104)$$

where $f_{v_i}, f_{c_i}, d_i \in \mathbb{R}$ are the corresponding viscous, coulomb, and zero-drift parameters respectively for each $i$th actuator. (3.104) can be displayed in a compact form as follows:

$$Y = AX \quad (3.105)$$

7. Apply the Least Square Estimation method [Miller, 2006] to identify $f_{v_i}, f_{c_i}, d_i$ for each $i$th actuator as follows:

$$X = (A^T A)^{-1} A^T Y \quad (3.106)$$

Figure 3.7 summarizes the identification procedure in a block schema.

### 3.5 Redundantly actuated PKMs: Elimination of antagonistic internal forces

As discussed before, actuation redundancy in parallel manipulators can be achieved by adding additional actuated kinematic chains. It holds several advantages to parallel
3.5. REDUNDANTLY ACTUATED PKMS: ELIMINATION OF ANTAGONISTIC INTERNAL FORCES

Figure 3.7 – Block diagram of the Least Squares Estimation method applied to identify the friction parameters.

Manipulators such as it eliminates singularities and, thus, increase the workspace, reduces the energy consumption, and adds more stiffness to the structure. However, it results some antagonistic forces that have no effect on the motion of the manipulator appear as internal forces called prestress [Mueller, 2009].
3.5.1 Basic concept

The motion equation of a redundantly actuated PKM of \( n \) actuators and \( m \)-DOFs can be written the following form [Mueller, 2009]:

\[
\bar{M}(q)\ddot{q} + \bar{C}(q, \dot{q})\dot{q} + \bar{G}(q) = J_m^T \Gamma 
\]  

(3.107)

where \( q \) is the overall vector of the joint coordinates (passive and actuated), \( q_2 \) is the \( m \)-independent generalized coordinates (considered here the Cartesian coordinates), \( \bar{M} \) is the generalized mass matrix, \( \bar{C} \) is the generalized Coriolis matrix, and \( \bar{G} \) represents all non-inertial forces including end-effector loads, and \( J_m \) being the inverse Jacobian matrix that describes the relevant part of the generalized control forces which are the control forces in the actuated joints \( \Gamma \).

For redundantly actuated PKMs, \( J_m \in \mathbb{R}^{n \times m} \) is full rank \( m \) unless at singularity configurations. Let the degree of redundancy be \( \rho = n - m \), then \( J_m^T \) has a \( \rho \)-dimensional kernel. This means that (3.107) has no unique solution for control force \( \Gamma \). Thus, only those control input forces that are not in the kernel of \( J_m^T \) are effective on the structure of the PKM \( (J_m^T \Gamma \neq 0) \). While the actuator forces that belong to \( \rho \)-dimensional null space of \( J_m^T \) have no effect on the motion and appear as internal forces. Moreover, the load distribution overall the drives is not unique.

3.5.2 Effect of measurement errors

The effect of measurement errors on both decentralized and model-based control schemes was addressed in [Hufnagel and Muller, 2012]. The generated control forces that have no effect on the motion due to actuation redundancy may deteriorate the performance in the presence of measurements errors. A good example that illustrates the effect of those errors was introduced in [Hufnagel and Muller, 2012] on the linear PD control scheme. Consider the measured tracking actuated joint error with measurement imperfections as follows:

\[
\hat{e} = \hat{q} - q_d = (q + \Delta q) - q_d = e + \Delta q
\]  

(3.108)
3.5. REDUNDANTLY ACTUATED PKMS: ELIMINATION OF ANTAGONISTIC INTERNAL FORCES

where $\Delta q$ represents the imperfections in the measurement. The PD control fed with the measured tracking error and generating the necessary control forces can be given as follows:

$$\Gamma = -K_p \tilde{e} - K_d \dot{\tilde{e}}$$  \hspace{1cm} (3.109)

where $K_p$ and $K_d$ are diagonal positive definite gain matrices. Considering that PKM attains a setpoint reference position and stays stationary, the effect of the control forces belonging to the null space of $J_m^T$ is expressed as follows:

$$0 = J_m^T \Gamma^0 = -J_m^T K_p \tilde{e} = -J_m^T K_p (e + \Delta q)$$  \hspace{1cm} (3.110)

where $\Gamma^0$ are the generated control forces in the null space of $J_m^T$. It is clear from (3.110) that in case of perfect measurement and geometric model ($\Delta q = 0$), the tracking error converges to zero. However, when measurement errors and geometric imperfections exist, the tracking error doesn't converge to zero but a value dependant on those imperfections.

3.5.3 Elimination of antagonistic internal forces

To this end, it has been shown that the internal forces are caused by the generated control forces that full in the null space of the inverse Jacobian matrix. Hence, the antagonistic forces can be eliminated by projecting the control forces vector into the range space of the inverse Jacobian matrix as follows [Hufnagel and Muller, 2012]:

$$\Gamma^* = R_{J_m^T} \Gamma$$  \hspace{1cm} (3.111)

where $\Gamma^*$ is the effective control forces applied to the redundant parallel manipulator and $R_{J_m^T} = J_m^T + J_m^T$ is the projection matrix, called also the regularization matrix. Note that this projection method does not change the drive action since $J_m^T \Gamma^* = J_m^T \Gamma$. Indeed, the antagonistic forces are projected to the null space of $J_m^T$ using the regularization matrix $R_{J_m^T}$ as
follows:
\[
\begin{align*}
\Gamma^* &= R_{J_m} \Gamma \\
&= \left( I - (I - J_m^T J_m^T) \right) \Gamma = \left( I - N_{J_m} \right) \Gamma \\
&= \Gamma - N_{J_m} \Gamma = \Gamma - \Gamma^0
\end{align*}
\]

with \( N_{J_m} = I - J_m^T J_m \) being the projection matrix to the null space. Thus, from (3.112), the generated control forces \( \Gamma \) can be decoupled into effective forces \( \Gamma^* \) fed in the actuators and antagonistic forces \( \Gamma^0 \) eliminated by projecting them to the null space of inverse Jacobian matrix \( \Gamma = \Gamma^* + \Gamma^0 \).

In this thesis, for the redundantly actuated parallel manipulators (Spider4 robot), the generated control input, from a proposed control solution, is treated by the projection method introduced above before feeding the actuators. Precisely, the control input vector \( \Gamma \) is projected to the range space of the inverse Jacobian matrix as in (3.111), and the obtained effective control input \( \Gamma^* \) enters then the actuators.

### 3.6 Conclusion

This chapter was dedicated for presenting and detailing the proposed control solutions of this thesis. The main objective was to design robust and performant control strategies capable of compensating the errors coming from the controller itself, the actuator dynamics, the friction in the articulations, the system nonlinearities, the external disturbances, the measurements noise, etc.

The main contributions can be mentioned briefly as follows:

1. A new time-varying feedback Robust Integral of the Sign of the Error (RISE) control strategy was developed for parallel manipulators. Replacing some static feedback gains in the original RISE controller may lead to more robustness towards disturbances, dynamic changes, and uncertainties.

2. A novel model-based super-twisting sliding mode control was proposed as an extension of the original second order super-twisting algorithm. The control structure
3.6. CONCLUSION

comprises a feedforward dynamic term, the standard super-twisting algorithm, and a feedback stabilizing term. This may lead to high dynamic performances in terms of precision, robustness towards operating condition changes, and disturbance-rejection.

3. An actuator and friction dynamics formulation was proposed within a model-based control strategy. Compensating for the errors coming from actuator dynamics and friction in the articulations can improve the global dynamic performance of parallel manipulators.

In the next chapter, the experimental results obtained from the proposed control solutions will be presented and discussed.
4
Real-time experiments and results

Contents

4.1 Introduction ........................................... 145
4.2 Experimental platforms and implementation issues .............. 146
4.3 Experimental results of contribution 1: Time-varying feedback RISE control 159
4.4 Experimental results of contribution 2: Model-based ST-SMC algorithm 168
4.5 Experimental results of contribution 3: PDFF control with actuator and friction dynamics ........................................... 186
4.6 Conclusion ............................................... 194

4.1 Introduction

In this chapter, the experimental results obtained by the proposed control solutions are demonstrated and discussed. The experimental setups and implementation issues of the existing parallel robot prototypes are introduced. Three PKM prototypes (presented in Chapter 2) are used for the control validation during this thesis: a 3-DOF Delta robot, at EPFL, Switzerland, a 5-DOF SPIDER4 robot and a 4-DOF VELOCE robot at LIRMM.

Delta robot is introduced as experimental platform for the control validation of Contri-
Contributions 1 and 2 (time-varying feedback RISE and feedforward super-twisting SMC). Moreover, SPIDER4 robot is proposed for the experimental validation of Contribution 2. For Contribution 3 (actuator and friction dynamics formulation), VELOCE robot is adopted as an experimental validation setup.

Different scenarios are conducted for each experimental test such as nominal case, robustness towards payload changes, and robustness towards speed changes. The purpose behind these scenarios is to test our proposed controllers at different dynamic operating conditions. The results for each experimental test are plotted, clarified, and discussed in terms of the dynamic performance of the parallel manipulator. Finally, this chapter ends up with a conclusion regarding the proposed control solutions and the obtained results.

4.2 Experimental platforms and implementation issues

This section provides a description about the experimental testbeds that were used to validate our proposed controllers. The trajectory generation for each parallel robot prototype is introduced. Three platforms will be exposed in the sequel: 3-DOF Delta robot, 5-DOF SPIDER4 robot, and 4-DOF VELOCE robot.

4.2.1 Experimental testbed of the 3-DOF Delta robot

The Delta parallel robot used for the real-time experiments is shown in Figure 4.1. It is located at Robotics Systems Laboratory, EPFL, Switzerland. Three direct-drive motors integrated with the fixed-base allow the motion of the kinematic chains generating three translational movements of the traveling-plate in x, y and z axes. Each motor can deliver a maximum torque of 23 Nm. The overall mechanical structure can reach up 50 G as peak acceleration.

The control algorithms are implemented in C++ language level using Visual Studio software from Microsoft, running on a Windows XP operating system. RTX extension is used to establish the real-time communication. The internal timer (HAL timer) of RTX is configured to 100 µs in which the control loop is set to 10 times this value for synchronization, leading to a sample time 1 ms, and a sampling frequency of 1 KHz.
4.2. EXPERIMENTAL PLATFORMS AND IMPLEMENTATION ISSUES

4.2.1.1 Reference trajectory generation

The motion control profile used for Delta robot is the point-to-point move. It means that from a stop point, the robot accelerates to a constant velocity. Then, the robot decelerates such that the final acceleration, and velocity, are zero at the final desired point.

An S-curve velocity profile is adopted for Delta robot such that the velocity increasing and decreasing phases are not linear but S-curved. Thus, the rapid change in the acceleration will be smooth (linear and not instant switching) and the vibration of the mechanical system will be reduced. An illustrative plot is shown in Figure 4.2 of the S-curve motion profile of a point-to-point move.

For each phase of motion in Figure 4.2, the continuous form equation used to compute the position variation with time is given as follows:

\[ x_d = p_i + v_i(t - t_i) + \frac{1}{2}a_i(t - t_i)^2 + \frac{1}{6}j(t - t_i)^3 \]  \hspace{1cm} (4.1)

where \( x_d \) is the obtained desired trajectory, \( t_i, p_i, v_i, a_i, j \) are the corresponding initial time, initial position, initial velocity, initial acceleration, and the desired jerk (time rate of change of acceleration) respectively for each phase. The velocity and acceleration profiles can be obtained by deriving equation (4.1) with respect to time.
Figure 4.2 – Point-to-point S-curve profile motion illustration.

The reference trajectory is generated using semi-ellipse geometric motions producing a pick-and-place trajectory in Cartesian space. This trajectory is mainly used in industry for food packaging applications. A 3D illustrative view is shown in Figure 4.3 for the pick-and-place cycle to be followed by the robot’s traveling-plate.

The desired actuated joints trajectory is computed using the inverse kinematic and differential kinematic models of Delta robot (presented in Chapter 2) having at the end the desired joints position, velocity, and acceleration.

### 4.2.2 Experimental testbed of the 5-DOF SPIDER4 robot

The real-time experimental platform of SPIDER4 robot is shown in Figure 4.4. The parallel structure of SPIDER4 robot consists of four TPM⁺ high torque rotary motors responsible of generating the three translational motions $x, y$ and $z$ at the level of the nacelle. A gearbox of gear ratio 22 is merged seamlessly to the motor forming one compact versatile unit.
The peak torque that can be delivered by the motor after the gear transformation reaches up 3100 Nm. The maximum speed for each motor after the gear can reach up to 189 rpm. The overall structure provides at the traveling-plate level a maximum speed of 2 m/s and a maximum acceleration of 4 G.

The control program of SPIDER4 robot is established within Matlab/Simulink environment provided from MathWorks. Using the library of B&R automation studio, a target for Simulink tool compiles the code and converts it to C/C++ environment which can be accessible from Automation Studio software provided by B&R Perfection in Automation. The trajectory generation process is done using the numerical control programming language named G-code (Geometric Code) used mostly with CNC machines (Computer Numerical Control machines). The experimental testbed of SPIDER4 robot is exposed in Figure 4.5.

The overall program is then downloaded to an industrial PC (APC910) from B&R responsible of communication with the motion control system. The operational clock cycle of its processor is 2.5 KHz leading to sampling period of 0.4 ms. The motion control of SPIDER4 robot consists of X20 modules and inverter modules named ACOPOSmulti system, the new drive generation from B&R Perfection in Automation.
As we discussed before (refer to Chapter 2), we are concerned only about the translational motion of the nacelle of SPIDER4 robot having a redundantly actuated parallel robot, four motors and 3-DOFs.

4.2.2.1 Reference trajectory generation

As said before, the trajectory generation of SPIDER4 robot uses the programming language for CNC machines known as G-Code. In general, there are three basic types of motions for a CNC machine:

1. Rapid move: a rapid move to a desired position as fast as possible.
2. Feed move: a linear move to a desired position at a defined feedrate.
3. Circular move: a circular move to a desired position at a defined feedrate.

The G-code instructions tell the machine which type of motion to follow and how to per-
form it. The generated trajectory of SPIDER4 robot considered for experimental tests employs the two types of motion: feed and circular moves.

The feed move is expressed in the G-code instructions as G1 linear interpolation. In this mode, the machine will move at a defined feedrate to the programmed point. All the axes move to simultaneously meet at the programmed point. The formula that represents the linear interpolation can be given as follows:

\[ x_d = P_i + V_i(t - t_i) \]  \hspace{1cm} (4.2)

where \( x_d \) is the obtained desired trajectory. \( t_i, P_i, V \) are the corresponding initial time, initial position, and the velocity (feedrate). An illustration of the G1 linear interpolation motion profile is depicted in Figure 4.6 showing how the exact desired path is converted to multiples of point-to-point linear motions.

The circular move is presented in the G-code instructions as G3 circular interpolation. It allows the point-to-point motion of the tool along a circular arc at a defined feedrate.
The required information to execute this command are: an endpoint, a feedrate, a center, a radius, and a direction of movement (counter-clockwise for G3 command). The basic formula used to perform the circular interpolation is given as follows:

$$R = \sqrt{(x_i + i)^2 + (y_i + j)^2 + (z_i + k)^2} \quad (4.3)$$

where $R$ is the radius of the arc, $[x_i, y_i, z_i]^T$ is the position of the starting point of the arc, and $i, j, k$ are the corresponding coordinate variations between the start point and the center of the arc. Then, knowing that $R$ is constant, a small increment is performed on $i, j, k$ to obtain the new position on the arc (after this small increment). And so on, the final point of the arc is reached after several increments and computations. For more clarification, Figure 4.7 shows a circular motion in 2D plane, between two points, executed using small increments to $x$ and $y$ coordinates.

Using the two aforementioned interpolation types (G1 linear interpolation and G3 circular interpolation), the G-Code generating the desired trajectory of SPIDER4 robot is written. 3D views of the generated trajectories are sketched in Figures 4.8 and 4.9 illustrating the points traversed by the robot within the workspace. The sequence of the interpolated points of trajectory I depicted in Figure 4.8 can be given in order as follows:

1. Linear motion: $P_0, P_1, P_2, P_3, P_4, P_5, P_1, P_6$.
2. Circular motion: from $P_0$ to $P_0$ following circle of diameter $D$.
3. Linear motion: $P_6, P_7, P_0$. 

Figure 4.6 – G1 linear interpolation motion profile.
While the sequence of the interpolated points used to generate trajectory II depicted in Figure 4.9 is given in order as follows:

1. Linear motion: \( P_0, P_1 \).
2. Circular motion: from \( P_1 \) to \( P_1 \) following circle of diameter \( D_2 \).
3. Circular motion: from \( P_1 \) to \( P_1 \) following circle of diameter \( D_1 \).
4. Linear motion: \( P_1, P_0 \).
5. Circular motion: from \( P_0 \) to \( P_0 \) following circle of diameter \( D_3 \).

The corresponding desired actuated joint trajectory is computed using the inverse kinematic model of the robot (presented in Chapter 2).

### 4.2.3 Experimental testbed of the 4-DOF VELoce robot

VELOCE PKM prototype used for experimental tests is shown in Figure 4.10. VELoce robot is equipped with four TMB0140-100-3RBS ETEL direct-drive motors that can provide a peak torque of 127 Nm and reach a maximum speed of 550 rpm. The end-effector is capable of traversing three translational movements along \( x, y, z \) axes and one rotational movement \( \theta_z \) around \( z \)-axis. The overall structure provides at the level of the end-effector a maximum speed of 10 m/s and a maximum acceleration of 20 G.
Figure 4.8 – 3D view of Trajectory I for SPIDER4 robot in Cartesian space.

Figure 4.9 – 3D view of Trajectory II for SPIDER4 robot in Cartesian space.

The control algorithm of VELOCE robot is implemented using Matlab/Simulink environment provided by MathWorks. Thanks to the real-time toolboxes from ETEL, the project is compiled and transferred to an industrial PC named XPC target from MathWorks. The XPC target operates at a clock rate of 10 KHz providing a sampling time of 0.1 ms. Then, an UltimET board connected to the AccurET drives, both from ETEL, with a TransnET cable takes the management task of the different axes of the robot. Figure 4.11 shows the full experimental setup of VELOCE robot.
4.2. EXPERIMENTAL PLATFORMS AND IMPLEMENTATION ISSUES

Figure 4.10 – View of the real VELOCE parallel robot used for real-time experiments.

Figure 4.11 – Experimental setup of VELOCE robot.

4.2.3.1 Reference trajectory generation

The motion control profile used for VELOCE robot is the point-to-point move. A fifth degree polynomial is adopted to generate a smooth motion profile of the Cartesian posi-
tion. The profiles of the Cartesian velocity and acceleration can be obtained by deriving with respect to time the position. This profile motion is governed by the equation that gives the desired position as follows:

\[ x_d = P_i + (P_f - P_i)r(t) \]

\[ r(t) = 10 \left( \frac{t-t_0}{T} \right)^3 - 15 \left( \frac{t-t_0}{T} \right)^4 + 6 \left( \frac{t-t_0}{T} \right)^5 \]  

(4.4)

where \( x_d \) is the generated trajectory between the initial position \( P_i \) and the final one \( P_f \).

This method avoids the infinite derivative of the acceleration (in case of discontinuous accelerations) which can produce an impulsive jerk in the motion of the robot. Figure 4.12 shows an illustration of a point-to-point desired trajectory using the fifth degree polynomial motion profile.

![Illustration of point-to-point motion profile](image.png)

Figure 4.12 – Point-to-point fifth degree polynomial profile motion illustration.

Using the motion profile (4.4), a desired trajectory is generated in Cartesian space stimulating the four DOFs of VELOCE robot and going over different points of the workspace. A 3D view of the motion sequences to be followed by the end-effector of VELOCE robot is shown in Figure 4.13.
Then, using the inverse kinematic model of VELOCE robot (presented in Chapter 2), the desired trajectory at the actuated joints level is computed.

### 4.2.4 Performance evaluation criteria

In order to quantify the relevance of the control algorithms, we need to define a certain performance index. One of our main objectives is to enhance the precision and increase the tracking accuracy of the parallel robots through the proposed controllers.

An accuracy evaluation tool frequently used to evaluate differences between a desired trajectory and a measured one is the Root-Mean-Square Error (RMSE) criterion. It can quantify approximately the error between the desired trajectory and the actual one traversed by the robot.

For all the used PKM prototypes, the actuated joint positions are measured using encoders equipped with the motors, and the actual Cartesian position vectors are then computed using the forward kinematic models (presented in Chapter 2).

For Cartesian space, the performance index of the translational motions is defined as
RMSE\textsubscript{T}, and that of the rotational motions is defined as RMSE\textsubscript{R}. For joint space, the performance index is defined as RMSE\textsubscript{J}. Then, the aforementioned performance indices can be given as follows:

\[
\text{RMSE}_T = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left( \sum_{j=1}^{m_1} (e_j^2(i)) \right)}
\]

\[
\text{RMSE}_R = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left( \sum_{j=1}^{m_2} (e_j^2(i)) \right)}
\]

\[
\text{RMSE}_J = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left( \sum_{j=1}^{n} (e_j^2(i)) \right)}
\]

(4.5)

where \( N \) is the number of the collected samples overall the whole trajectory, \( e \) represents the difference between the desired and measured positions (at Cartesian or joint space), \( m_1, m_2 \) are the corresponding numbers of translational and rotational motions can be performed by the end-effector respectively (\( m_1 + m_2 = \text{DOFs} \)), and \( n \) is the number of the actuators.

In order to estimate the energy consumption, the input-torques-based criterion is adopted given as follows:

\[
E_\Gamma = \sum_{j=1}^{n} \sum_{i=1}^{N} \|f_j(i)\| \]

(4.6)

where \( E_\Gamma \) is the total summation of the absolute values of the input torques delivered by the \( n \) actuators.

### 4.2.5 Tuning of the control gains

A popular method for tuning of the control gains in experiments, used for complex robotic systems, is the Trial and Error method. It is characterized by trying manually and continuously different sets of control gains in real-time framework until the desired control performance is achieved. It is used mostly when the formulated dynamic model does not exactly match the physical system, and thus the automatic numerical closed-loop tuning methods may give unsuitable control gains for real-time experiments.
4.3 Experimental results of contribution 1: Time-varying feedback RISE control

Delta robot (Figure 4.1) is used as an experimental testbed to validate the proposed time-varying feedback RISE controller. To demonstrate the effectiveness of the proposed control solution, both the original RISE (refer to (3.9)) and the proposed one (refer to (3.16)) are implemented on Delta robot. Then, a comparative study between the results of the two implementations is done. Two main scenarios are conducted for both controllers: scenario 1: nominal case and scenario 2: robustness towards payload and speed changes.

4.3.1 Tuning procedure of the control gains

4.3.1.1 Tuning of the standard RISE control gains

The tuning process of the standard RISE control gains is performed by the following simple procedure:

1. Set $\alpha_2 = 0$ and $\beta = 0$,
2. tune $\alpha_1$ and $k_s$ as if dealing with a PD controller, given that $\alpha_1(k_s + 1)$ is the proportional gain and $(k_s + 1)$ is the derivative one till a satisfied tracking is reached,
3. start increasing $\alpha_2$ with modifying again $\alpha_1$ and $k_s$ either increasing or decreasing till we reach as best performance index as possible,
4. increase $\beta$ until obtaining acceptable chattering input signal and better performance index.

Following the above procedure, the standard RISE control gains were tuned in real-time experiments, and the obtained final values are summarized in Table 4.1.

4.3.1.2 Tuning of the proposed time-varying feedback control gains

For the tuning process of the proposed time-varying feedback RISE controller, and especially tuning the nonlinear feedback gains, a similar manner for the one proposed in [Shang et al., 2009] to tune the nonlinear PD control gains is used in our case.
CHAPTER 4. REAL-TIME EXPERIMENTS AND RESULTS

The main steps of this procedure are described as follows:

1. Initialization: $\epsilon_1 = 1$, $\epsilon_2 = 1$, $\alpha_{20} = 0$, $\beta = 0$,

2. increase $\alpha_1$ and $k_{s0}$ starting both from zero until obtaining an acceptable tracking performance,

3. increase the value of $\alpha_{20}$ to get a better tracking performance, then make a trade-off between $\alpha_1, k_{s0}$, and $\alpha_{20}$.

4. find $(e_2)_{\text{max}}$ and $(\int e_2)_{\text{max}}$ values and set their halves as values of $\delta_1$ and $\delta_2$ respectively,

5. decrease the value of $\epsilon_1$ within the interval $[0.5, 1]$ and increase the value of $\epsilon_2$ within the interval $[1, 1.5]$, retune again the values of $k_{s0}$ and $\alpha_{20}$ making a compromise among the four values,

6. repeat steps 4 and 5 until obtaining the best possible RMSE,

7. increase $\beta$ until obtaining better performance index.

Based on the above tuning algorithm, the control parameters of the proposed time-varying feedback RISE controller are tuned experimentally, and the obtained final values are summarized in Table 4.1.

Table 4.1 – The control gains of the original RISE and the proposed time-varying feedback RISE controllers.

<table>
<thead>
<tr>
<th>Original RISE</th>
<th>Proposed time-varying feedback RISE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_1 = 360$</td>
<td>$\alpha_1 = 450$</td>
</tr>
<tr>
<td>$k_s = 0.35$</td>
<td>$k_{s0} = 0.35$</td>
</tr>
<tr>
<td>$\alpha_2 = 0.66$</td>
<td>$\epsilon_1 = 0.65$</td>
</tr>
<tr>
<td>$\beta = 1.5$</td>
<td>$\delta_1 = 0.05$</td>
</tr>
</tbody>
</table>

4.3.2 Scenario 1: nominal case

In this scenario, the traveling-plate of Delta robot does not carry any additional payload and the robot is operating at acceleration of 2.5 G (with a speed of 1500 mm/s).
4.3. EXPERIMENTAL RESULTS OF CONTRIBUTION 1: TIME-VARYING FEEDBACK RISE CONTROL

Following the reference trajectory shown in Figure 4.3, the Cartesian and joint tracking errors for both controllers (original RISE and proposed one) are registered and plotted in Figure 4.14. One can observe a small attenuation in the peak errors of Cartesian and joint spaces overall the reference trajectory. We can notice that replacing the static feedback gains in the standard RISE controller with nonlinear feedback gains reduces the tracking errors in all axes. The RMSE performance index is evaluated in both Cartesian and joint spaces and reported in Table 4.2. It shows an improvement of 18% in terms of tracking precision for both Cartesian and joint spaces.

Figure 4.14 – Scenario 1: Evolution of the Cartesian and joint tracking errors of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.

The generated control input torques of the three direct-drive motors of Delta robot for both controllers are depicted in Figure 4.15. It is clear that both control algorithms generate an input signal within the admissible limits of the motors. No significant improvement is observed in terms of energy consumption in this scenario.

The evolution of the nonlinear feedback gains \((K_s(.) + 1)\) and \((k_{s0} + 1)\alpha_2(.)\) overall the reference trajectory is depicted in Figures 4.16 and 4.17 respectively. The produced
Figure 4.15 – Scenario 1: Evolution of the control input torques of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.

variations of both feedback gains versus time give always strictly positive bounded values (see Figures 4.16-a and 4.17-a). Figure 4.16-b shows the behavior of \((k_s(\cdot) + 1)\) feedback gain versus the variation of the combined error \((e_2)\), which meets our expectations and desired performance. Moreover, Figure 4.17-b exposes the attitude of \((k_{s0} + 1)\alpha_2(\cdot)\) gain versus the variation of \((\int e_2)\) showing the performance in prospect, knowing that \((\int e_2)\) is always in the negative side.

Table 4.2 – Scenario 1: Control performance evaluation of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.

<table>
<thead>
<tr>
<th></th>
<th>RMSEE (_C) [mm]</th>
<th>RMSEE (_J) [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original RISE</td>
<td>0.993</td>
<td>0.2341</td>
</tr>
<tr>
<td>New time-varying feedback RISE</td>
<td>0.8129</td>
<td>0.192</td>
</tr>
<tr>
<td>Improvements</td>
<td>18.1 %</td>
<td>18 %</td>
</tr>
</tbody>
</table>

This scenario shows clearly that the proposed time-varying feedback RISE control over-
4.3. EXPERIMENTAL RESULTS OF CONTRIBUTION 1: TIME-VARYING FEEDBACK RISE CONTROL

Figure 4.16 – Scenario 1: Evolution of the nonlinear feedback gain \((K_S(.) + 1)\) of the proposed time-varying feedback RISE controllers on Delta robot.

Figure 4.17 – Scenario 1: Evolution of the nonlinear feedback gain \(((k_S + 1)\alpha_2(.))\) of the proposed time-varying feedback RISE controllers on Delta robot.

comes the original RISE control in terms of precision and performance thanks to the extended nonlinear feedback gains and their special behavior.

4.3.3 Scenario 2: robustness towards payload and speed changes

The main intended role of Delta parallel robot is performing rapid pick-and-place cycles for industrial applications such as food packaging. Accordingly, a payload of 225 g was attached to the traveling-plate of the robot accompanied by increasing the operating
acceleration to 7.5 G (with a speed of 1500 mm/s). The aim of this scenario is to test the robustness of the proposed time-varying feedback RISE towards payload and speed changes.

The same desired trajectory illustrated in Figure 4.3 is followed by the robot in this scenario. The resulted tracking errors in Cartesian and joint spaces for both controllers are displayed in Figure 4.18. Comparing to the original RISE tracking errors, a remarkable downsizing in the magnitudes of the tracking errors is obtained with the proposed time-varying feedback RISE controller. A significant improvement of 30.5 % in terms of Cartesian space accuracy is noticed when evaluating the RMSE performance index and 28.3 % for the joint space accuracy (see Table 4.3). Less oscillations are produced by the proposed time-varying feedback RISE control compared to the standard RISE control, especially in actuators 1 and 2 as shown in Figure 4.18.

![Graphs showing tracking errors](image)

Figure 4.18 – Scenario 2: Evolution of the Cartesian and joint tracking errors of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.

Figure 4.19 displays the evolution of the control input torques along the reference trajectory (shown in Figure 4.3) for both controllers. Both standard RISE and proposed time-varying feedback RISE controllers guarantee an input signal within the safety margins of
4.3. EXPERIMENTAL RESULTS OF CONTRIBUTION 1: TIME-VARYING FEEDBACK RISE CONTROL

Delta robot actuators. Nevertheless, it is worth to highlight the reduced input torques generated by the time-varying feedback RISE control compared to the standard RISE controller. In this scenario of high dynamic operating conditions, an improvement of 19.1 % in terms of energy consumption is noticed and reported in Table 4.3 using the input-torques-based criterion proposed in (4.6).

![Graph showing control input torques](image)

Figure 4.19 – Scenario 2: Evolution of the control input torques of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.

The dynamic variations of the nonlinear feedback gains along the desired trajectory are depicted in Figures 4.20 and 4.21. It is obvious from Figures 4.20-a and 4.21-a to observe that the nonlinear gains remain positive and bounded even with the high changes of the dynamic operating conditions. Figure 4.20-b displays the action of \((K_s(\cdot) + 1)\) gain versus the combined error \(e_2\) with a smaller constant zone compared to scenario 1. This explains the better performance of the proposed controller compared to the original RISE controller by the dynamic behavior of the feedback gains with the change of the operating conditions. The same holds for \((k_{s0} + 1)\alpha_2(\cdot)\) gain in Figure 4.21-b, where the nonlinear variation is increased compared to scenario 1, leading to better control performance and more robustness.
(a) \((K_s(\cdot) + 1)\) versus time.

(b) \((K_s(\cdot) + 1)\) versus \(e_2\).

Figure 4.20 – Scenario 2: Evolution of the nonlinear feedback gain \((K_s(\cdot) + 1)\) of the proposed time-varying feedback RISE controllers on Delta robot.

(a) \((k_{\alpha_0} + 1)\alpha_2(\cdot)\) versus time.

(b) \((k_{\alpha_0} + 1)\alpha_2(\cdot)\) versus \(\int e_2\).

Figure 4.21 – Scenario 2: Evolution of the nonlinear feedback gain \((K_s(\cdot) + 1)\) of the proposed time-varying feedback RISE controllers on Delta robot.

To sum up, we can say that the proposed time-varying feedback RISE control is more robust towards payload and speed variations than the original RISE algorithm thanks to the dynamic behavior of the proposed nonlinear feedback gains.
4.3. EXPERIMENTAL RESULTS OF CONTRIBUTION 1: TIME-VARYING FEEDBACK RISE CONTROL

Table 4.3 – Scenario 2: Control performance evaluation of the original RISE and the proposed time-varying feedback RISE controllers on Delta robot.

<table>
<thead>
<tr>
<th></th>
<th>RMSE_C [mm]</th>
<th>RMSE_J [deg]</th>
<th>E_Γ [Nm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original RISE</td>
<td>5.3985</td>
<td>1.2577</td>
<td>1.7692 \times 10^4</td>
</tr>
<tr>
<td>New time-varying feedback RISE</td>
<td>3.7542</td>
<td>0.9012</td>
<td>1.4318 \times 10^4</td>
</tr>
<tr>
<td>Improvements</td>
<td>30.5 %</td>
<td>28.3%</td>
<td>19.1 %</td>
</tr>
</tbody>
</table>

4.3.4 Performance index versus operating acceleration

In this section, the operating acceleration is increased gradually starting from 2.5 G reaching up 10 G. Both controllers have been tested in the same scenarios: with and without additional payload (225 g).

Figures 4.22-a and 4.22-b are two bar graphs showing the variation of the Cartesian RMSE in (mm) with respect to the operating acceleration (G) in case of no added payload and payload of 225 g respectively.

The quantified improvement of the new time-varying feedback RISE controller at each acceleration is written at the top of the corresponding column. It can be clear that the performance of time-varying feedback RISE is better than that of standard RISE in all cases. However, the gathered improvements of the proposed controller are much better in the case of added payload than that of no payload.

It is verified that the proposed nonlinear control law based on time-varying feedback gains is much appropriate for nature of PKMs especially when operating at high dynamics such as payload and acceleration. It is noticeable that at acceleration of 10 G in the case of added payload, the generated joint errors override 10 degrees, the specified safety margins for the robot to turn off, with RISE controller. While time-varying feedback RISE controller produces acceptable errors always within the defined safety margins.
Figure 4.22 – Clustered column chart of RMSE_C of the original RISE and the proposed time-varying feedback RISE controllers versus operating acceleration on Delta robot.

4.4 Experimental results of contribution 2: Model-based ST-SMC algorithm

In order to validate the proposed feedforward super-twisting sliding mode control, two PKM prototypes are used as experimental platforms: SPIDER4 robot, the redundantly actuated PKM, and Delta robot (Figures 4.4 and 4.1 respectively). A Comparative study is done between the proposed control approach and the conventional ones on both experimental setups. This section provides all the necessary experimental issues as well as the obtained results on each prototype.

4.4.1 Experimental results of contribution 2 on SPIDER4 robot

In this section, the experimental results of the conventional Computed-Torque ST-SMC (CT-ST-SMC) algorithm (of control equation (3.59)) and the proposed FeedForward ST-SMC (FF-ST-SMC) (of control equation (3.60)) are demonstrated in comparison with the standard PID with computed FeedForward (PIDFF) control.

The control gains of the three controllers are tuned in real-time framework. The same gains are considered for the four axes of SPIDER4 robot because of the symmetric struc-
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

ture of the robot and the same type of motors is used to actuate the joints. The list of the obtained values of the gains are addressed in Table 4.4.

Table 4.4 – The control gains of the conventional CT-ST-SMC, the proposed FF-ST-SMC, and the standard PIDFF controllers used on SPIDER4 robot.

<table>
<thead>
<tr>
<th>Standard PIDFF</th>
<th>Conventional CT-ST-SMC</th>
<th>Proposed FF-ST-SMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K_p = 3500 )</td>
<td>( \Lambda = 80 )</td>
<td>( \Lambda = 90 )</td>
</tr>
<tr>
<td>( K_d = 40 )</td>
<td>( K_1 = 3 )</td>
<td>( K_1 = 7.5 )</td>
</tr>
<tr>
<td>( K_i = 1500 )</td>
<td>( K_2 = 2 )</td>
<td>( K_2 = 5 )</td>
</tr>
<tr>
<td>( K_3 = 18 )</td>
<td>( K_3 = 25 )</td>
<td></td>
</tr>
</tbody>
</table>

4.4.1.1 Feedforward PID versus conventional CT-ST-SMC

In this section, the conventional CT-ST-SMC algorithm and the standard PIDFF control are implemented on SPIDER4 robot. Two scenarios are adopted for this experimental demonstration:

- Scenario 1: Trajectory I shown in Figure 4.8, at feedrate of 12000 Inch/Minute.
- Scenario 2: Trajectory II shown in Figure 4.9, at feedrate of 12000 Inch/Minute.

Scenario 1

The conventional CT-ST-SMC algorithm shows a bad behavior when the robot follows Trajectory I. The evolution of the control signals provided by the conventional CT-ST-SMC control law is plotted in Figure 4.23. The high chattering effect that appears clearly in the control signals induced a lot of vibrations into the mechanical structure of the robot. The generated control signal may heat the electrical circuits and lead to premature wear in actuators. It is harmful to the actuators and this scenario was not repeated any more.

As discussed before (refer to Chapter 3), the chattering signal coming from the standard super-twisting control, which reduces chattering and not totally eliminates it, may be stimulated within a computed-torque control formulation. Moreover, the measurement noise of the experimental platform can elevate the effect of chattering phenomena deteriorating the dynamic performance.
In order to reduce the effect of chattering, the measured signals are treated with second order filters and more smooth trajectory is adopted in scenario 2.

**Scenario 2**

In this scenario, the robot’s nacelle follows Trajectory II at feedrate of 12000 Inch/Minute considering that circular motions can be more smooth on the actuators. The measured signals and the generated output are treated with second order filters only for the conventional CT-ST-SMC algorithm. A comparison between the standard PIDFF control and the conventional CT-ST-SMC algorithm is demonstrated in the sequel.

The Cartesian tracking errors for both controllers are plotted in Figure 4.24. One can observe the superiority of the PIDFF control law on the conventional CT-ST-SMC in terms of precision overall the reference trajectory. The RMSE performance index is evaluated in both Cartesian and joint spaces for the two controllers and reported in Table 4.5.

The generated control input torques of the four motors of SPIDER4 robot for both controllers are depicted in Figure 4.25. It is clear that both control algorithms generate an
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

Figure 4.24 – Scenario 2: Evolution of the Cartesian tracking errors of the standard PIDFF and the conventional CT-ST-SMC controllers on SPIDER4 robot.

input signal within the admissible limits of the motors. However, still low vibrations can be observed on the mechanical system of the robot. The control signal of the conventional CT-ST-SMC is more frequent than that of the standard PIDFF control as shown in Figure 4.25.

Thus, the generated control signals can explain the bad performance of the conventional CT-ST-SMC. The high nonlinearities abundant in SPIDER4 robot with large parameter values (inertia and mass matrix) are inconvenient for the computed-torque-based super-twisting control formulation.

Table 4.5 – Scenario 2: Control performance evaluation of the standard PIDFF and the conventional CT-ST-SMC controllers on SPIDER4 robot.

<table>
<thead>
<tr>
<th></th>
<th>RMSE$_C$ [mm]</th>
<th>RMSE$_J$ [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional CT-ST-SMC</td>
<td>1.9895</td>
<td>0.2751</td>
</tr>
<tr>
<td>Standard PIDFF</td>
<td>0.6785</td>
<td>0.0521</td>
</tr>
</tbody>
</table>
Finally, this experimental demonstration proved that computed-torque based control approaches with sliding mode controllers are sensitive to chattering effect and measurement noises. This was more effective especially when dealing with dynamical systems of high nonlinearities and large parameter values such as SPIDER4 robot.

### 4.4.1.2 Feedforward PID versus proposed feedforward ST-SMC

Within this section, the experimental results on SPIDER4 robot of a PIDFF control and the proposed FeedForward ST-SMC (FF-ST-SMC) algorithm are demonstrated. The considered trajectory for these experiments is Trajectory I in which both linear and circular motions are generated.
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

Two main scenarios are conducted in this experimental demonstration: scenario 1: nominal case, scenario 2: robustness towards speed changes.

Scenario 1: nominal case

This scenario allows the robot’s nacelle to follow Trajectory I with a feedrate of 6000 Inch/Minute.

Following that trajectory, the Cartesian tracking errors for both controllers (standard PIDFF and proposed FF-ST-SMC) are registered and plotted in Figure 4.26. One can observe a good error regulation is performed by the proposed controller on all the translational axes compared to the classical PIDFF control. In particular, the tracking error at y-axis is dragged towards zero with the proposed controller by a remarkable compensation can be noticed clearly in Figure 4.26. Due to the horizontal inclination of SPIDER4 robot and its heavy parts, the y-axis motion is highly subjected to the effect of gravity. Thus, we can notice from the tracking errors that the proposed FF-ST-SMC is more robust towards gravitational effects than the standard PIDFF control.

![Graph showing Cartesian tracking errors for standard PIDFF and proposed FF-ST-SMC controllers on SPIDER4 robot.](image)

Figure 4.26 – Scenario 1: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot.
The RMSE performance indices of the Cartesian and joint tracking errors are evaluated for both controllers over all the reference trajectory and the obtained values are summarized in Table 4.6. A significant improvement in the dynamic performance by the proposed FF-ST-SMC algorithm is noticed compared to the standard PIDFF. The RMSE of the Cartesian space is reduced by 55.4% and that of the joint space is reduced by 44.1%.

Compared to the PIDFF control structure, the FF-ST-SMC approach has two sign-based functions in terms of the sliding surface instead of the integral term in the PIDFF. These robust functions proved experimentally their wonderful performance in terms of disturbance-rejection during dynamic motions and stationary positions (at the end of the trajectory, see Figure 4.26).

The generated control input torques of the four motors of SPIDER4 robot for both controllers are illustrated in Figure 4.27. It is clear that both control algorithms generate an input signal within the capabilities of the motors. Both control signals are smooth enough to function the motors normally.

Figure 4.27 – Scenario 1: Evolution of the control input torques of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot.
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

Table 4.6 – Scenario 1: Control performance evaluation of both controllers of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot.

<table>
<thead>
<tr>
<th></th>
<th>RMSE&lt;sub&gt;C&lt;/sub&gt; [mm]</th>
<th>RMSE&lt;sub&gt;J&lt;/sub&gt; [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard PIDFF</td>
<td>0.6026</td>
<td>0.0472</td>
</tr>
<tr>
<td>Proposed FF-ST-SMC</td>
<td>0.2689</td>
<td>0.0264</td>
</tr>
<tr>
<td>Improvements</td>
<td>55.4 %</td>
<td>44.1 %</td>
</tr>
</tbody>
</table>

This scenario validated the relevance and applicability of the proposed FF-ST-SMC in real-time experiments. It showed a high dynamic performance by the proposed controller compared to the classical PIDFF control law.

Scenario 2: robustness towards speed changes

In this scenario, the feedrate of the robot’s nacelle is increased to 36000 Inch/Minute following the reference trajectory: Trajectory I. The intend behind this scenario is to test the performance of the proposed controller at high-speed motions when the nonlinearity effects of the parallel manipulator increase considerably.

The Cartesian tracking errors for both controllers (standard PIDFF and proposed FF-ST-SMC) are depicted in Figure 4.28. Knowing that the peak errors of both controllers are greater than the obtained ones during scenario 1, but still the proposed FF-ST-SMC controller perform better than standard PIDFF in terms of precision. Similar observations to scenario 1 are noticed in this scenario. The dynamic error is reduced considerably by the proposed controller compared to the PIDFF control law as well as the static error.

The evaluations of the performance indices of both controllers are summarized in Table 4.7. Remarkable improvements are obtained by the proposed controller compared to the standard PIDFF. The RMSE in Cartesian space is reduced by 44.3 % while that of joint space is also reduced by 38.4 %. The disturbance-rejection and high nonlinearities compensation at high-speed motions are verified by the proposed FF-ST-SMC approach improving the global dynamic performance of SPIDER4 robot.

The evolution of the control input torques generated by the two controllers is plotted
CHAPTER 4. REAL-TIME EXPERIMENTS AND RESULTS

Figure 4.28 – Scenario 2: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot.

in Figure 4.29. The control signals show a good behavior within the allowable capacities of the motors.

This scenario demonstrates the relevance and effectiveness of the proposed FF-ST-SMC approach in terms of nonlinearities compensation, disturbance-rejection, and precision. The superiority of the proposed control solution is verified experimentally compared to the basic PIDFF control law. Moreover, the applicability in a simple way and less computational efforts of the proposed FF-ST-SMC algorithm is validated.

Table 4.7 – Scenario 2: Control performance evaluation of the standard PIDFF and the proposed FF-ST-SMC controllers on SPIDER4 robot.

<table>
<thead>
<tr>
<th></th>
<th>RMSE$_C$ [mm]</th>
<th>RMSE$_J$ [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard PIDFF</td>
<td>0.92064</td>
<td>0.08421</td>
</tr>
<tr>
<td>Proposed FF-ST-SMC</td>
<td>0.5127</td>
<td>0.0519</td>
</tr>
<tr>
<td>Improvements</td>
<td>44.3 %</td>
<td>38.4 %</td>
</tr>
</tbody>
</table>
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

Last but not least, the real-time experiments on SPIDER4 robot of the proposed feedforward ST-SMC showed a good global performance at low and high dynamic operating conditions compared to the conventional computed torque ST-SMC and the standard PID control with computed feedforward. Furthermore, it has been shown experimentally the high sensitivity of the conventional computed torque ST-SMC algorithm to measurement noise especially when dealing with highly nonlinear PKMs of large dynamic parameters.

4.4.2 Experimental results of contribution 2 on Delta robot

In order to validate the proposed feedforward super-twisting SMC approach for pick-and-place industrial operations, real-time experiments of the standard PIDFF controller and the proposed one are conducted on the Delta robot. In this experimental demonstration, three scenarios are adopted as follows:

- Scenario 1: nominal case.
- Scenario 2: robustness towards payload and speed changes.
• Scenario 3: robustness towards very high accelerations.

The trajectory generated for the three scenarios is the industrial pick-and-place cycle motions depicted in Figure 4.3. The control gains obtained by Trial-and-Error tuning method on the real-time experimental platform are summarized in Table 4.8.

Table 4.8 – The control gains of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.

<table>
<thead>
<tr>
<th>Standard PIDFF</th>
<th>Proposed FF-ST-SMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p = 720$</td>
<td>$\Lambda = 360$</td>
</tr>
<tr>
<td>$K_d = 2$</td>
<td>$K_1 = 0.25$</td>
</tr>
<tr>
<td>$K_i = 3600$</td>
<td>$K_2 = 1.5$</td>
</tr>
<tr>
<td></td>
<td>$K_3 = 2$</td>
</tr>
</tbody>
</table>

4.4.2.1 Scenario 1: nominal case

In this scenario, Delta robot is allowed to follow the reference trajectory at acceleration of 2.5 G, at speed of 1500 mm/s, and without any additional payload. The end-effector traverses the proposed trajectory for 10 cycles of the pick-and-place motions shown in Figure 4.3.

The evolution of the Cartesian tracking errors of both implemented controllers is depicted in Figure 4.30. It is clear from the figure the reduced dynamic errors by the proposed FF-ST-SMC algorithm compared to the standard PIDFF control. The produced peak errors by the standard PIDFF control can be noticed larger than that of the proposed FF-ST-SMC overall the whole trajectory and in the three axes.

To quantify the achieved improvement by the proposed control solution, the RMSE performance index is evaluated for both PIDFF and FF-ST-SMC controllers in Cartesian and joint spaces. The evaluation of the RMSEs show an improvement of 26 % and 31 % in terms of Cartesian and joint tracking errors respectively (see Table 4.9). Better performance is verified by the proposed model-based super-twisting algorithm in terms of precision during dynamic motions.
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

Figure 4.30 – Scenario 1: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.

The generated control signals of both controllers are illustrated in Figure 4.31. The proposed FF-ST-SMC delivers control inputs of less amplitudes than that delivered by the standard PIDFF. This remarkable observation is quantified using the input-torque-based criterion \((4.6)\) giving an improvement of 19.5\% in terms of energy consumption (see Table 4.9).

Table 4.9 – Scenario 1: Control performance evaluation of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.

<table>
<thead>
<tr>
<th></th>
<th>RMSE(_C) [mm]</th>
<th>RMSE(_J) [deg]</th>
<th>(E_T) [Nm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard PIDFF</td>
<td>0.1392</td>
<td>0.0362</td>
<td>(2.9486 \times 10^4)</td>
</tr>
<tr>
<td>Proposed FF-ST-SMC</td>
<td>0.1031</td>
<td>0.025</td>
<td>(2.3737 \times 10^4)</td>
</tr>
<tr>
<td>Improvements</td>
<td>26%</td>
<td>31%</td>
<td>19.5%</td>
</tr>
</tbody>
</table>

This scenario verified the relevance of the proposed feedforward ST-SMC algorithm for
Figure 4.31 – Scenario 1: Evolution of the control input torques of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.

pick-and-place cycle motions. It showed an improved dynamic performance in terms of precision and energy consumption.

4.4.2.2 Scenario 2: robustness towards payload and speed changes

This scenario imposes on the robot a high operating acceleration of 20 G and speed of 2150 mm/s while attaching to its end-effector a payload of mass 210 g. For safety purposes, the performed number of cycles is reduced to 2 cycles. The idea behind this scenario is to test the performance of the proposed control solution at high dynamic operating conditions (high acceleration and added payload).

The generated Cartesian tracking errors of both PIDFF and FF-ST-SMC controllers are sketched in Figure 4.32. The proposed control solution attains better tracking performance compared to the standard PIDFF at high-speed motions with an attached payload to the end-effector. Less oscillations can be noticed in the Cartesian tracking errors coming from the proposed FF-ST-SMC in comparison to those from the standard PIDFF.
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

Figure 4.32 – Scenario 2: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.

The control performance is quantified for both controllers in Cartesian and joint spaces by the RMSE performance index. The obtained RMSEs values are summarized in Table 4.10. Significant improvements are monitored in Cartesian and joint tracking errors by the proposed FF-ST-SMC. For Cartesian tracking errors, a reduction of 58 % from the standard PIDFF to the FF-ST-SMC algorithm is remarked, while a reduction of 53.5 % is remarked for joint tracking errors. Thanks to the robust terms of the proposed FF-ST-SMC, more disturbance-rejection is achieved compared to the standard PIDFF. Thus, the robustness towards payload and speed changes of the proposed control approach is validated.

Figure 4.33 displays the evolution of the control signals provided by both controllers overall the reference trajectory. All the control signals are within the admissible range of the motors. However, less peak torques are produced by the proposed controller compared to the PIDFF one. This reduction in energy consumption, from PIDFF to FF-ST-SMC controller, is evaluated by 22.1 % as shown in Table 4.10.

Last but not least, this scenario confirmed the robustness feature of the proposed FF-
ST-SMC towards high dynamic operating conditions. The proposed FF-ST-SMC approach performed much better than the standard PIDFF at high accelerations with a handled payload. The conclusion drawn is that the proposed FF-ST-SMC algorithm improves the dynamic performance of parallel manipulators in terms of high-speed motions, precision, robustness, and energy consumption.
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

4.4.2.3 Scenario 3: robustness towards very high accelerations

For more challenging task, Delta robot is configured to operate at very high acceleration of 30 G and speed of 2650 mm/s in this scenario. The pick-and-place trajectory of Figure 4.3 is followed without any additional payload. Two cycles are performed in this scenario.

The evolution of the Cartesian tracking errors of both controllers are depicted in Figure 4.34. Smaller dynamic errors and less oscillations are obtained from the proposed FF-ST-SMC algorithm compared to the standard PIDFF. The Cartesian tracking trajectories of the PIDFF and FF-ST-SMC controllers are illustrated in 3D views in Figures 4.35 and 4.36 respectively. One can be observe better tracking accuracy for the proposed FF-ST-SMC algorithm compared to the standard PIDFF controller.

![Figure 4.34 – Scenario 3: Evolution of the Cartesian tracking errors of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.](image)

The performance indices of both controllers for this scenario are summarized in Table 4.11. Improvements of 25 % and 28.9 % are remarked in the Cartesian and joint tracking errors respectively for the proposed FF-ST-SMC over the standard PIDFF. The good dynamic
performance and robustness of the proposed control approach at extremely high-speed motions are verified.

![Standard PIDFF control](image)

**Figure 4.35 – Scenario 3: Evolution of the Cartesian tracking trajectories with the standard PIDFF control on Delta robot.**

The generated control signals from both controllers overall the reference trajectory are depicted in Figure 4.37. The standard PIDFF control and the proposed one deliver control input torques within the capabilities of the motors. However, the control signal of the FF-ST-SMC algorithm features less amplitudes than that of the PIDFF control. The improvement in terms of energy consumption by the proposed controller is quantified by 15.2 % as shown in Table 4.11.

This scenario verified the effectiveness of the proposed feedforward ST-SMC algorithm at extremely high-speed motions. It showed the superiority of the proposed model-based ST-SMC approach in terms of robustness, nonlinearities compensation, and disturbance-rejection compared to the standard model-based PID control.
4.4. EXPERIMENTAL RESULTS OF CONTRIBUTION 2: MODEL-BASED ST-SMC ALGORITHM

Figure 4.36 – Scenario 3: Evolution of the Cartesian tracking trajectories with the proposed FF-ST-SMC algorithm on Delta robot.

Table 4.11 – Scenario 3: Control performance evaluation of the standard PIDFF and the proposed FF-ST-SMC controllers used on Delta robot.

<table>
<thead>
<tr>
<th></th>
<th>RMSE_C [mm]</th>
<th>RMSE_J [deg]</th>
<th>E_Γ [Nm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard PIDFF</td>
<td>1.0732</td>
<td>0.2916</td>
<td>$1.1556 \times 10^4$</td>
</tr>
<tr>
<td>Proposed FF-ST-SMC</td>
<td>0.8058</td>
<td>0.2074</td>
<td>$9.7987 \times 10^3$</td>
</tr>
<tr>
<td>Improvements</td>
<td>25 %</td>
<td>28.9%</td>
<td>15.2 %</td>
</tr>
</tbody>
</table>

To this end, the real-time experiments conducted on SPIDER4 and Delta robots demonstrated a good dynamic performance provided by the proposed feedforward ST-SMC. It has been shown the simple and easy implementation of the FF-ST-SMC in terms of computational efforts and sensitivity to measurement noise. The superiority of the proposed control approach among the conventional computed torque ST-SMC and PID control with computed feedforward has been validated.
4.5 Experimental results of contribution 3: PDFF control with actuator and friction dynamics

VELOCE robot (shown in Figure 4.10) is used as an experimental testbed to validate the proposed PDFF control with actuator and friction dynamics. To validate the relevance of the proposed control solution, the standard PDFF control with the classical dynamic model, the extended PDFF control with the actuator dynamics (inertia and damping coefficients) (Ex-PDFF I), and the PDFF control with both actuator and friction dynamics (Ex-PDFF II) are implemented on VELOCE robot (refer to equation (3.86)).

Then, a comparative study between the results of the three implementations is done. Two main scenarios are conducted for the implemented controllers: scenario 1: nominal case, scenario 2: robustness towards payload and speed changes.

The proportional and derivative feedback gains are chosen using the Trial-and-Error tuning method in real-time framework which gives $K_p = \text{diag}(4000,4000,4000,4000)$ and
4.5. EXPERIMENTAL RESULTS OF CONTRIBUTION 3: PDFF CONTROL WITH ACTUATOR AND FRICTION DYNAMICS

\[ K_d = \text{diag}(6, 6, 6, 6) \]. The parameters of the actuator dynamics are identified from the provided manuals of the motors driving VELOCE robot. The inertia of the actuator’s rotor is \( J = 0.0041 \text{ kg.m}^2 \) and the damping coefficient is provided as \( B = 0.0024 \). Note that VELOCE robot direct-drive motors are without gearboxes, and thus the gear ratio is 1 \( (R_G = 1) \).

4.5.1 Friction parameters identification results

The identification process of the friction parameters (explained in Section 3.4.3 of Chapter 3) is conducted on VELOCE robot with two operating speeds:

- Low speed: point-to-point motion duration of 0.5 s (acceleration 1 G),
- High speed: point-to-point motion duration of 0.15 s (acceleration 10 G).

The obtained values of the viscous friction, Coulomb friction, and zero-drift coefficient for each active joint at low and high speeds are illustrated in Figure 4.38. The negative deviation of the estimated value of viscous friction of actuator 3 may come from the modeling errors or not sufficient exciting trajectories compared with the measurement perturbations. However, the estimated friction parameters at both operating speeds approximately matches for all the axes. The used friction parameters for the control implementation are addressed in Table 4.12.

<table>
<thead>
<tr>
<th>Active Joint</th>
<th>( f_{vi} )</th>
<th>( f_{ci} )</th>
<th>( d_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1198</td>
<td>0.3019</td>
<td>0.1811</td>
</tr>
<tr>
<td>2</td>
<td>0.2252</td>
<td>0.8879</td>
<td>-0.5834</td>
</tr>
<tr>
<td>3</td>
<td>0.2354</td>
<td>0.0584</td>
<td>0.3194</td>
</tr>
<tr>
<td>4</td>
<td>0.3269</td>
<td>0.7104</td>
<td>-0.5891</td>
</tr>
</tbody>
</table>

4.5.2 Scenario 1: nominal case

In this scenario, the standard PDFF controller and the two formulated extensions (Ex-PDFF I and Ex-PDFF II) are applied on VELOCE PKM without any payload with a point-to-point motion duration fixed to \( T = 0.5 \text{ s} \) which gives an acceleration of 1 G.
Following the reference trajectory shown in Figure 4.13, the Cartesian tracking errors (where $\alpha = \theta_z$, the rotational angle around z-axis) for the three controllers are plotted in Figure 4.39. One can observe the reductions in the errors of Cartesian space overall the reference trajectory with the proposed dynamic controller.

For clarification purpose, the plots are zoomed to the interval $[4.5, 5.5]$ seconds as shown in Figure 4.40. We can notice that the best dynamic performance is obtained by the Ex-PDFF II controller showing the high impact of friction compensation on the global performance of parallel robots.

The RMSE performance indices of the three controllers are evaluated in Cartesian space and reported in Tables 4.13 and 4.14. It shows a small improvement of 5.62 % for translational tracking error and 3 % for rotational one when compensating only the actuator dynamics. While a significant improvement is validated for translational and rotational tracking errors, 19.1 % and 25.65 % respectively, when compensating in addition the friction dynamics.
4.5. EXPERIMENTAL RESULTS OF CONTRIBUTION 3: PDFF CONTROL WITH ACTUATOR AND FRICTION DYNAMICS

![Graph showing Cartesian tracking errors](image)

Figure 4.39 – Scenario 1: Evolution of the Cartesian tracking errors of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Control</th>
<th>$\text{RMSE}_T$ [mm]</th>
<th>$\text{RMSE}_R$ [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard PDFF</td>
<td></td>
<td>0.089</td>
<td>0.7614</td>
</tr>
<tr>
<td>Scenario 1</td>
<td>Ex-PDFF I</td>
<td>0.084</td>
<td>0.7386</td>
</tr>
<tr>
<td>Improvements</td>
<td></td>
<td>5.62 %</td>
<td>3 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Control</th>
<th>$\text{RMSE}_T$ [mm]</th>
<th>$\text{RMSE}_R$ [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard PDFF</td>
<td></td>
<td>0.089</td>
<td>0.7614</td>
</tr>
<tr>
<td>Scenario 1</td>
<td>Ex-PDFF II</td>
<td>0.072</td>
<td>0.5661</td>
</tr>
<tr>
<td>Improvements</td>
<td></td>
<td>19.1 %</td>
<td>25.65 %</td>
</tr>
</tbody>
</table>

The evolution of the control input signals of the four direct-drive motors of VELOCE robot for the three controllers are depicted in Figure 4.41. It is clear that all controllers
generate an input signal within the allowed capabilities of the motors. No significant improvement is observed in terms of energy consumption.

This scenario showed clearly the high positive impact of friction compensation on the dynamic performance of VELOCE robot in terms of precision and robustness towards non-linearities variation with respect to the position in workspace. Indeed, the actuator dynamics compensation lead to a small improvement of the dynamic performance in this scenario.

### 4.5.3 Scenario 2: robustness towards payload and speed changes

In this scenario, the standard PDFF controller and the two formulated extensions (Ex-PDFF I and Ex-PDFF II) are applied on VELOCE PKM with an additional payload to the end-effector of mass 200 g and with a point-to-point motion duration fixed to $T = 0.15$ s which gives an acceleration of 10 G. The purpose of this scenario is to test the robustness
4.5. EXPERIMENTAL RESULTS OF CONTRIBUTION 3: PDFF CONTROL WITH ACTUATOR AND FRICTION DYNAMICS

The same reference trajectory shown in Figure 4.13 is adopted within this scenario. The Cartesian tracking errors for the three controllers are plotted in Figure 4.42. The high peak errors of the standard PDFF control can be observed clearly, especially the tracking errors of \(x\) and \(y\) axes.

For better illustration, the plots are zoomed to the interval \([1.5, 2]\) seconds as shown in Figure 4.43. One can notice the superiority of the Ex-PDFF I and Ex-PDFF II controllers compared to the standard PDFF controller in terms of the precision. While, approximately, an equivalent behavior is detected for the two extended controllers.

Again, the RMSE performance index is evaluated in Cartesian space for the three controllers and reported in Tables 4.15 and 4.16. It quantifies an improvement of 28.08 % for translational tracking errors and 7.4 % for rotational tracking error when compensating only the actuator dynamics. Moreover, more improvements for translational and rota-
Figure 4.42 – Scenario 2: Evolution of the Cartesian tracking errors of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELoce robot.

Additional tracking errors, 33.81 % and 17.56 % respectively, are validated when compensating in addition the friction dynamics.

Table 4.15 – Scenario 2: Control performance evaluation of the standard PDFF and the Ex-PDFF I controllers on VELoce robot.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Control</th>
<th>$\text{RMSE}_T$ [mm]</th>
<th>$\text{RMSE}_R$ [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario 2</td>
<td>Standard PDFF</td>
<td>0.349</td>
<td>1.5119</td>
</tr>
<tr>
<td></td>
<td>Ex-PDFF I</td>
<td>0.251</td>
<td>1.4</td>
</tr>
<tr>
<td>Improvements</td>
<td>28.08 %</td>
<td>7.4 %</td>
<td></td>
</tr>
</tbody>
</table>

One can conclude that friction dynamics is less effective on the dynamic performance of parallel manipulators when operating at high-speed motions with a handled payload in comparison with the nominal scenario. It has been shown experimentally that compensating the actuator dynamics (especially inertia parameter) at high dynamic operating conditions can improve highly the tracking error ($\approx 28 \%$ at translational motions), while
Figure 4.43 – Scenario 2: Evolution of the Cartesian tracking errors of the standard PDFF, the Ex-PDFF I, and the Ex-PDFF II controllers on VELOCE robot within the interval [1.5, 2] sec.

Table 4.16 – Scenario 2: Control performance evaluation of the standard PDFF and the Ex-PDFF II controllers on VELOCE robot.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Control</th>
<th>RMSE_T [mm]</th>
<th>RMSE_R [deg]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Standard PDFF</td>
<td>0.349</td>
<td>1.5119</td>
</tr>
<tr>
<td>Scenario 2</td>
<td>Ex-PDFF II</td>
<td>0.231</td>
<td>1.2464</td>
</tr>
<tr>
<td></td>
<td>Improvements</td>
<td>33.81 %</td>
<td>17.56 %</td>
</tr>
</tbody>
</table>

Friction compensation can add a small but not negligible improvement (from 28.08 % to 33.81 % at translational motion).

The evolution of the control input signals of the four direct-drive motors in this scenario are illustrated in Figure 4.44. All the generated control signals are within the capabilities of the motors knowing that no significant improvement is observed in terms of energy consumption.
This scenario confirms the improved performance at high dynamic operating conditions when taking into consideration the actuator and friction dynamics within the control design. It has been shown less tracking errors of the control with the feedforward compensator of actuator and friction dynamics compared to the control without that compensator. Without loss of generality, real-time experiments showed that actuator dynamics have a high impact on the parallel manipulator performance at high dynamic operating conditions and low impact at nominal conditions, while vice-versa for friction dynamics.

### 4.6 Conclusion

This chapter provided the experimental validation and demonstration of the proposed control solutions within this thesis. It began with descriptions of the existing experimental setups and the trajectory generation techniques adopted for each parallel manipulator prototype. Three experimental platforms were used for the control validation: a 3-DOF Delta robot, at EPFL, Switzerland, a 5-DOF SPIDER4 robot and a 4-DOF VELOCE robot at LIRMM.
4.6. Conclusion

Delta robot was provided for the control validation of the proposed time-varying feedback RISE control, Contribution 1, and feedforward super-twisting sliding mode control, Contribution 2. Real-time experiments of the proposed time-varying feedback RISE control verified its superiority over the original RISE algorithm in terms of precision and robustness towards payload and speed variations. This is due to the special behavior of the proposed nonlinear feedback gains with the new time-varying feedback RISE control. Moreover, experimental results of the proposed feedforward ST-SMC algorithm showed a high dynamic performance in terms of extremely high-speed motions, precision, and robustness towards payload and speed changes.

Moreover, SPIDER4 robot was used for the experimental validation of the proposed feedforward super-twisting SMC algorithm (Contribution 2). Experimental results showed that the proposed control solution overcomes the conventional computed torque super-twisting algorithm and the standard PID with computed feedforward in terms of dynamic performance. It has been shown that the conventional computed torque super-twisting algorithm is more sensitive to measurements noise compared to the standard PIDFF and the proposed controller especially when dealing with highly nonlinear systems of large dynamic parameter values.

VELOCE robot has been used for the experimental validation of Contribution 3, the actuator and friction dynamics formulation in closed-loop control. It has been verified that incorporating more dynamic terms within a model-based control strategy can improve the dynamic performance of parallel manipulators in terms of precision, nonlinearities compensation, and robustness towards payload and speed variations. The high effect of the actuator and friction dynamics on the global performance of parallel robots was demonstrated experimentally.
The objectives of this thesis have aimed at improving the dynamic performances of parallel manipulators by developing robust control strategies and compensating for the errors coming from the actuator dynamics, the frictions in the articulations, the motors drivers, etc. Several parallel manipulator prototypes were available to validate the proposed control strategies in real-time experiments at different operating conditions showing their effectiveness in terms of motion speed, precision, and robustness.

Summary of the work

Control of parallel manipulators is not a trivial task since of their highly nonlinear dynamics which may increase considerably when operating at high accelerations, often leading to mechanical vibrations. Moreover, uncertainties are abundant in such systems due to model simplifications, the wear of the components of the robot and the variations of the environment. Furthermore, their highly coupled dynamics, singularities and actuation redundancy in some mechanisms give rise to very complex and challenging control issues. Consequently, the developed control schemes should take into account all the previously mentioned issues and challenges.

In this thesis, the proposed, analysed, and validated control solutions can be mentioned as follows:
1. An extended version of the standard RISE feedback control strategy has been proposed and developed in which some static feedback gains were replaced by time-varying ones. The idea was motivated by the proved effectiveness and robustness of the nonlinear feedback gains used with different control schemes such as PD control. RISE control law can accommodate a large class of different uncertainties and disturbances with limited restrictions on the system dynamics. Thus, the proposed time-varying feedback RISE control law takes the advantages of the nonlinear feedback gains and the robustness of the RISE controller. The stability of the proposed control solution has been studied in the sense of Lyapunov stability showing an asymptotic convergence of the tracking error. A 3-DOF Delta robot has been used to validate the proposed controller in real-time experiments. Real-time experiments verified the superiority of the proposed time-varying feedback RISE control over the original RISE algorithm in terms of precision and robustness towards payload and speed variations.

2. A novel model-based super-twisting sliding mode control was proposed and designed such that the dynamic compensation term relies on the desired trajectories instead of the measured ones. The conventional super-twisting algorithm developed for robotic manipulator dynamics has the structure of a computed torque control which is sensitive to measurements noise. This can deteriorate the dynamic performance of the manipulator and reduce the robustness towards changes of operating conditions. Moreover, relying on the desired trajectory in the case of feedforward control strategies is more computationally efficient than the computed torque control. The proposed feedforward super-twisting algorithm comprises a feedforward dynamic compensator, the super-twisting control, and a feedback stabilizing term. The stability analysis of the proposed control solution has been addressed in the sense of Lyapunov ensuring the local asymptotic convergence of the tracking error and the finite time convergence of the sliding variable. Two parallel robot prototypes have been used to validate the proposed control strategy: a 3-DOF Delta robot and a 5-DOF SPIDER4 robot. Real-time experiments have shown the superiority of the proposed control approach among the conventional computed torque super-twisting
algorithm and the PID control with computed feedforward in different operating conditions (nominal case, payload changes, and operating speed changes).

3. Actuator and friction dynamics formulation has been proposed and used as a compensator within a closed-loop control, a PD with computed feedforward. The main motivation behind this proposition was to compensate for the errors coming from the actuator and friction dynamics aiming at better performances of parallel manipulators. The proposed control incorporating the actuator and friction dynamics has been studied and analysed using a Lyapunov function candidate. The stability analysis showed a global asymptotic convergence of the tracking error. A 4-DOF VELOCE robot has been adopted for the real-time experimental validation of the proposed control formulation. An offline friction parameters identification technique has been conducted on VELOCE robot and the obtained values were used in the control law. Experimental results showed the effectiveness and relevance of the proposed dynamic formulation in terms of precision and robustness towards changes of operating conditions.

**Future works**

In this thesis, several strategies have been employed in order to improve the dynamic performance of parallel manipulators in terms of precision, robustness, and changes of operating conditions. Indeed, different possibilities exist to extend the proposed control solutions in this work and achieve better performances. One can mention the extension possibilities as follows:

- Extend the proposed time-varying feedback RISE control law with a dynamic compensating term in the form of computed or adaptive feedforward. This can accommodate for the nonlinear dynamics of parallel manipulators enhancing the tracking precision and the robustness.

- Apply an online dynamic adaptation for the proposed feedforward super-twisting sliding mode control taking into account the time-varying parameters. Consider the
scenario of real machining with SPIDER4 robot with addressing the problems of contact forces, compliance errors, stiffness, etc.

- For the proposed actuator and friction dynamics formulation, design an online estimator of the friction parameters using adaptive control techniques. Consider more complicated nonlinear models of friction dynamics incorporating those of passive joints. Try to look for more tools that may be considered as sources of errors such as: electrical dynamics of the actuators, cogging ripple torques, amplifiers, etc.


M. Asgari and M. A. Ardestani. Dynamics and improved computed torque control of a novel medical parallel manipulator: applied to chest compressions to assist in car-


M. Bennehar, A. Chemori, and F. Pierrot. A new extension of desired compensation adaptive control and its real-time application to redundantly actuated PKMs. In *2014


BIBLIOGRAPHY


Property 8. Given the vector of the residual dynamics of a robotic manipulator incorporating the friction parameters as follows:

\[
\begin{align*}
\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) &= \left[ M(\mathbf{q}_d) - M(\mathbf{q}) \right] \ddot{\mathbf{q}}_d + \left[ C(\mathbf{q}_d, \dot{\mathbf{q}}_d) - C(\mathbf{q}, \dot{\mathbf{q}}) \right] \dot{\mathbf{q}}_d \\
&\quad + \left[ G(\mathbf{q}_d) - G(\mathbf{q}) \right] + F_c \left[ \tanh(\dot{\mathbf{q}}_d) - \tanh(\dot{\mathbf{q}}) \right]
\end{align*}
\]  
(A.1)

where the actuator inertia is included with the mass and inertia matrix \( M \), the damping coefficient and viscous friction are included within the Coriolis and centrifugal matrix \( C \), and \( \tanh(x) = \left[ \tanh(x_1), \ldots, \tanh(x_n) \right]^T \) is the vectorial hyperbolic hyperbolic tangent function for any \( x \in \mathbb{R}^n \). Knowing that the norms of the desired velocity and acceleration vectors are upper bounded by \( \| \ddot{\mathbf{q}}_d \|_M \) and \( \| \dddot{\mathbf{q}}_d \|_M \) respectively, then there exist two positive constants \( k_{h_1}, k_{h_2} > 0 \) such that

\[
\| \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \| \leq k_{h_1} \| \dot{\mathbf{e}} \| + k_{h_2} \| \tanh(\mathbf{e}) \|
\]  
(A.2)

for all \( \mathbf{e}, \dot{\mathbf{e}} \in \mathbb{R}^n \).

Proof. The norm of the above residual dynamics function \( \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \) can be upper bounded
APPENDIX A. BOUNDED RESIDUAL DYNAMICS OF ROBOTIC MANIPULATORS

as follows:

\[
\|h(q, \dot{q})\| \leq \|(M(q_d) - M(q))q_d\| + \|(C(q_d, \dot{q}_d) - C(q, \dot{q}))q_d\| + \|G(q_d) - G(q)\|
\]

\[
+ \|F_c(\tanh(\dot{q}_d) - \tanh(\dot{q}))\|
\]

(A.3)

Regarding the first term at the right-hand side of (A.3) that consists of the mass and inertia matrix, two upper bounds can be derived from Property 1 (refer to Chapter 1) as follows:

\[
\|(M(q_d) - M(q))q_d\| \leq k_M\|\dot{q}_d\|_M\|\dot{e}\|
\]

\[
(A.4)
\]

where \(k_M\) and \(k'_M\) are two positive constants. Similarly, the second term of (A.3) can be upper bounded by two bounds using Property 2 (refer to Chapter 1) as follows:

\[
\|(C(q_d, \dot{q}_d) - C(q, \dot{q}))\dot{q}_d\| \leq k_C1\|\dot{q}_d\|_M\|\dot{e}\| + k_C2\|\dot{q}_d\|_M^2\|\dot{e}\|
\]

\[
(A.5)
\]

where \(k_{C1}\) and \(k_{C2}\) are two positive constants. Making use of Property 3 (refer to Chapter 1), the third term of (A.3) can be upper bounded by two bounds as follows:

\[
\|G(q_d) - G(q)\| \leq k_G\|\dot{e}\|
\]

\[
(A.6)
\]

where \(k_G\) and \(k'_G\) are two positive constants. The last term of (A.3) consisting of Coulomb friction dynamics can be upper bounded also by two bounds. The first upper bound can be established using the inequality \(\|\tanh(x)\| \leq \|x\|\) \(\forall x \in \mathbb{R}^n\) as follows:

\[
\|F_c(\tanh(\dot{q}_d) - \tanh(\dot{q}))\| \leq 2f_c\|\dot{q}_d\|_M + f_c\|\dot{e}\|
\]

\[
(A.7)
\]

where \(f_c\) is a positive constant value such that \(\|F_c\| \leq f_c\).

Furthermore, using property of the vectorial hyperbolic tangent function (B.1) derived in Appendix B, one can derive another upper bound as follows:

\[
\|F_c(\tanh(\dot{q}_d) - \tanh(\dot{q}))\| \leq f_c\|\dot{e}\|\|S\|\left(\|I_v\| + \|\tanh(\dot{q}_d)\|\|S\|\|\tanh(\dot{q})\|\right)
\]

\[
(A.8)
\]
where \( S = [1, 0, …, 0]^T \), \( I_v = [1, …, 1]^T \) are \( n \)-dimensional vectors. Considering the following bounds: \(|S| \leq 1\), \(|I_v| \leq \sqrt{n}\), \(|\tanh(\dot{e})| \leq |\dot{e}|\), \(|\tanh(\dot{q}_d)| \leq \sqrt{n}\), and \(|\tanh(\dot{q})| \leq \sqrt{n}\), (A.8) can be upper bounded as follows:

\[
||F_c(\tanh(\dot{q}_d) - \tanh(\dot{q}))|| \leq f_c(n + \sqrt{n})||\dot{e}||
\]  

(A.9)

Finally, the upper bounds on the residual dynamics (A.3) can be established using (A.4), (A.5), (A.6), (A.7), and (A.9) as follows:

\[
||h(q, \dot{q})|| \leq (k_{C_1}||\dot{q}_d||_M + f_c(n + \sqrt{n}))(||\dot{e}|| + \left(k_M||\dot{q}_d||_M + k_{C_2}||\dot{q}_d||_M^2 + k_G\right)||e||
\]  

(A.10)

Knowing that \( f_c < f_c(n + \sqrt{n}) \), one can combine the two inequalities of (A.10) as follows:

\[
||h(q, \dot{q})|| \leq k_{h_1}||\dot{e}|| + S(e)
\]  

(A.11)

where

\[
k_{h_1} \geq f_c(n + \sqrt{n}) + k_{C_1}||\dot{q}_d||_M
\]  

(A.12)

and

\[
S(e) = \begin{cases} s_1||e||, & \text{if } ||e|| < s_2/s_1 \\ s_2, & \text{if } ||e|| \geq s_2/s_1 \end{cases}
\]  

(A.13)

with

\[
s_1 = k_G + k_M||\dot{q}_d||_M + k_{C_2}||\dot{q}_d||_M^2
\]  

\[
s_2 = 2\left(k_G' + k_M'||\dot{q}_d||_M + k_{C_1}||\dot{q}_d||_M^2 + f_c||\dot{q}_d||_M\right)
\]  

(A.14)

The scalar function \( S(e) \) can be illustrated in Figure A.1 where the dotted region represents the belonging region of the two upper bounds \( s_1||e|| \) and \( s_2 \). From Figure A.1, one can upper bound the function \( S(e) \) by a hyperbolic tangent function of \( ||e|| \) as follows:

\[
|S(e)| \leq k_{h_2}\tanh(||e||)
\]  

(A.15)

where \( k_{h_2} \) is a number that satisfies the following:

\[
k_{h_2} \geq \frac{s_2}{\tanh\left(\frac{s_2}{s_1}\right)}
\]  

(A.16)
Making use of the fact that $\tanh(||x||) \leq ||\tanh(x)||$ for all $x \in \mathbb{R}^n$, the upper bound of the residual dynamics can be finally given as follows:

$$||h(q, \dot{q})|| \leq k_{h_1} ||\dot{e}|| + k_{h_2} ||\tanh(e)||$$  \hspace{1cm} (A.17)

and the proof is concluded. \qed
Property 9. For any two vectors \( x, y \in \mathbb{R}^n \), the following equality holds:

\[
\tanh(x) - \tanh(y) = \tanh(x - y) S^T \left( I_v - \tanh(x) S^T \tanh(y) \right) \tag{B.1}
\]

where \( \tanh(u) = [\tanh(u_1), ..., \tanh(u_n)]^T \) is the vectorial hyperbolic tangent for any \( u \in \mathbb{R}^n \), \( S = [1, 0, ..., 0]^T \in \mathbb{R}^n \), and \( I_v = [1, ..., 1]^T \in \mathbb{R}^n \).

Proof. Developing and expanding the left hand side of (B.1) leads to the following:

\[
\tanh(x) - \tanh(y) = \begin{pmatrix} \tanh(x_1) \\ \vdots \\ \tanh(x_n) \end{pmatrix} - \begin{pmatrix} \tanh(y_1) \\ \vdots \\ \tanh(y_n) \end{pmatrix} = \begin{pmatrix} \tanh(x_1) - \tanh(y_1) \\ \vdots \\ \tanh(x_n) - \tanh(y_n) \end{pmatrix} \tag{B.2}
\]

Using the subtraction formula of the conventional hyperbolic tangent function given below:

\[
\tanh(a - b) = \frac{\tanh(a) - \tanh(b)}{1 - \tanh(a) \tanh(b)} \quad \forall \quad a, b \in \mathbb{R}
\]
one can develop (B.2) as follows:

\[
\tanh(x) - \tanh(y) = \begin{pmatrix}
\tanh(x_1 - y_1)(1 - \tanh(x_1)\tanh(y_1)) \\
\vdots \\
\tanh(x_n - y_n)(1 - \tanh(x_n)\tanh(y_n))
\end{pmatrix}
\]

\[= \begin{pmatrix}
\tanh(x_1 - y_1) \\
\vdots \\
\tanh(x_n - y_n)
\end{pmatrix} (I_0 \ldots 0) \begin{pmatrix}
1 \\
\vdots \\
1
\end{pmatrix} - \begin{pmatrix}
\tanh(x_1) \\
\vdots \\
\tanh(x_n)
\end{pmatrix} (I_0 \ldots 0) \begin{pmatrix}
1 \\
\vdots \\
1
\end{pmatrix}
\]

\[= \tanh(x) - \tanh(y) = \tanh(x - y) S^T (I_v - \tanh(x) S^T \tanh(y)) \]

\[= \tanh(x - y) S^T (I_v - \tanh(x) S^T \tanh(y)) \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (B.3)
\]

and the proof is concluded. \qed
Abstract

Parallel Kinematic Manipulators (PKMs) have gained an increased popularity in the last few decades. This interest has been stimulated by the significant advantages of PKMs compared to their serial counterparts, such as better precision and higher acceleration capabilities. Efficient and performant control algorithms play a crucial role in improving the overall performance of PKMs. Control of PKMs is often considered in the literature a challenging task due to their highly nonlinear dynamics, abundant uncertainties, parameters variation, and actuation redundancy. In this thesis, we aim at improving the dynamic performance of PKMs in terms of precision and robustness towards changes of operating conditions. Thus, we propose robust control strategies being extensions of (i) the standard Robust Integral of the Sign of the Error (RISE) feedback control and (ii) the super-twisting Sliding Mode Control (SMC). Moreover, an actuator and friction dynamics formulation is proposed within a model-based control strategy to compensate for their resulting errors. Lyapunov-based stability analysis is established for all the proposed controllers verifying the asymptotic convergence of the tracking errors. In order to validate the proposed controllers, real-time experiments are conducted on several parallel robot prototypes: the 3-DOF Delta robot at EPFL, Switzerland, the 4-DOF VELOCE robot, and the 5-DOF SPIDER4 robot at LIRMM, France. Several experiments are tested including nominal scenarios, robustness towards speed variation, and robustness towards payload changes. The relevance of the proposed control schemes is proved through the improvement of the tracking errors at different dynamic operating conditions.

Keywords: Parallel manipulators, dynamic model, sliding mode control, RISE control, stability analysis, real-time experiments.
Résumé

Les robots manipulateurs parallèles ont acquis une popularité croissante au cours des dernières décennies. Cet intérêt a été stimulé par leurs grands avantages par rapport à leurs homologues sériels, en termes de précision et d’accélérations élevées. Le développement d’approches de commande efficaces et performantes joue un rôle primordial dans l’amélioration des performances globales des robots parallèles. La commande des robots parallèles est souvent considérée dans la littérature comme un challenge en raison de leur dynamique hautement non linéaire, de leurs incertitudes abondantes, des variations paramétriques et de la redondance d’actionnement. Dans cette thèse, nous visons à améliorer les performances dynamiques des robots parallèles en matière de précision et de robustesse vis-à-vis des changements dans les conditions opérationnelles. Ainsi, des approches de commande robustes ont été proposées, résultantes de l’extension de (i) la commande RISE (Robust Integral of the Sign of the Error) standard, (ii) la commande par mode glissant d’ordre supérieure (Super Twisting). D’autre part, une nouvelles formulation à base de dynamique d’actionneurs et de frottement a été proposée dans une approche de commande basée-modèle pour la compensation de leurs erreurs résultantes. La stabilité des approches de commande proposées a été analysée par des techniques de Lyapunov, vérifiant la convergence asymptotique des erreurs de suivi. Afin de valider les solutions de commande proposées, des tests expérimentaux ont été réalisés sur différents prototypes de robots parallèles, à savoir : le robot Delta à 3 ddl (degrés de liberté) À l’EPFL, en Suisse, le robot VELOCE è 4 ddl et le robot SPIDER4 à 5 ddl au LIRMM, en France. Différents scénarios d’expérimentation ont été effectués, y compris le cas nominal, le test de robustesse vis-à-vis des variations de vitesse, et le test de robustesse vis-à-vis des variations de charge utile. La pertinence des approches de commande proposées a été prouvée à travers l’amélioration des erreurs de suivi pour différentes conditions opérationnelles dynamiques.

Mots clefs : Robot Manipulateurs parallèles, modèle dynamique, commande par mode glissant, commande RISE, analyse de stabilité, expérimentation en temps réel.