

Lecture Notes in Mechanical Engineering

Raviprakash R. Salagame
Palaniappan Ramu
Indira Narayanaswamy
Dhish Kumar Saxena *Editors*

Advances in Multidisciplinary Analysis and Optimization

Proceedings of the 2nd National
Conference on Multidisciplinary Analysis
and Optimization

Lecture Notes in Mechanical Engineering

Series Editors

Francisco Cavas-Martínez, Departamento de Estructuras, Universidad Politécnica de Cartagena, Cartagena, Murcia, Spain

Fakher Chaari, National School of Engineers, University of Sfax, Sfax, Tunisia

Francesco Gherardini, Dipartimento di Ingegneria, Università di Modena e Reggio Emilia, Modena, Italy

Mohamed Haddar, National School of Engineers of Sfax (ENIS), Sfax, Tunisia

Vitalii Ivanov, Department of Manufacturing Engineering Machine and Tools, Sumy State University, Sumy, Ukraine

Young W. Kwon, Department of Manufacturing Engineering and Aerospace Engineering, Graduate School of Engineering and Applied Science, Monterey, CA, USA

Justyna Trojanowska, Poznan University of Technology, Poznan, Poland

Lecture Notes in Mechanical Engineering (LNME) publishes the latest developments in Mechanical Engineering—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNME. Volumes published in LNME embrace all aspects, subfields and new challenges of mechanical engineering. Topics in the series include:

- Engineering Design
- Machinery and Machine Elements
- Mechanical Structures and Stress Analysis
- Automotive Engineering
- Engine Technology
- Aerospace Technology and Astronautics
- Nanotechnology and Microengineering
- Control, Robotics, Mechatronics
- MEMS
- Theoretical and Applied Mechanics
- Dynamical Systems, Control
- Fluid Mechanics
- Engineering Thermodynamics, Heat and Mass Transfer
- Manufacturing
- Precision Engineering, Instrumentation, Measurement
- Materials Engineering
- Tribology and Surface Technology

To submit a proposal or request further information, please contact the Springer Editor of your location:

China: Dr. Mengchu Huang at mengchu.huang@springer.com

India: Priya Vyas at priya.vyas@springer.com

Rest of Asia, Australia, New Zealand: Swati Meherishi at
swati.meherishi@springer.com

All other countries: Dr. Leontina Di Cecco at Leontina.dicecco@springer.com

To submit a proposal for a monograph, please check our Springer Tracts in Mechanical Engineering at <http://www.springer.com/series/11693> or contact Leontina.dicecco@springer.com

Indexed by SCOPUS. The books of the series are submitted for indexing to Web of Science.

More information about this series at <http://www.springer.com/series/11236>

Raviprakash R. Salagame · Palaniappan Ramu ·
Indira Narayanaswamy · Dhish Kumar Saxena
Editors

Advances in Multidisciplinary Analysis and Optimization

Proceedings of the 2nd National Conference
on Multidisciplinary Analysis
and Optimization



Springer

Editors

Raviprakash R. Salagame
APTV
Bengaluru, Karnataka, India

Indira Narayanaswamy
Bengaluru, Karnataka, India

Palaniappan Ramu
Department of Engineering Design
Indian Institute of Technology Madras
Chennai, Tamil Nadu, India

Dhish Kumar Saxena
Department of Mechanical
and Industrial Engineering
Indian Institute of Technology Roorkee
Roorkee, Uttarakhand, India

ISSN 2195-4356

ISSN 2195-4364 (electronic)

Lecture Notes in Mechanical Engineering

ISBN 978-981-15-5431-5

ISBN 978-981-15-5432-2 (eBook)

<https://doi.org/10.1007/978-981-15-5432-2>

© Springer Nature Singapore Pte Ltd. 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721,
Singapore

Conference Committee

Patrons

Dr. M. R. Jayaram, Chancellor, RUAS

Dr. Kota Harinarayana, Design Dvn, AeSI

General Conference Committee

Chair: Dr. Kota Harinarayana, Design Dvn, AeSI

Prof. H. K. Narahari, RUAS

Prof. G. K. Ananthasuresh, IISc

Prof. Palaniappan Ramu, IIT Madras

Organizing Committee

Chair: Prof. H. K. Narahari, RUAS

Prof. Indira Narayanaswamy, RUAS

Rakesh Kumar, ADE

Prof. R. Raja, RUAS

Advisory Committee

Chair: Prof. K. Sudhakar, IIT Mumbai

Sri N. C. Shekar, CFO, RUAS

Prof. Govind R. Kadambi, Pro VC, RUAS

Dr. A. R. Upadhy, Bengaluru

Prof. P. Mannar Jawahar, Karunya University

Prof. Raman Singh, Monash University

Prof. Ashok Belegundu, Penn State University

Prof. R. Raja, RUAS

Technical Program Committee

Chair: Dr. Raviprakash R Salagame, Aptiv
Prof. Palaniappan Ramu, IIT Madras
Mr. P. Jayasimha, Ex HAL and ADA
Prof. Dhish Saxena, IIT Roorkee
Prof. Anupam Saxena, IIT Kanpur
Prof. S. Srikari, RUAS
Mr. Vinay Ramanath, Siemens
Mr. Murali Balasubramaniam, FCA, Chennai
Dr. Santhosh Koruthu, ADA
Dr. Pankaj Priyadarshi, VSSC, ISRO
Dr. Somnath Nagendra, P&W, UTC, USA
Dr. Prajwal Shivaprakasha, DLR, Germany
Prof. H. K. Narahari, RUAS
Dr. Shankar Venugopal, M&M

Publication Committee

Dr. Indira Narayanaswamy, RUAS
Dr. Rahul M. Cadambi, RUAS
Prof. Srikari S., RUAS
Prof. B. V. Vijay, RUAS
Dr. M. Sivapragasam, RUAS
Dr. A. T. Sriram, RUAS

Reviewers

- Dr. Pankaj Priyadarshi, VSSC (ISRO), Trivandrum, e-mail: pankaj_priyadarshi@vssc.gov.in
- Dr. Raviprakash R. Salagame, Aptiv Technical Center, Bengaluru, e-mail: ravi.salagame@aptiv.com
- Prof. Srikari Srinivasan, M. S. Ramaiah University of Applied Sciences, Bengaluru, e-mail: srikari.aae.et@msruas.ac.in
- Dr. Nagendra Somanath, Pratt and Whitney, Connecticut, USA, e-mail: nagendra.somanath@pw.utc.com
- Dr. Srinivasa Gunti, Mahindra and Mahindra, Chennai, e-mail: srinivas.gunti@mahindra.com
- Dr. Santhosh Koruthu, ADA, Bengaluru, e-mail: koruthu@jetmail.ada.gov.in
- Vinay Ramanath, Siemens Corporate Research Center, Bengaluru, e-mail: vinay.ramanath@siemens.com
- Dr. Vinod Bhantia, M. S. Ramaiah University of Applied Sciences, Bengaluru, e-mail: vinod_bhantia@hotmail.com
- Dr. Indira Narayanaswamy, M. S. Ramaiah University of Applied Sciences, Bengaluru, e-mail: indira1204@gmail.com
- Dr. Dhish Saxena, Indian Institute of Technology Roorkee, e-mail: dhishfme@iitr.ac.in
- Dr. Rajib Chowdhury, Indian Institute of Technology Roorkee, e-mail: rajibfce@iitr.ac.in
- Prof. H. K. Narahari, M. S. Ramaiah University of Applied Sciences, Bengaluru, e-mail: narahari.aae.et@msruas.ac.in
- Dr. Saravana Kumar, Indian Institute of Technology Madras, e-mail: gsaravana@iitm.ac.in
- Dr. Subhendu Saha, ADA, Bengaluru, e-mail: subhendusaha2015@gmail.com
- Dr. Sourav Rakshit, Indian Institute of Technology Madras, e-mail: srakshit@iitm.ac.in
- Dr. Biswajit Tripathi, General Motors Technical Centre, Bengaluru, e-mail: biswajit.tripathy@gm.com

Prof. K. Sudhakar (Retd), Indian Institute of Technology Bombay, e-mail: sudhakar.iitb@gmail.com

Prof. Raghavendra Kulkarni, M. S. Ramaiah University of Applied Sciences, Bengaluru, e-mail: raghavendra.ec.et@msruas.ac.in

Dr. Narendran Balan, Boeing India Technical Center, Bengaluru, e-mail: narendran_balan@hotmail.com

Mr. Rajapandian Ramamoorthy, General Motors Technical Centre, Bengaluru, e-mail: rajapandian.ramamoorthy@gm.com

Mr. Sesha Kiran Gundavarapu, Siemens Corporate Research Center, Bengaluru, e-mail: sesha.gundavarapu@siemens.com

Prof. M. Ramanathan, Indian Institute of Technology Madras, e-mail: mraman@iitm.ac.in

Prof. R. Raja, M. S. Ramaiah University of Applied Sciences, Bengaluru, e-mail: raja.aae.et@msruas.ac.in

Prof. Salil Kulkarni, Indian Institute of Technology Bombay, e-mail: salil.kulkarni@iitb.ac.in

Dr. Phani Adduri, VR&D, USA, e-mail: padduri@vrand.com

Dr. Pavan Kumar, DVTG, NAL Bengaluru, e-mail: pavan@nal.res.in

Preface

This book contains selected papers from the 2nd National Conference on Multidisciplinary Design, Analysis and Optimization (NCMDAO-2) which was held at Bengaluru between March 21 and 23, 2019. The conference was held at Ramaiah University of Applied Sciences (RUAS) in collaboration with Design Division of Aeronautical Society of India (AeSI), Mechanical Engineering Department of Indian Institute of Science (IISc) and Engineering Design Department of Indian Institute of Technology (IIT), Madras. This conference was second in annually planned event to create a platform for researchers and industry professionals working in the area of multidisciplinary design, analysis and optimization to share their current work.

The need for optimization is heightened in today's multidisciplinary R&D environment. Optimization helps us push the boundaries to make the most out of limited resources. Aerospace, automotive, manufacturing, biomedical and many other fields significantly benefit from optimization and gain a competitive edge. Advances in optimization theory, algorithms and software have made it possible to improve the performance and economy of components, devices, processes and entire systems. Efficient analysis and design are possible even for those problems where analytical and computational models are not available. Robust design under uncertainty is also pursued vigorously. The conference was organized with these goals in mind.

The response from various government laboratories, academic institutions, aerospace industries and private companies was overwhelming with roughly 140 registered participants attending the two-day conference. Over 30% of the contributed papers were from the manufacturing and consulting industries, while rest of them were from academic institutions and government research laboratories such as ISRO, ADA and DRDO. Over the two days of the conference, about 65 papers were presented out of which over 40 full papers were contributed. These papers were peer-reviewed through an online and offline system, and a total of 27 high-quality papers are being included in this proceedings.

The conference was held over two days with a pre-conference workshop which was introduced to disseminate information on new areas of study in optimization. Pre-conference workshop included four talks and had close to 100 registrants. The session included a talk on advances in machine learning, AI and deep learning by

Dr. Somanath Nagendra, P&W USA; a talk on surrogate modeling techniques by Mr. Mohan Sangli, Intuceo; and followed by a talk on probabilistic design by Mr. Vinay Ramanath, Siemens Corporate Research. Finally, a talk and demo on optimization of composite materials by Mr. Sushil Mane, Altair Engineering, capped end of the master class.

The main conference included three keynote talks from industry stalwarts, two invited talks, sponsor presentations and a panel discussion. One of the keynotes was delivered by Prof. (Dr.) Garret Vanderplaats, CEO of VR&D, and internationally acclaimed professor and researcher in structural optimization. Dr. Vanderplaats presented a recorded talk on perspectives and history of optimization which enthralled the audience with a glimpse of growth of optimization technology and key contributors to the field. The conference agenda was divided into 12 sessions of which six were parallel sessions. Each session had roughly five to six papers. A highlight of the conference was a session for rapid-fire introduction of papers being presented during each of the two halves of the day. During this session, each presenter would speak about his paper for roughly 2 min, providing an opportunity for the participants to select sessions of their interest. Each paper was scheduled based on session theme. The following were the themes of the conference:

Session 1A	MDO and Applications: Aerospace Engineering
Session 1B	Machine Learning and Neural Networks in Optimization
Session 2A	MDO and Applications: Automotive Engineering
Session 2B	Optimization Algorithms and Methods
Session 3A	MDO and Applications: Automotive Engineering
Session 3B	Optimization Algorithms and Methods
Session 4A	Shape and Topology Optimization Applications
Session 4B	Approximation Methods and Surrogate Models
Session 5A	MDO and Applications: Aerospace Engineering
Session 5B	Optimization Algorithms and Methods
Session 6A	MDO and Applications: Automotive Engineering
Session 6B	General Application of Optimization in Engineering

The conference was sponsored by leading industries in the field of optimization such as MSC Software, VR&D (Optisign), Metacomp Technologies, Zeus Numerix, Acer and Siemens besides strong support from SAEINDIA, Aerospace Society of India and Springer.

The papers included in this proceedings were selected based on a rigorous review considering theme of the conference in mind. We hope that this collection of work would ignite further interest in this field in the country and will continue to be active for years to come with increased contribution.

Bengaluru, India

Dr. Raviprakash R. Salagame
Prof. Palaniappan Ramu
Prof. Indira Narayanaswamy
Prof. Dhish Kumar Saxena

Acknowledgements

We thank Dr. Kota Harinarayana who urged us and the entire team to step up MDAO in India, Prof. K. Sudhakar for continued advice and Dr. M.R. Jayaram, Chancellor of RUAS, for wholeheartedly supporting the event. We are grateful to our sponsors (Metacomp Technologies, MSC Software, Zeus Numerix, Acer, SAEINDIA, Siemens, Optisign, Springer and Tamil Nadu Technology Development & Promotion Centre). On the logistics side, we thank the heads and members of the following RUAS research centers: (a) Centre for Aeronautical and Micro Air Vehicles Research, (b) Automotive Technologies Research Centre, (c) Composite Materials and Technologies Research Centre and (d) Structural Design and Analysis Centre. We thank the Department of Automotive and Aeronautical Engineering, Faculty of Engineering and Technology, RUAS. We also thank Indian Institute of Technology Madras and the Design Division of the Aeronautical Society of India for their support.

Editors and Organizing Committee
NCMDAO-2

Contents

Optimization Applications: Aerospace	
Gerlach Shaping Inspired Air Intake Duct Optimization	3
Valliammai Somasundaram and N. Balakrishnan	
Multi-disciplinary Analysis of a Reentry Vehicle	19
K S Lakshmi and Pankaj Priyadarshi	
Optimization of Multiple Gravity Assist Trajectories with Deep Space Maneuver Using Evolutionary Algorithms	33
H. R. Sahana, Pooja Dutt, and A. K. Anilkumar	
Conceptual Design Optimization of High-Altitude Airship Having a Tri-Lobed Envelope	49
M. Manikandan and Rajkumar S. Pant	
Multi-objective Aerodynamic Optimization of a Hypersonic Scramjet Inlet.	63
Lipi Roy and Pankaj Priyadarshi	
Study of Wing Flutter at Preliminary Stages of Design	77
A. T. Sriram and H. K. Narahari	
Coupled Aero-structural Simulation Techniques Using High-Fidelity Analysis and Design Tools	85
Ravi Kumar Kovvali, Beerinder Singh, Srivatsan Lakshminarasimhan, and Sukumar Chakravarthy	
Optimization of B-Spline Launch Vehicle Payload Fairing.	95
K. Nihal and Pankaj Priyadarshi	
Design of Vortex Flaps for Reducing Approach Speed of a Supersonic Naval Fighter Aircraft	109
Soumyajit Behera, C. Muthuraj, Balmiki Kumar, U. R. Srikanth, Sathish Sunnam, and R. Jolly	

Optimization Applications: Automotive

Performance Comparison of Discrete Kalman Filter and Dynamic Programming Technique for Pavement Roughness Identification	121
O. A. Shereena, C. G. Krishnanunni, and B. N. Rao	
Multi-disciplinary Optimization of Sensor Bracket	131
Rajapandian Ramamoorthy and Shabareesh Kumar	
Kriging Driven Optimization of Vehicle Components Using Equivalent Static Load (ESL) Methodology	141
Kshitiz Swaroop and Varun Agarwal	
Reliability-Based Design Optimization of a Door Beam for FMVSS214 Requirements	151
Kshitiz Swaroop, Nihal Kakodkar, and Biswajit Tripathy	
Design Exploration Using Unique Multidisciplinary Design Optimization Framework	161
Garima Singh and N. C. Praveen	
Model-Based Design and Multi-objective Robust Optimization of Electric Vehicle for Performance, Range and Top Speed Objectives	173
Aniket Karandikar, Ashok Kumar Ravi, Elavarasan Dharumaseelan, and Elanchezhian Tamilselvam	
General Applications of Optimization	
Multi-disciplinary Optimization Using ISIGHT to Reduce Thermal Gradients	191
Krishna Nelanti and Venugopal Lingala	
Optimization of Kalman Filter for Target Tracking Applications	203
T. Sunil Kumar and Punithavathi Duraiswamy	
Shape Optimization of Beams with Scaled Boundary Finite Element Method and B-Splines	213
Kandula Eswara Sai Kumar, A. Sundaramoorthy, Sundararajan Natarajan, and Sourav Rakshit	
Innovative Design of Hydraulic Actuation System for Operator Fatigue Reduction and Its Optimization	225
Sukrit Mittal, Divyam Aggarwal, and Dhish Kumar Saxena	
Design of FSAPDS Using Multidisciplinary Design Optimization (OpenMDAO)	235
Rishav Kanth and Basant Kumar Gupta	

Reducing RCS of Aerospace and Marine Vehicles by Shaping	249
Sumit Jana, Mohd. Naimuddin, and Jason Jose	
Performance Analysis and Design Optimization of Two-Baffle Muzzle	
Brake of 155 mm Artillery Gun	255
Anubhav Tiwari and Vighnesh Pawar	
Neural Networks and Machine Learning in Optimization	
Inverse Response Surface Method for Structural Reliability	
Analysis	267
M. Nagesh and A. S. Balu	
Smart Thermal Data Match Using Machine Learning	275
Krishna Nelanti, Suresh Thiraviyam, Naresh Nidamanuri, and Raviraj Barapu	
Identification of Aerodynamic Derivatives of a Re-entry Module	289
C. Priyant Mark and K. Srinivasan	
Topology Optimization Using Convolutional Neural Network	301
Baki Harish, Kandula Eswara Sai Kumar, and Balaji Srinivasan	
Rank Aggregation Approach to Feature Selection for Improved	
Model Performance	309
Mohan Sangli and Anish Ravishankar	

Editors and Contributors

About the Editors



Dr. Raviprakash R. Salagame brings over 25 years of industry experience. He is currently a senior global leader at APTIV Technical Center India. Previously, Ravi held leadership and engineering positions at GE Corporate Research Center (at JFWTC) and IBM Almaden Research Center at San Jose, California. He completed his Ph.D. in Mechanical Engineering from Pennsylvania State University in 1995, Master's degree from Indian Institute of Science in 1988 and Bachelor's degree from BMS College of Engineering in 1986. He is a certified Six Sigma black belt and a technical expert in Design Optimization, CAE, data science and statistical methods. He has published over 15 international journal and conference papers and delivered several industry keynotes in India. He also serves on the board of IUCEE (Indo-Universal Collaboration for Excellence in Engineering Education) and a few of the leading universities in India.



Dr. Palaniappan Ramu is an Associate Professor in the Department of Engineering Design, IIT Madras. He heads the Advanced Design Optimization and Probabilistic Techniques (ADOPT) Laboratory where work revolves around using probability, statistics and optimization for design decisions. He investigates problems of varied interests in different areas. It includes optimal designs of computer models built in FEM/CFD towards light weighting, understanding performance of machine learning algorithms as surrogates, digital twins and visual exploration of design spaces. He holds a B.Eng. in Mechanical from Madurai Kamaraj University, India and a Ph.D. in Aerospace Engineering from University of Florida, USA. He has 10+ years' experience working on optimization for engineering applications. He teaches courses on optimization, reliability, surrogates and DfMA.



Dr. Indira Narayanaswamy is a research Professor in Ramaiah University of Applied Sciences, Bangalore, India. She has previously served as a scientist and technology Director in the Aeronautical Development Agency, Bangalore. Dr. Narayanaswamy's areas of specialization are CFD applications for Aerodynamics Shape Optimization and Multidisciplinary Design Optimization (MDO), MDO for MAV design, Numerical Investigations on Supersonic Retro Propulsion (SRP). She has authored more than 150 research papers and technical reports.



Dr. Dhish Kumar Saxena is an Associate Professor in the Department of Mechanical & Industrial Engineering, IIT Roorkee. His main research interests lie in multi-objective, multi-disciplinary optimization, and multi-criterion decision making. He is currently the principal investigator/co-investigator of different international projects in these domains, focusing on large scale industrial problems. He is also on the Reviewer Board and Project Monitoring Committee member for different national/international projects launched by the Ministry of Electronics & Information Technology, and Department of Science & Technology. Dr. Saxena

brings on board his work experience in the United Kingdom where he worked with universities like Cranfield and Bath, in collaboration with companies like British Aerospace Systems, Hewlett Packard, and Airbus.

Contributors

Varun Agarwal General Motors Technical Centre India, Bengaluru, India

Divyam Aggarwal Department of Mechanical & Industrial Engineering, Indian Institute of Technology Roorkee, Roorkee, India

A. K. Anilkumar Director, Directorate of Space Surveillance and Directorate of Space Situational Awareness and Management (DSSAM), ISRO - HQ, Bengaluru, India

N. Balakrishnan Indian Institute of Science, Bangalore, India

A. S. Balu Department of Civil Engineering, National Institute of Technology Karnataka, Surathkal, India

Raviraj Barapu Cyient Limited, Hyderabad, India

Soumyajit Behera Aeronautical Development Agency, Bangalore, India

Sukumar Chakravarthy Metacomp Technologies Inc., Agoura Hills, CA, USA

Elavarasan Dharmaseelan Modelon Engineering Private Limited, Trichy, India

Punithavathi Duraiswamy Department of Electronics and Communication Engineering, M S Ramaiah University of Applied Sciences, Bangalore, India

Pooja Dutt Applied Mathematics Division, Vikram Sarabhai Space Centre, ISRO, Thiruvananthapuram, India

Kandula Eswara Sai Kumar Indian Institute of Technology Madras, Chennai, Tamil Nadu, India

Garima Singh Altair Engineering India Pvt. Ltd., Bengaluru, India

Basant Kumar Gupta Zeus Numerix Private Limited, Pune, India

Baki Harish Indian Institute of Technology Madras, Chennai, Tamil Nadu, India

Sumit Jana Zeus Numerix Pvt. Ltd, Pune, India

R. Jolly Aeronautical Development Agency, Bangalore, India

Jason Jose Zeus Numerix Pvt. Ltd, Pune, India

Nihal Kakodkar Indian Institute of Technology, Kharagpur, India

Rishav Kanth Zeus Numerix Private Limited, Pune, India

Aniket Karandikar ESTECO Software India Private Limited, Pune, India

Ravi Kumar Kovvali Metacomp Technologies Inc., Agoura Hills, CA, USA

C. G. Krishnanunni Indian Institute of Technology Madras, Chennai, India

Balmiki Kumar Aeronautical Development Agency, Bangalore, India

Kandula Eswara Sai Kumar Indian Institute of Technology, Chennai, India

Shabareesh Kumar General Motors Technical Center India (GMTCI), Bangalore, India

K S Lakshmi Vikram Sarabhai Space Center, ISRO, Trivandrum, Kerala, India

Srivatsan Lakshminarasimhan Metacomp Technologies Pvt. Ltd, Chennai, India

Venugopal Lingala Cyient Limited, Hyderabad, India

M. Manikandan Indian Institute of Technology Bombay, Mumbai, India

C. Priyant Mark Department of Instrumentation and Control Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, KA, India

Sukrit Mittal Department of Mechanical & Industrial Engineering, Indian Institute of Technology Roorkee, Roorkee, India

C. Muthuraj Aeronautical Development Agency, Bangalore, India

M. Nagesh Department of Civil Engineering, National Institute of Technology Karnataka, Surathkal, India

Mohd. Naimuddin Zeus Numerix Pvt. Ltd, Pune, India

H. K. Narahari Ramaiah University of Applied Sciences, Bangalore, India

Sundararajan Natarajan Indian Institute of Technology, Chennai, India

Krishna Nelanti Cyient Limited, Hyderabad, India

Naresh Nidamanuri Cyient Limited, Hyderabad, India

K. Nihal Department of Aerospace Engineering, Indian Institute of Space Science and Technology, Trivandrum, Kerala, India

Rajkumar S. Pant Indian Institute of Technology Bombay, Mumbai, India

Vighnesh Pawar Zeus Numerix Pvt. Ltd, Pune, India

N. C. Praveen Altair Engineering India Pvt. Ltd., Bengaluru, India

Pankaj Priyadarshi Vikram Sarabhai Space Center, ACDD, VSSC, ISRO, Trivandrum, Kerala, India

Sourav Rakshit Indian Institute of Technology, Chennai, India

Rajapandian Ramamoorthy General Motors Technical Center India (GMTCI), Bangalore, India

B. N. Rao Indian Institute of Technology Madras, Chennai, India

Ashok Kumar Ravi Modelon Engineering Private Limited, Trichy, India

Anish Ravishankar ICUBE Consulting Services India Pvt. Ltd, Bangalore, India

Lipi Roy Indian Institute of Space Science and Technology, Thiruvananthapuram, India

H. R. Sahana Aerospace Engineering, Manipal Institute of Technology, MAHE, Manipal, India

Mohan Sangli ICUBE Consulting Services India Pvt. Ltd, Bangalore, India

Dhish Kumar Saxena Department of Mechanical & Industrial Engineering, Indian Institute of Technology Roorkee, Roorkee, India

O. A. Shereena Indian Institute of Technology Madras, Chennai, India

Beerinder Singh Metacomp Technologies Inc., Agoura Hills, CA, USA

Valliammai Somasundaram Aeronautical Development Agency, Bangalore, India

U. R. Srikanth Aeronautical Development Agency, Bangalore, India

Balaji Srinivasan Indian Institute of Technology Madras, Chennai, Tamil Nadu, India

K. Srinivasan Wind Tunnel Data Division, Wind Tunnel Group, Aeronautics Entity, Vikram Sarabhai Space Centre, Thiruvananthapuram, KL, India

A. T. Sriram Ramaiah University of Applied Sciences, Bangalore, India

A. Sundaramoothy Indian Institute of Technology, Chennai, India

T. Sunil Kumar Department of Electronics and Communication Engineering, M S Ramaiah University of Applied Sciences, Bangalore, India

Sathish Sunnam Aeronautical Development Agency, Bangalore, India

Kshitiz Swaroop General Motors Technical Centre India, Bengaluru, India

Elanchezhian Tamilselvam Modelon Engineering Private Limited, Trichy, India

Suresh Thiraviyam Cyient Limited, Hyderabad, India

Anubhav Tiwari Zeus Numerix Pvt. Ltd, Pune, India

Biswajit Tripathy General Motors Technical Centre India, Bengaluru, India

Optimization Applications: Aerospace

Gerlach Shaping Inspired Air Intake Duct Optimization



Valliammai Somasundaram and N. Balakrishnan

Abstract Due to the importance of survivability and low observability in the next-generation aircraft and significant contribution of rotating engine components to RCS, inhibiting direct line of sight of aerodynamic interface plane (AIP) is the main design focus of engine air intake duct. This constraint leads to a highly curved shape inducing strong secondary flows, consequently taking engine face distortion and pressure recovery to unacceptable levels. With an objective to eliminate the secondary flow causing lateral pressure gradient without reducing the velocity of the main flow, Gerlach derived a correction factor for modifying the duct cross-sectional shape. Inspired by the performance enhancement this simple procedure could achieve in case of serpentine air intake, the current work aims at evolving a computational fluid dynamics-based optimization procedure for the design of highly curved ducts exploiting the fundamental principle of Gerlach shaping. The highlight of the proposed procedure is that the duct is evolved employing an inviscid modeling of the flow and the candidate optimal configuration thus selected exhibits significantly better performance as compared to the baseline configuration.

Keywords Gerlach shaping · Secondary flow · Intake distortion

1 Introduction

Current generation aircraft have air intakes with short length, high diffusion, severe curvature as opposed to the conventional long air intake ducts with small diffusion and large radius of curvatures. These features induce secondary flows due to lateral pressure gradients and thereby probably flow separations too. A simple technique was proposed by Gerlach and Schroeder [1] in 1969 to minimize radial pressure

V. Somasundaram (✉)
Aeronautical Development Agency, Bangalore, India
e-mail: valli@jetmail.ada.gov.in

N. Balakrishnan
Indian Institute of Science, Bangalore, India
e-mail: nbalak@iisc.ac.in

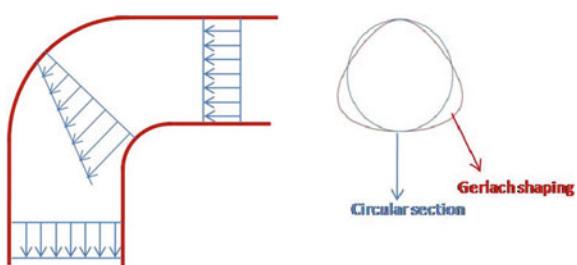
gradient in pipe bends, and its use in air intake duct design has been reported in the open literature [2, 3]. Gerlach shaping works on morphing of the channel or duct cross section in such a way that the radial pressure gradient responsible for the onset of secondary flow is effectively reduced without significantly compromising on the mean fluid velocity.

This paper presents intake distortion reduction techniques derived from Gerlach shaping. The highlight of this procedure is that it finds an inviscid solution to a problem which otherwise is understood to be intensely viscous. This in itself is not surprising as the problem becomes viscous dominant on boundary layer separation, while the radial pressure gradient is the trigger for separation are still consequences of inviscid flow physics.

2 Gerlach Shaping

Gerlach shaping technique proposed by Gerlach et al. [1] is an effective way of minimizing radial pressure gradient, thereby reducing secondary flows. Consider Fig. 1, which illustrates the flow of an ideal fluid through a bend. To sustain the radial acceleration resulting from flow turning, a radial pressure gradient sets in. Consequently, inner wall pressure is lower than outer wall pressure. This requires velocity near the inner wall is greater than that near outer wall. Due to this pressure gradient, low energy fluid migrates toward the inside of the bend and high energy flow moves toward the outside establishing a secondary flow. In order to minimize this radial pressure gradient and the resulting secondary flow, it is necessary to increase the outer wall velocity and to decrease the inner wall velocity. Gerlach shaping achieves this by the way of modifying the duct cross-sectional shape in such a way that the desired local flow accelerations and decelerations are produced. In other words, the new design has an increased area on the inner side and decreased area on the outer side of the bend. Toward realizing this objective, a local geometric duct correction factor named $C(r)$ is defined and the width of the duct at any given radial location r is modified by $C(r)$. For example, as a consequence of Gerlach shaping, a circular section is expected to be transformed to a shape as depicted in Fig. 1.

Fig. 1 Flow across a bend and illustration of shaping



2.1 Application on RAE M2129

RAE M2129 S-duct is a serpentine double-bend circular duct for which experimental and computational data is widely available in the literature [4]. Flow in this duct is a standard test case widely used for CFD validation. Gerlach shaping is applied to RAE M2129 S-duct as a proof of concept. Original RAE M2129 and modified geometries are shown in Fig. 2. Flow inside the two ducts (original and modified) is simulated for AGARD and NPARC test cases. Grid is generated using commercial software, ICEMCFD, and flow simulations are carried out using Aeronautical Development Agency(ADA)'s in-house RANS solver VISP3D.

To highlight the benefits of Gerlach shaping, duct flow features for NPARC test cases are detailed in this section. Figure 3 presents pressure distribution on the duct symmetry plane, and the low pressure zone present at the first bend bottom is eliminated by shaping. It can be observed from Fig. 4 showing surface streamline that massive flow separation usually present in RAE duct is also avoided. Significant enhancement of air intake duct performance especially in total pressure distortion indices (refer Fig. 5) is achieved by Gerlach shaping.

3 Current Optimization Framework

The present work demonstrates a novel optimization framework based on Gerlach shaping for the design of serpentine air intakes. In the true spirit of Gerlach shaping, the basic design is allowed to evolve based on inviscid CFD and the final optimal

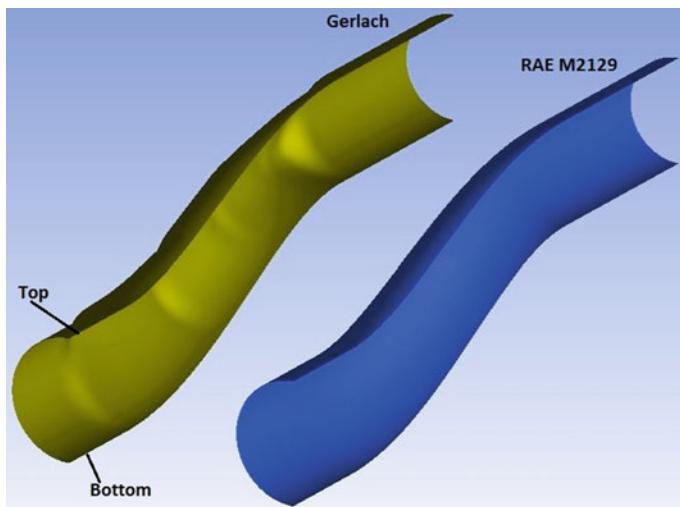


Fig. 2 Original and modified duct geometries

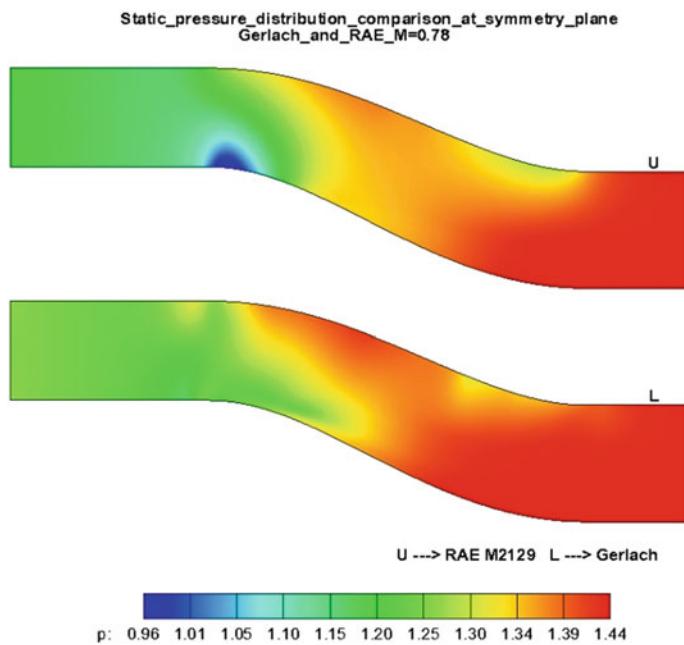


Fig. 3 Pressure distribution on the symmetric plane

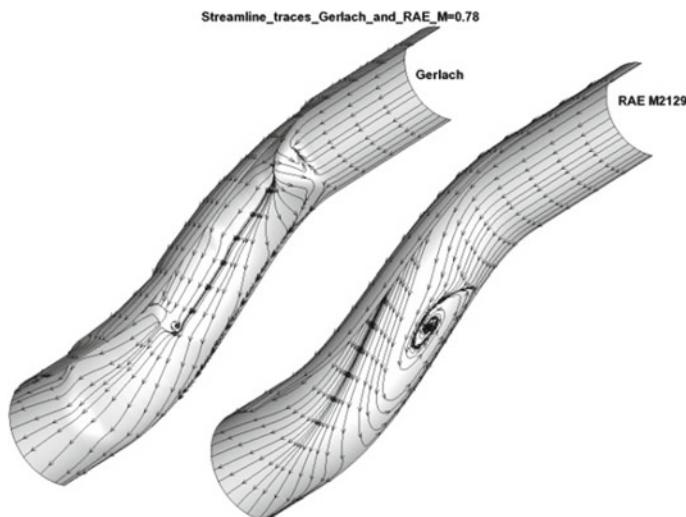


Fig. 4 Surface streamline

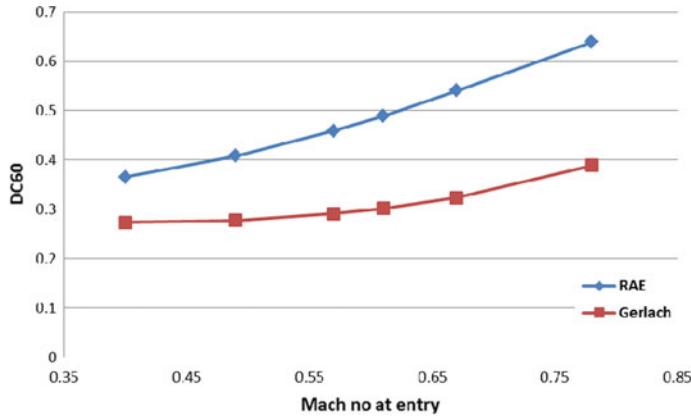


Fig. 5 Distortion characteristics improvement

design is vetted with a RANS solver. It is worth noting that, while the Gerlach shaping is purely based on heuristic arguments, the proposed design methodology relies on a finer description of fluid flow as obtained from a modern CFD tool. In our view, this step is critical, as it significantly enhances the accuracy of the design. This has been achieved primarily because of the availability of supercomputing platforms and the possibility of process automation with the present-day CFD tools.

Objective function inspired by principle of Gerlach shaping emphasizes the uniformity of radial acceleration at each cross section along the duct length. The current air intake duct design is multidisciplinary driven by both aerodynamics and stealth requirements. Therefore, a broad definition of the problem is as follows:

Minimize: An appropriate norm of variation of centrifugal acceleration across the duct cross sections, subject to the following design constraints:

1. Invisibility of aerodynamic interface plane (AIP) for low signature requirements
2. Diffusing duct
3. S-shaped centerline with maximum offset
4. Fixed throat and AIP areas, AIP offset, duct length.

Although the present design optimizes air intake duct for a given forebody by the way of including the forebody in the computations, an integrated design where the forebody is also optimized along with intake duct is a distinct possibility with the proposed optimization process and may be considered in the future. A genetic algorithm-based optimization available in the open-source optimizer DAKOTA is used in the design. An automated CFD process involving grid generation tool Gmsh and inviscid modules of the solver SU2 has been established for function evaluation. The optimal design resulting from the inviscid analysis is vetted using the RANS solver VISP3D.

4 Inviscid Solver for Duct Design—A Working Concept

Duct design optimization requires repeated CFD simulations on various design candidates. High-fidelity computations like RANS consume lots of resources in terms of grid size, CPU time, etc. Euler computations on the other hand are very economical. The big question now is about the adequacy of Euler simulations in air intake duct design using Gerlach principles.

Verification of the adequacy of Euler simulations for Gerlach shaped duct design is carried out in two steps. Effectiveness of CFD integrated Gerlach shaping procedure based on RANS simulations is at first demonstrated on RAE M2129 S-duct. Adequacy of Euler simulations for duct design is proved by replacing RANS simulation in by Euler simulations. As a post-procedure analysis while using Euler solver, effect of the shaping is verified by RANS simulations.

4.1 *Gerlach Shaping Using CFD Simulated RANS Flow Field*

Gerlach area shaping formulation uses a simple empirical formula for velocity distribution. As a part of the current task, we here reformulate the duct correction factor using velocity profile from high-fidelity CFD computations. Velocity profile from RANS at the symmetry plane is considered for calculating the average centrifugal force across a duct section. The average centrifugal force is considered to be the desired constant centrifugal force across the duct. For the centrifugal force to be constant across the duct, new required velocity profile is computed and duct width is modified so as to obtain the required velocity profile.

4.1.1 Implementation

RAE M2129 S-duct is a test case used to demonstrate CFD integrated Gerlach shaping. RANS simulation of flow field inside RAE duct shows a flow separated region. Since Gerlach formulation basically assumes inviscid flows and flow separations cannot be obviously handled, an iterative procedure is adopted wherein shaping is applied only to the flow attached regions. To evaluate the duct performance and obtain flow field parameters for shaping, RANS simulations are carried out using VISP3D. Shaping is applied from the entry section to a section ahead of the separation.

4.1.2 Execution

CFD integrated Gerlach shaping procedure requiring 5 iterations has enhanced the duct performance significantly. Figures 6, 7, 8, 9, 10 and 11 are presented here to bring out the improvement in duct flow field characteristics. Figure 6 shows pressure distribution confirms reduction of pressure gradient in the radial direction, and the main objective of Gerlach shaping is achieved. It can be seen from the x-velocity plot presented in Fig. 7 that the high-speed turning at the bottom of first bend slowed down. With every iteration, the flow separation region diminished in size and also moved downstream (represented by the reversed flow region/negative u region in Fig. 8). It is hardly visible in iterations 4 and 5. Total pressure distribution along the duct symmetric plane (Fig. 9) shows the size of low total pressure zone reducing with iteration. Total pressure distribution at AIP (Fig. 10) shows that the low pressure zone has reduced in intensity and size and also it has spread, thereby lowering the distortion index. Change in the entry shape with iterations can be seen in Fig. 11.

Pressure recovery and distortion coefficient dc60 for each iteration are given in Table 1.

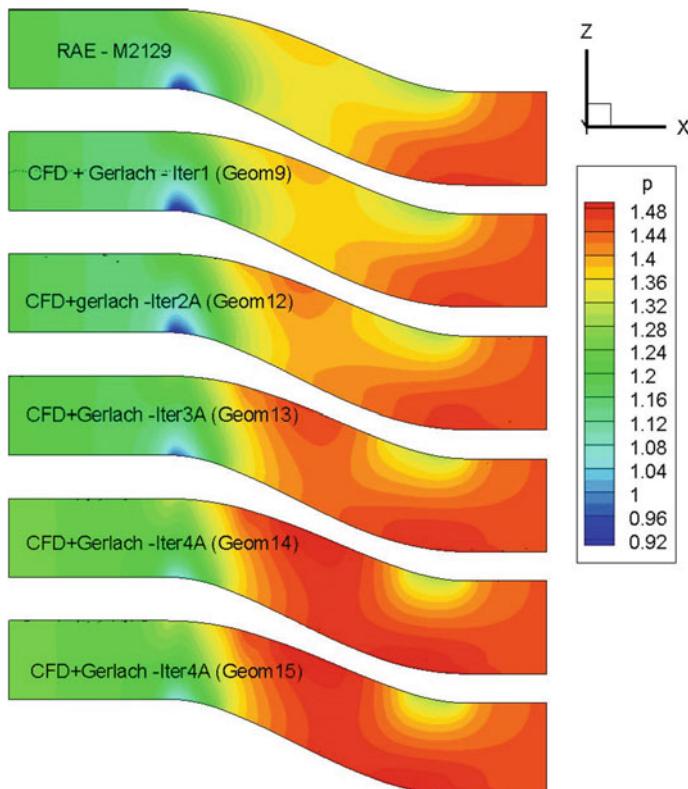


Fig. 6 Pressure distribution on the symmetric plane

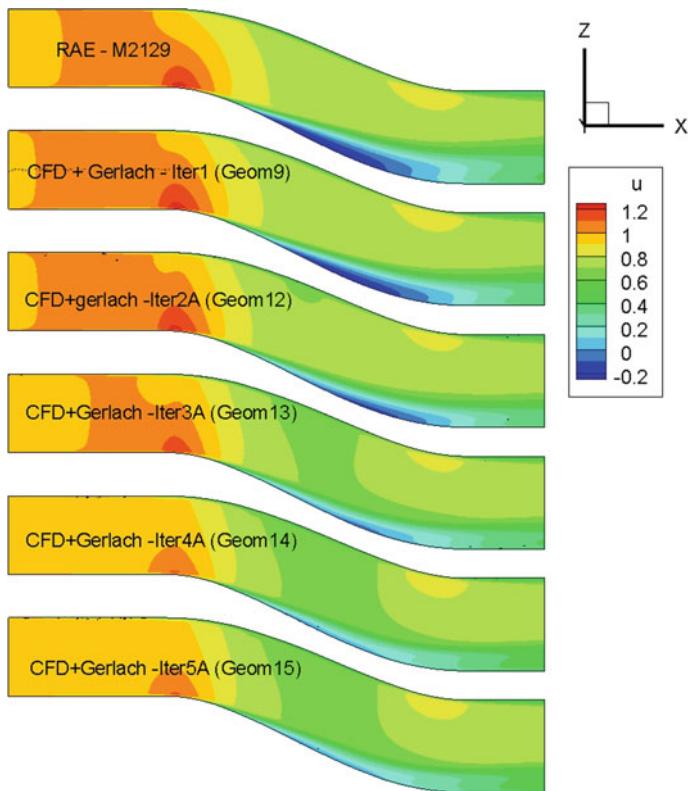


Fig. 7 u distribution

4.2 *Gerlach Shaping Using CFD Simulated Inviscid Flow Field*

4.2.1 Implementation

An iterative procedure is developed to progressively Gerlach shape the duct using Euler flow field and is applied on RAE M2129 S-duct. An automated framework is created with necessary tools, Gmsh for grid generation, SU2 for Euler simulations and Gerlach shaper for geometry morphing. High-fidelity RANS simulations are carried out to ascertain the benefits of Euler-based Gerlach shaping technique. ICEMCFD and VISP3D are used for grid generation and RANS simulations, respectively.

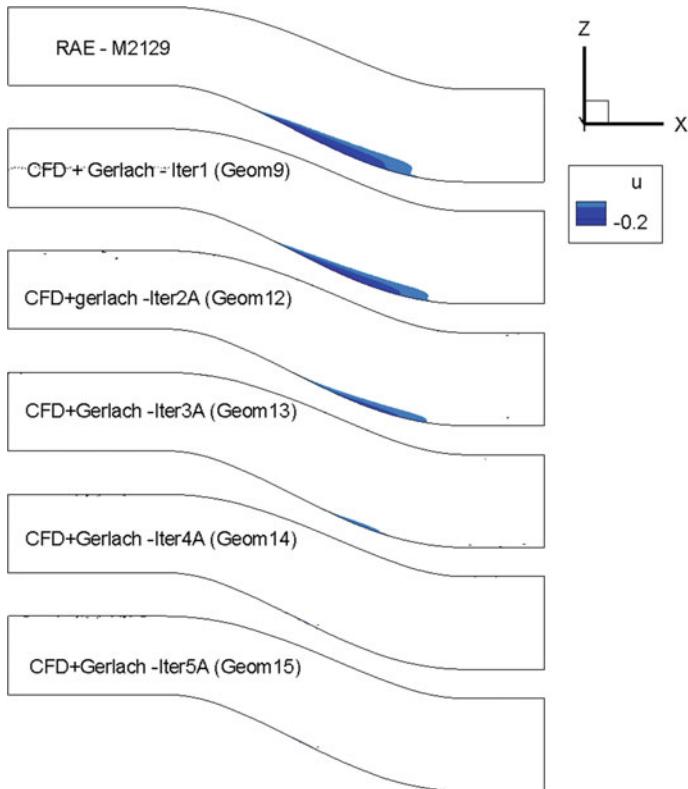


Fig. 8 Negative u region

4.2.2 Execution

CFD integrated Gerlach shaping procedure using Euler simulations requiring 6 iterations has enhanced the duct performance significantly. Figures 12 and 13 show the evolution of duct shape and its inviscid flow field characteristics. The flow acceleration at the first bend has been reduced to a large extent. As a post-procedure analysis, RANS simulations are carried out inside the duct. Total pressure distribution along the duct symmetric plane (Fig. 14) shows the reduced size of low total pressure zone. The flow separation region diminished in size and also moved downstream. This can be observed in the negative u -velocity component region (Fig. 15). Pressure recovery and distortion coefficient $dc60$ are given in Table 2.

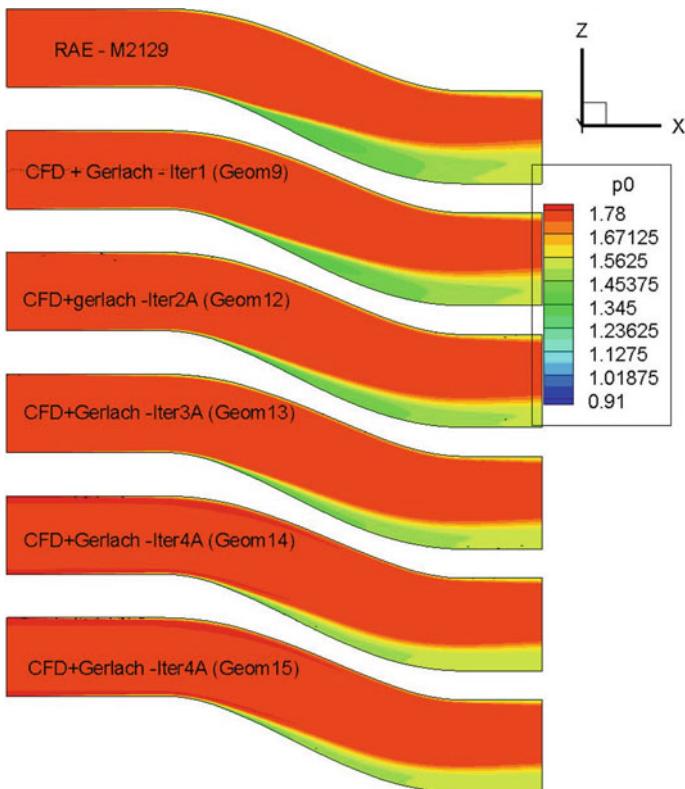


Fig. 9 p_0 across the duct

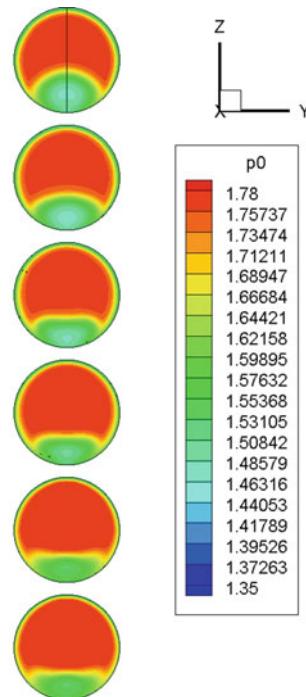
4.3 A Working Concept

A methodology to integrate Gerlach shaping with flow field based on CFD simulations is established. Huge benefits derived from the proposed procedure are demonstrated on the standard test case for internal flows, i.e., RAE M2129 S-duct. The adequacy of Euler simulations for Gerlach shaped duct design using Gerlach is verified by the current exercise.

5 Results and Discussion

Duct created using Gerlach shaping-inspired optimization shown in Fig. 16 is evaluated using ADA's in-house RANS solver VISP3D. Main features examined for acceptance are flow separation, distortion and pressure recovery. A test case simulating high speed and high mass flow rate is used to showcase the optimization

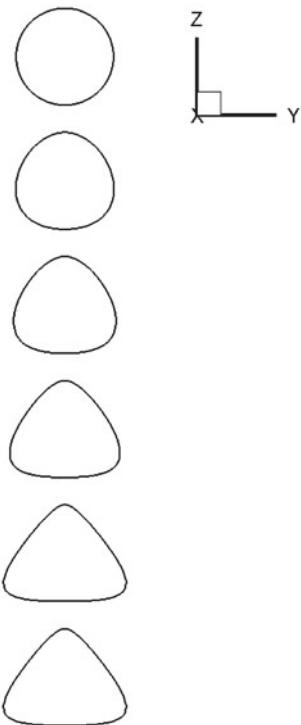
Fig. 10 Total pressure contours at AIP



benefits. The total pressure distribution in the symmetric plane shown in Fig. 17 demonstrates minimal total pressure loss due to the serpentine nature of the duct. It can be seen from the pressure distribution and streamline plot in Fig. 18 that the pressure across duct is almost uniform and there is no flow separation.

6 Conclusion

An optimization procedure inspired by Gerlach shaping is demonstrated for a serpentine duct design. Though duct flow is viscous and dominant, this procedure uses inviscid flow field for its design followed by RANS simulations as a post-optimization analysis. In an independent exercise, viability of concept of inviscid CFD solver use for duct design is proved.

Fig. 11 Entry section**Table 1** Evolution of duct performance parameters

Iteration	Pressure recovery (%)	DC60
0	95.96	0.674754
1	96.13	0.611487
2	96.55	0.512104
3	97.23	0.452906
4	98.12	0.406016
5	98.12	0.406016

Fig. 12 Pressure distribution on the symmetric plane

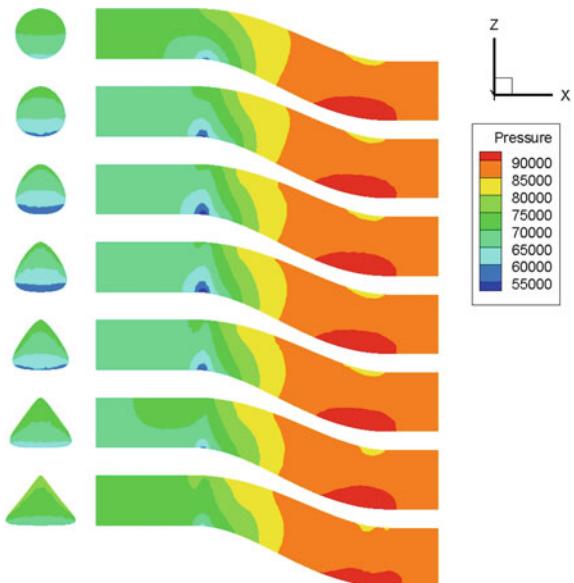
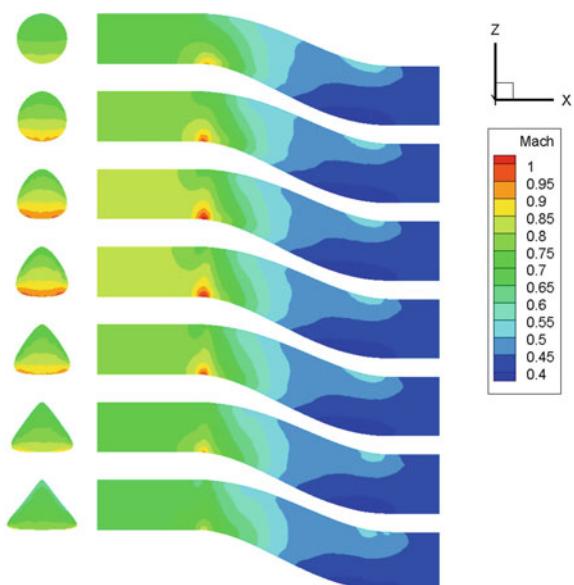


Fig. 13 Mach number contour



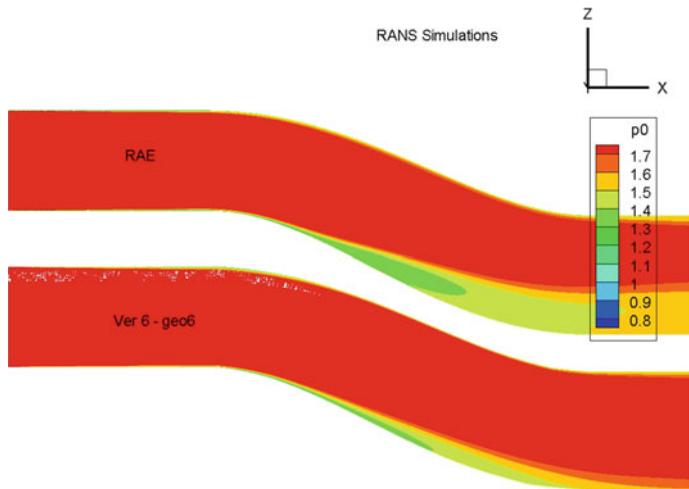


Fig. 14 Total pressure contour

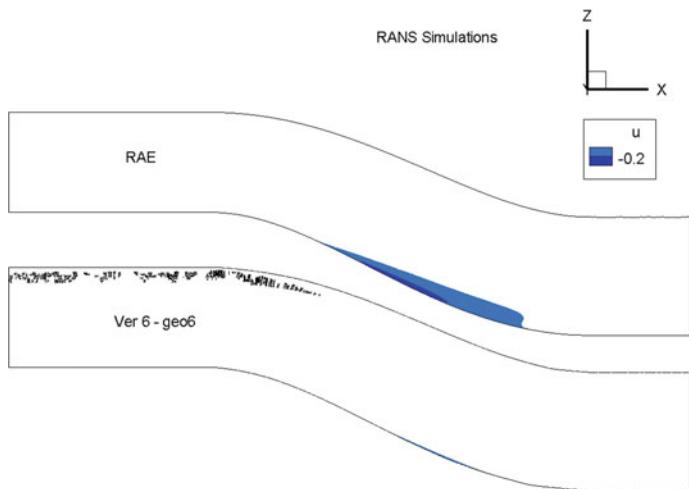


Fig. 15 Negative u region

Table 2 Evolution of duct performance parameters

Geometry	Pressure recovery (%)	DC60
RAE	95.99	0.656398
Shaped-Iter6	95.94	0.240414

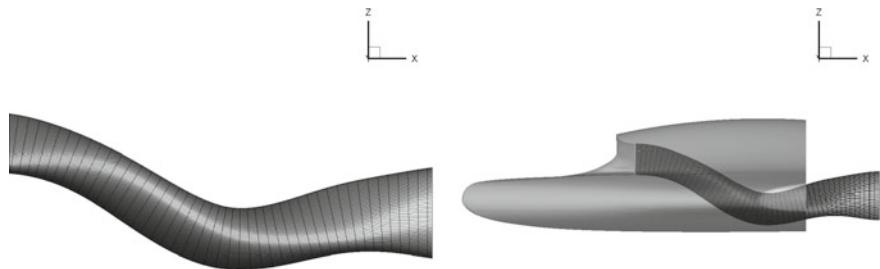


Fig. 16 Optimized duct geometry with forebody

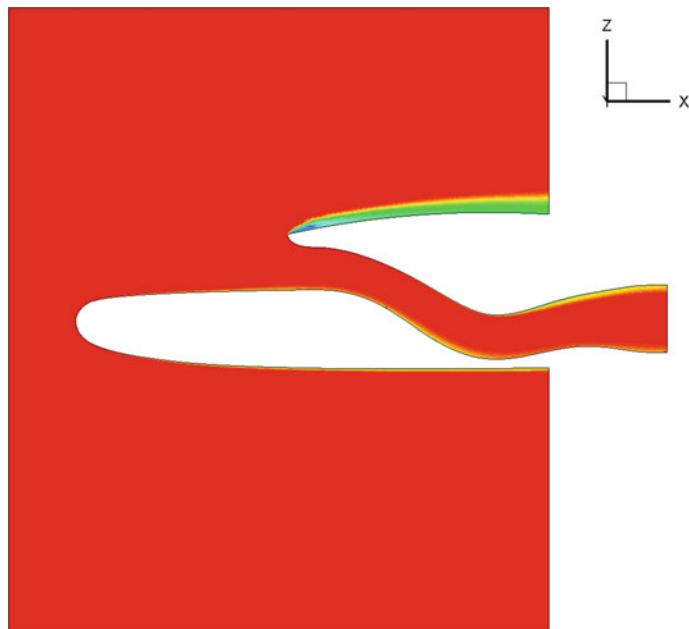


Fig. 17 Total pressure along the symmetric plane

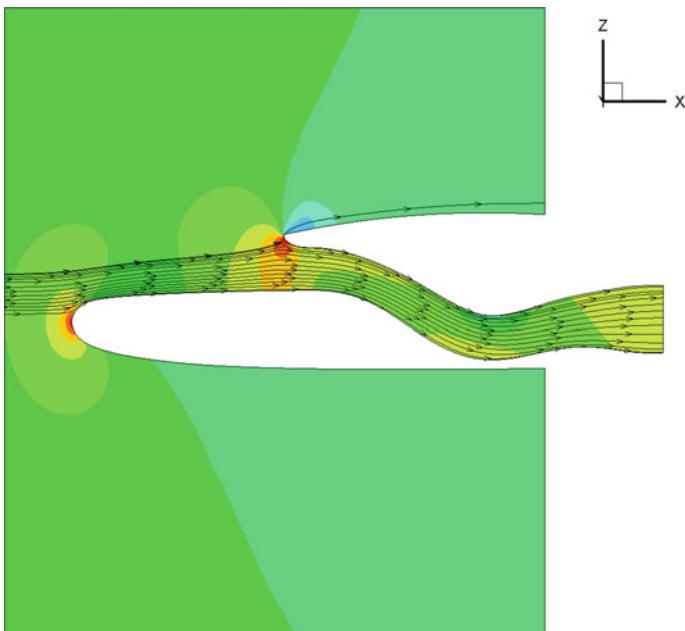


Fig. 18 Pressure and streamline on the symmetric plane

References

1. C.R. Gerlach, E.C. Schroeder, Study of minimum pressure loss in high velocity duct systems. Technical Report No 1, Contract No. NAS8-21133 (NASA, 1969)
2. C.C. Lee, C. Boedicker, Subsonic diffuser design and performance for advanced fighter aircraft, AIAA-85-3073
3. F. Papadopoulos, I. Valakos, I.K. Nikolos, Design of an S-duct intake for UAV applications. *Aircr. Eng. Aerosp. Technol. Int. J.* **84**, 439–456 (2012)
4. Fluid Dynamics Panel, Working group 13. 1991 Test Case 3—Subsonic/Transonic Circular Intake, AGARD Advisory Report 270

Multi-disciplinary Analysis of a Reentry Vehicle



K S Lakshmi and Pankaj Priyadarshi

Abstract During spacecraft reentry into earth atmosphere, it has to encounter wide range of Mach numbers varying from hypersonic to subsonic. For safety of the payload/crew, vehicle structure should be maintained within certain temperature limits. Further, there are significantly high deceleration levels on the body. In the present work, software package is developed which carries out multi-disciplinary analysis (MDA) of axi-symmetric reentry vehicle configuration with trajectory and aerodynamic heating as two participating disciplines. The geometry parameters and initial conditions are perturbed for ballistic and semi-ballistic entries, and their influence is seen on reentry trajectory and aerodynamic heating rates. This work is useful for quick assessment of the data for various configurations during reentry vehicle's initial design phase.

Keywords Trajectory · Aerodynamic heating · Thermal response analysis

Nomenclature

C_P	Specific heat (J/kgK)
D	Drag (N)
Fo	Fourier number
g	Acceleration due to gravity (m/s ²)
h	Heat transfer coefficient (W/m ² K)
k	Thermal conductivity (W/mK)
L	Lift (N)
m	Mass (kg)
n	Temporal index
q	Heat flux (W/m ²)
r	Radial distance from center of earth (m)

K. S. Lakshmi · P. Priyadarshi
Vikram Sarabhai Space Center, ISRO, Trivandrum, Kerala, India
e-mail: lakshmi_kota@vssc.gov.in

R_e	Radius of earth (m)
T	Thrust (N)
t	Time (s)
T	Temperature (K)
V	Velocity (m/s)
x	Distance between nodes (m)
A	First material
B	Second material
L/D	Lift-to-drag ratio
β	Ballistic coefficient (kg/m^2)
γ	Flight path angle ($^\circ$)
Δ	Insulation thickness (m)
ρ	Density (kg/m^3)
φ	Latitude ($^\circ$)
ψ	Heading angle ($^\circ$)
Θ	Longitude ($^\circ$)
σ	Bank angle ($^\circ$)

Subscripts

PL	Payload
S	Structure
TPS	Insulation
i	Spatial index

1 Introduction

Reentry involves dissipation of higher specific energy of the spacecraft when it reenters through dense earth atmosphere. Typical reentry velocities vary from ~ 11 to ~ 7 km/s for planetary and low earth orbit missions. During reentry, part of vehicle's total energy is transferred to the vehicle as aerodynamic heating. Reentry temperatures can be as high as 10,000 K, and deceleration levels of the vehicle can be as high as 10 g and can be even higher depending on entry flight path angle for ballistic missions. If the initial flight path angle is steep, then the flight time and heat load are lower. However, peak heat flux and deceleration levels are higher. For safe landing of the payload/crew, vehicle should be maintained within certain structural/temperature limits. Hence, large numbers of studies are required for trade-off.

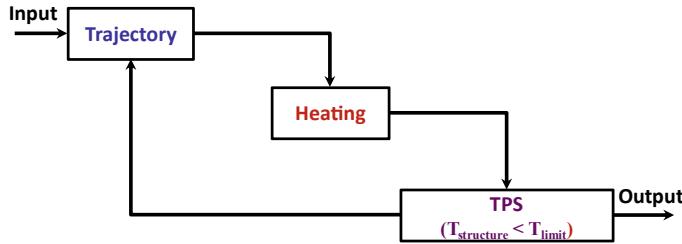


Fig. 1 Design structure matrix (DSM) for multi-disciplinary analysis of trajectory and heating

This paper briefs about the **multi-disciplinary analysis (MDA)** carried for an axi-symmetric reentry vehicle configuration. Two different disciplines involved are reentry trajectory simulation and aerodynamic heating rates estimation with transient thermal response analysis. An optimal insulation thickness is achieved so as to restrict vehicle's structural temperature within its specified limits. Figure 1 shows the design structure matrix (DSM) of multidisciplinary analysis carried out for a reentry vehicle.

2 Methodology

2.1 Reentry Trajectory

Reentry trajectories can be classified based on the lifting characteristics, geometry, and initial state. Based on lifting characteristics, atmospheric reentries are classified as: (a) ballistic (b) semi-ballistic, and (c) lifting entries. Figure 2 shows coordinate system: 'x' axis is along the position vector, 'y' axis is in equatorial plane, and \vec{V} —velocity vector

Primary factors affecting reentry trajectory system are as follows:

(a) ballistic coefficient (β), (b) lift-to-drag ratio (L/D), and (c) flight path angle (γ_E) at reentry.

The equations governing the trajectory of the body while considered as a point mass are given below

Kinematic Equations

$$\frac{dr}{dt} = V \sin \gamma \quad (1)$$

$$\frac{d\theta}{dt} = \frac{V \cos \gamma \cos \psi}{r \cos \varphi} \quad (2)$$

$$\frac{d\varphi}{dt} = \frac{V \cos \gamma \sin \psi}{r} \quad (3)$$

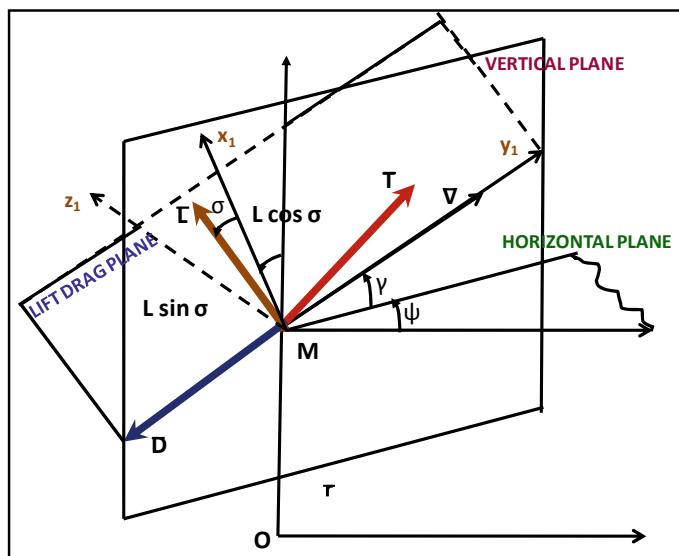
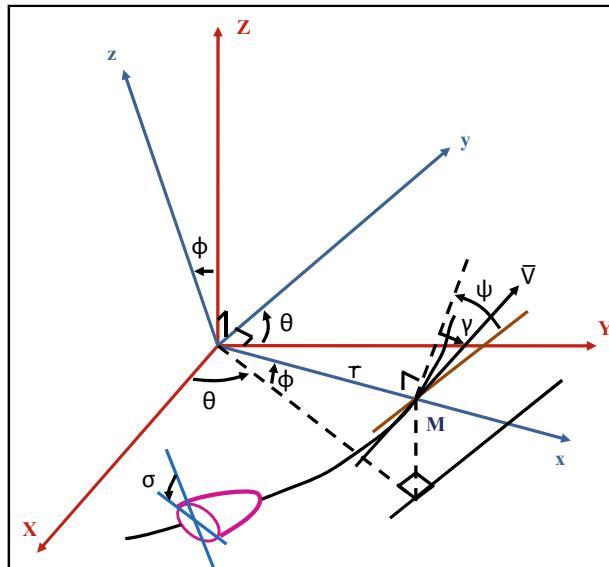


Fig. 2 Coordinate system used for the analysis

Dynamic Equations

$$\frac{dV}{dt} = -g \sin \gamma - \left(\frac{D}{m} \right) \quad (4)$$

$$\frac{d\gamma}{dt} = \left(\frac{L \cos \sigma}{m V} \right) - \left(\frac{g \cos \sigma}{V} \right) + \left(\frac{V \cos \sigma}{r} \right) \quad (5)$$

$$\frac{d\psi}{dt} = \left(\frac{L \sin \sigma}{m V \cos \gamma} \right) - \left(\frac{V \cos \gamma \cos \psi \tan \varphi}{r} \right) \quad (6)$$

Acceleration due to gravity wrt altitude is given as:

$$g = g_0 \left(\frac{R_e}{r} \right)^2 \quad (7)$$

Based on these equations, a three degree of freedom computational model is developed to get time histories of various trajectory parameters [1]. These differential equations are solved numerically using Runge–Kutta method. For the analysis, Indian atmospheric model is considered.

Methodology of Heat Flux Estimation

Based on the simulated reentry trajectory, flow field properties are computed using iterative procedure of normal/oblique shock waves [2] considering real gas effects using Srinivasan and Tannehill model [3]. Based on the trajectory parameters such as altitude and reentry velocity, geometry parameters and flow field properties aerodynamic heating rates are estimated using engineering methods.

Aerodynamic Heating

Reentry vehicle is divided into a combination of simple shapes, such as sphere for the nose and flat plate for the aft body. Engineering methods such as Fay and Riddell method [4] for stagnation point and Eckert's reference enthalpy method [5] for flat plate are used for heat flux estimation at different axial locations of the body.

Vehicle's structure should be provided with suitable thermal protection system (TPS) to protect it from aerodynamic heating. For this, one-dimensional transient thermal response analysis (for two materials A and B) model is developed to estimate temperatures of the structure (material B) with suitable thickness of thermal insulation (material A). One-dimensional transient heat conduction equation is given in [6]

$$\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) = \rho C_p \frac{\partial T}{\partial t} \quad (8)$$

Heat conduction equation is numerically discretized using implicit scheme. Boundary conditions imposed are: net heat flux on first node and convection losses

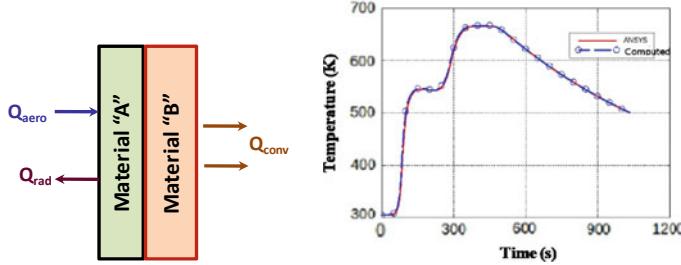


Fig. 3 Comparison between computed backwall temperature histories

on last node. General form of equation at any node ‘*i*’ is given as:

$$T_i^n = [-Fo]T_{i-1}^{n+1} + [1 + 2Fo]T_i^{n+1} + [-Fo]T_{i+1}^{n+1} \quad (9)$$

The resulting system is tri-diagonal in nature. Thomas algorithm is used for solving the system of equations. Mathematical representation of equation in typical matrix form is:

$$\begin{bmatrix} 1 + 2Fo & -2Fo & 0 & 0 \\ -Fo & 1 + 2Fo & -Fo & 0 \\ 0 & -Fo & 1 + 2Fo & -Fo \\ 0 & 0 & -2Fo & 1 + 2Fo \end{bmatrix} = \begin{bmatrix} T_0^{n+1} \\ T_1^{n+1} \\ T_2^{n+1} \\ T_3^{n+1} \end{bmatrix} = \begin{bmatrix} T_0^n + \frac{2\dot{q}\Delta t}{\rho_A C_{PA} \Delta x} \\ T_1^n \\ T_2^n \\ T_3^n + \frac{2hT_\infty \Delta t}{\rho_B C_{PB} \Delta x} \end{bmatrix} \quad (10)$$

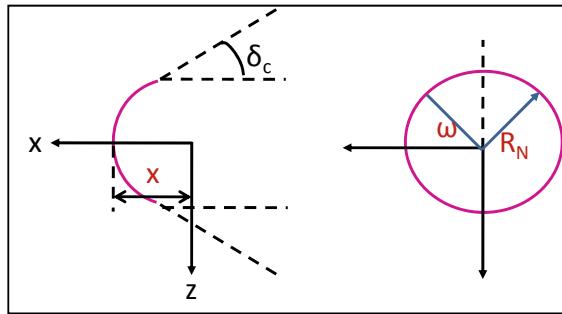
Temperature histories at different locations/nodes are obtained. Aerodynamic heating is applied as boundary condition with graphite as insulation material which does not reach ablating temperature and aluminum alloy [7] as vehicle’s structural material. Developed transient heat conduction model is validated with finite element software ANSYS [8]. Figure 3 shows the schematic of insulation/structure materials with boundary conditions and comparison of computed backwall temperature histories.

Based on these models reentry trajectory, aerodynamic heating and transient thermal response analysis are estimated for typical sphere cone geometry [9, 10]. Before carrying out the multidisciplinary analysis, all the developed modules were validated. Sensitivity studies have been carried out by varying initial state and geometric parameters, and its influence on deceleration and heating levels is seen.

3 Multi-disciplinary Analysis (MDA)

Typical sphere cone geometry considered for the analysis is shown in Fig. 4. In MDA, geometry parameters such as cone angle (δ_c) and nose radius (R_N) are perturbed for ballistic entries, and initial conditions such as flight path angle (γ_E) and mass (m)

Fig. 4 Typical sphere cone geometry



are perturbed for lifting entries. However, aerodynamic characteristics (L/D) are kept constant for these studies. For the reentry vehicle, materials considered are: aluminum alloy (vehicle's structure) and graphite (insulation).

3.1 Variation in Geometry Parameters

3.1.1 For Ballistic Entries

Trajectory parameters of altitude vs relative velocity and deceleration levels (Fig. 4a, b) are generated considering ballistic coefficient (m/C_{DA}) of 555.56 kg/m^2 ($m = 4000 \text{ kg}$) and $\gamma_E = -2^\circ$. For these conditions, peak deceleration level for ballistic entry is $\sim 80 \text{ m/s}^2$.

Using these trajectory parameters, aerodynamic heating levels are computed by perturbing geometry parameters such as nose radius and cone angle. It is seen that increase in R_N from 0.1 m to 2 m results in decrease of peak heat flux by 80% as shown in Fig. 5c. Hence, blunt geometries are preferred when compared with sharp geometries. Also, as the cone angle is increased, heating rates are increased as shown in Fig. 5d. Hence, lower cone angles are preferred for low heating rates. However, there will be change in the drag which will change the trajectory, but this has not been incorporated in this data.

3.1.2 For Semi-ballistic Entries

For semi-ballistic entry with $L/D = 0.2$, it was considered keeping the same ballistic coefficient (555.56 kg/m^2) and flight path angle. It is seen that for semi-ballistic entries, deceleration and heat flux levels ($R_N = 2.0 \text{ m}$) are lower as seen in Fig. 6b, c. Though semi-ballistic entries have lower deceleration and heat flux levels compared to ballistic entries, semi-ballistic entries take longer duration to reach earth's surface and has significant influence on total heat load.

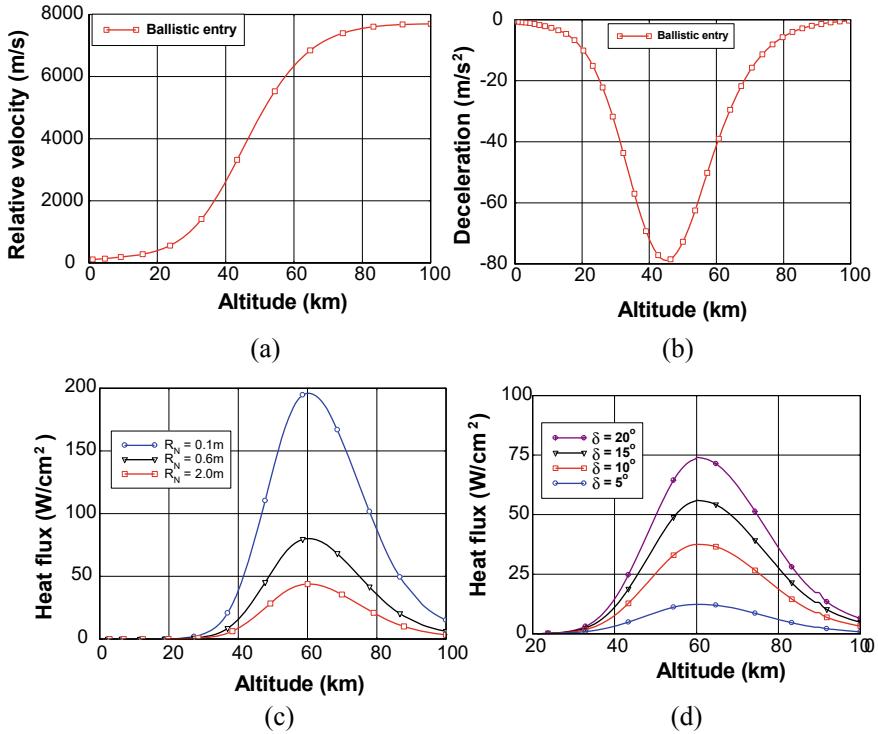


Fig. 5 Ballistic reentry trajectory parameters and influence on heating histories with variation in geometry parameters

3.1.3 Effect of Initial State for Lifting Entries

Initial conditions of trajectories are perturbed wrt semi-ballistic trajectory as shown in Fig. 6, and influence on deceleration and stagnation heating rates is seen.

1. Variation in Flight path angle:

Variation of entry flight path angle from -4° to -1° with same semi-ballistic entry conditions as shown in Fig. 6 is considered. Increase in flight path angle results in steeper trajectories ($\gamma = -4^\circ$) and resulting in higher deceleration and stagnation heating rates as shown in Fig. 7b, c. Increase in γ_E from -1° and -4° results in increase of peak deceleration by 51% and peak heat flux by 47%. In Fig. 7c, heat flux is compared wrt altitude and time. As γ (-4° to -1°) decreases, duration of flight also increases, which has increased total heat load for smaller entry flight path angles. It is seen that total heat load for trajectories with $\gamma = -4^\circ$, $\gamma = -2^\circ$, and $\gamma = -1^\circ$ is 4362.5 J/cm^2 , 5571.6 J/cm^2 , and 6276.7 J/cm^2 , respectively. Thermal response analysis has been carried out considering insulation and structure thickness as 0.025 m and 0.002 m, respectively, and backwall temperatures of structure as

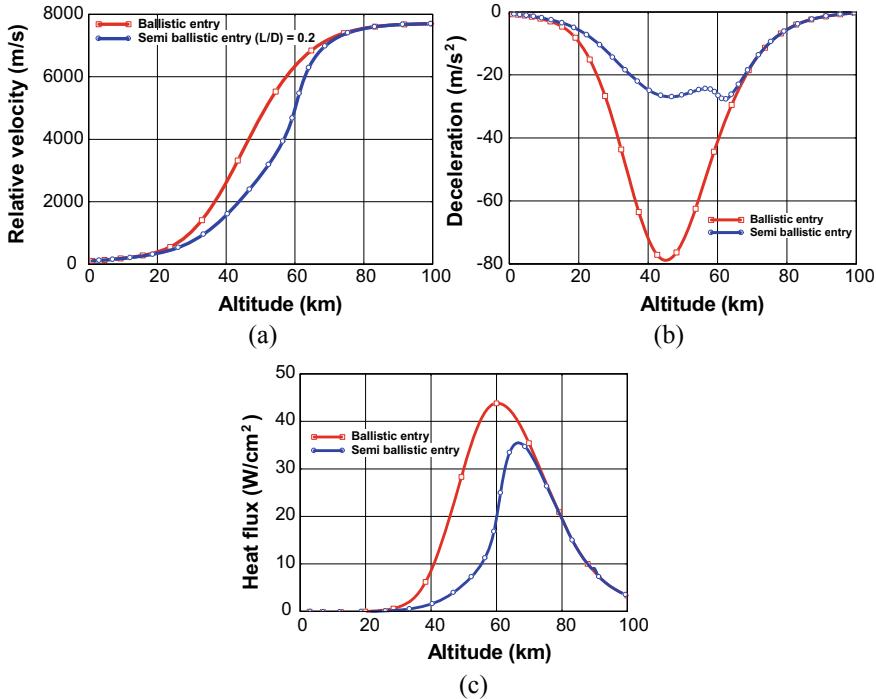


Fig. 6 Comparison of ballistic/semi-ballistic reentry trajectory parameters and heating histories

shown in Fig. 7d. For the same insulation thickness, maximum temperatures attained for smaller entry flight path angle is higher due to its higher heat load.

2. Variation in initial mass:

Semi-ballistic trajectories with ballistic coefficient with 277.8 kg/m^2 ($m = 2000 \text{ kg}$) and 1111.1 kg/m^2 ($m = 8000 \text{ kg}$), $L/D = 0.2$, $\gamma_E = -2^\circ$ are generated. Objects of heavier bodies correspond to steeper trajectories resulting higher deceleration and heating rates shown in Fig. 8a–c. Increase in mass from 2000 to 8000 kg results in increase of peak deceleration by 5.1% and peak heat flux by 60%.

3.2 Optimal Insulation Thickness by MDA of Trajectory and Heating

Total mass of the reentry vehicle consists of mass of payload, structure, and insulation.

$$m = m_{PL} + m_S + m_{TPS} \quad (11)$$

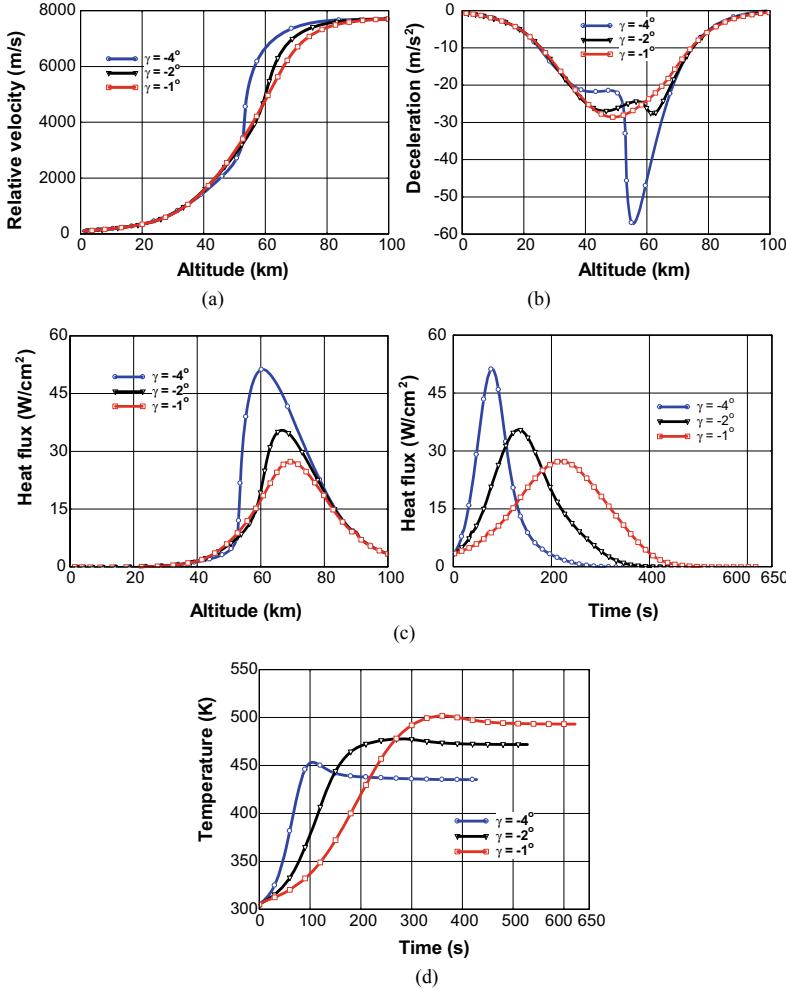


Fig. 7 **a** Semi-ballistic reentry trajectories **b** Deceleration **c** heat flux (altitude vs heat flux and time vs heat flux) **d** backwall temperature histories ($\gamma_E = -1^\circ$, -2° and -4°)

$$m_{\text{TPS}} = m_{\text{TPS}}(\Delta) \quad (12)$$

By considering mass of payload and structure to be constant, trajectory, aerodynamic heating, and thermal response analysis was carried out to study the influence of insulation thickness on the total mass of the reentry vehicle. Insulation thickness depends on aerodynamic heating experienced during the reentry. The aerodynamic heating depends on reentry trajectory:

$$q(t) = q(r(t), V(t), R_N) \quad (13)$$

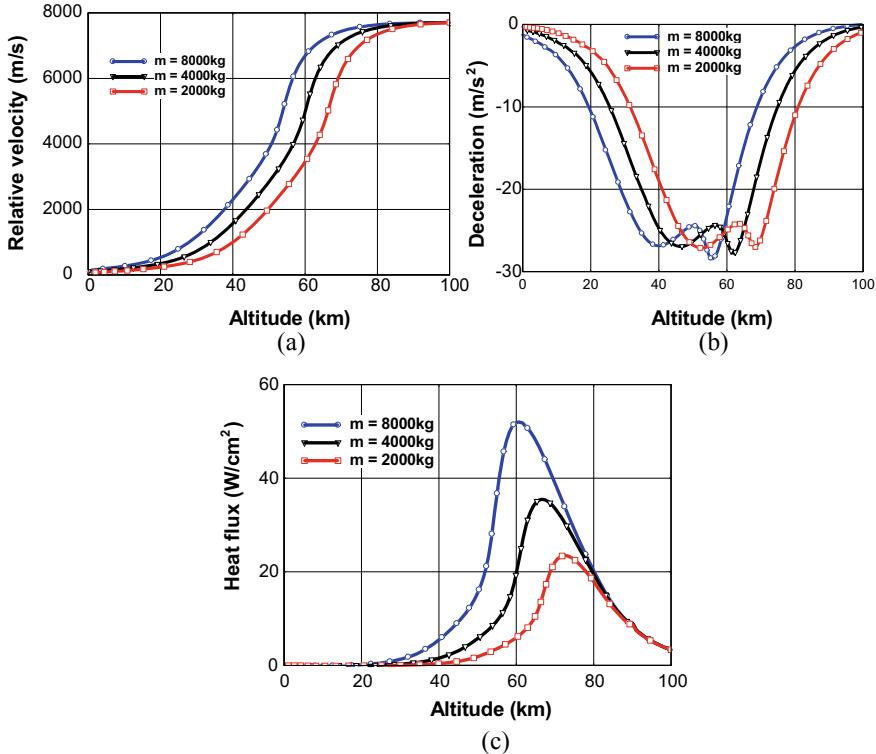


Fig. 8 Trajectory and heat flux histories for heavier and lighter geometries

The trajectory parameters depend on the initial state vector at the instant of reentry, L/D ratio, bank angle, mass, and the drag characteristics of the vehicle as discussed in Sect. 2.

Insulation thickness is computed based on stagnation heating ($R_N = 2.0$ m). For this analysis, uniform insulation thickness on the reentry vehicle is considered. Analysis is carried for a reentry vehicle with initial ballistic coefficient (m/C_{DA}) of 555.56 kg/m^2 ($m = 4000 \text{ kg}$), $L/D = 0.2$, $\gamma_E = -2^\circ$. Graphite is considered as insulation material to restrict aluminum alloy structure temperature to 473 K.

Based on DSM of multi-disciplinary analysis (MDA) as shown in Fig. 1, influence of insulation mass on trajectory and heating rates is studied to obtain optimal insulation thickness and restrict structure temperature to 473 K. Secant method is used for finding the optimal insulation thickness. Figure 9 shows the optimal achieved insulation thickness using Secant method. Optimal insulation thickness is achieved in five iterations. Figure 10a shows reentry trajectories for three masses. It can be seen that maximum change in the trajectory occurs ~ 60 km altitude. From Fig. 10b, it is seen that there is 9% increase in peak heating rates for the converged insulation mass from the initial guess.

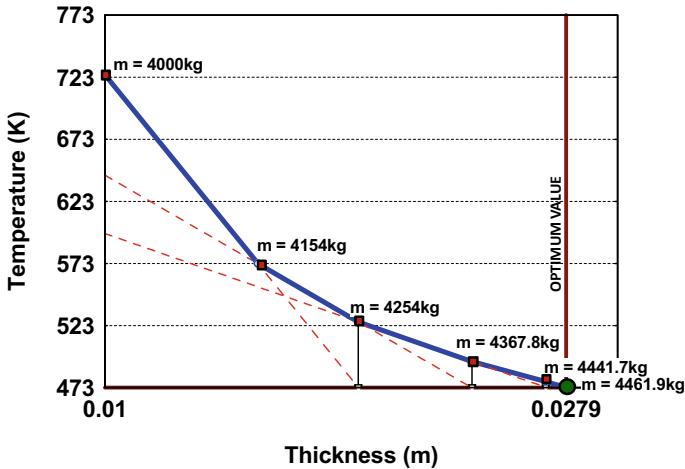


Fig. 9 Optimal computed insulation thickness after MDA of trajectory and heating

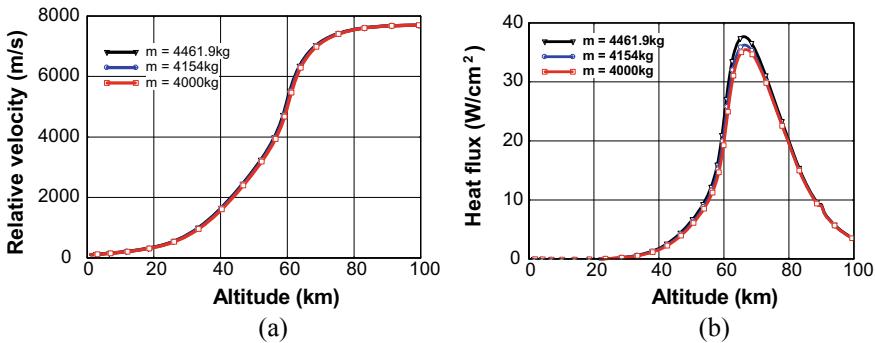


Fig. 10 Effect of insulation mass on trajectory and heating rates

4 Conclusions

- Software package has been developed to carry out multi-disciplinary analysis (MDA) to study the aerothermal performance of a typical reentry vehicle.
- Multi-disciplinary analysis of trajectory and aerodynamic heating is carried out to converge the insulation mass for a given a reentry vehicle configuration.
- Using MDA tool, effect of ballistic and lifting reentries is seen on aerodynamic heating and transient thermal response analysis.
- This analysis shows that the change in geometrical parameters and input parameters significantly influence deceleration and heating rates.

References

1. J. John C. Adams, Atmospheric reentry (June 2003)
2. S. James, Normal and oblique flow parameters in equilibrium air including attached shock solutions for surfaces at angles of attack, sweep and dihedral. (NASA SP 3093, 1975)
3. S. Srinivasan, J.C. Tannehill, Simplified curve fits for the thermodynamic properties of equilibrium air. (NASA Ref Publication, no. 1181, Aug 1987)
4. F.A. Fay, F.R. Riddell, Theory of stagnation point heat transfer in dissociated air. *J. Aeronaut. Sci.* **25**(2), 73–85 (1958)
5. E.R. Eckerts, Survey on heat transfer at high speeds, ARL 189, vol. 26 (US Airforce, Dec 1961)
6. Y. A. Cengel, Heat transfer, a practical approach, vol. 2nd edition (2003)
7. P.L. Kirillov, Thermo physical properties of materials for nuclear engineering. Institute for heat and mass transfer in nuclear power plants
8. ANSYS User manual
9. A. Viviani, G. Pezzella, *Aerodynamic and Aerothermodynamic Analysis of Space Mission Vehicles* (Springer Aerospace Technology, Springer, Berlin, 2015)
10. V. Carandente, R. Savino, M. Iacovazzo, C. Boffa, Aerothermal analysis of a sample-return reentry capsule. *FDMP* **9**(4), 461–481 (2013)

Optimization of Multiple Gravity Assist Trajectories with Deep Space Maneuver Using Evolutionary Algorithms



H. R. Sahana, Pooja Dutt, and A. K. Anilkumar

Abstract Multiple gravity assist (MGA) trajectories represent a particular class of space trajectories in which a spacecraft makes use of gravity assist (GA) of one or more celestial bodies to alter its path or velocity vector, in order to reach high ΔV targets with low propellant consumption. The search for optimal transfer trajectories can be formulated as a global optimization problem. A simple MGA problem without any deep space maneuver (DSM) considers the departure epoch and the transfer times of the trajectories between two planets as the design variables for the objective function evaluation with constraint on minimum periapsis radius at each planet. The introduction of DSM in this problem during a trajectory leg makes the model more flexible, but also more complex. Apart from the design variables taken for MGA problem, the bounds on additional variables relating to spacecraft's relative velocity at departure planet, the time instant at which each DSM takes place, the pericenter radius at each body and the turning angle of each hyperbola are considered for the objective function assessment. This paper evaluates some benchmark MGA mission problems with one DSM. The data of these problems are available under Global Trajectory Optimization Competition (GTOC) in European Space Agency (ESA) website. These problems are optimized using the evolutionary algorithms (EAs) like differential evolution (DE), genetic algorithm (GA), particle swarm optimization (PSO), and a comparison of the results are made.

Keywords Multiple gravity assist · Deep space maneuver · Evolutionary algorithms · Interplanetary missions

H. R. Sahana (✉)

Aerospace Engineering, Manipal Institute of Technology, MAHE, Manipal, India
e-mail: sahanahr98@gmail.com

P. Dutt

Applied Mathematics Division, Vikram Sarabhai Space Centre, ISRO, Thiruvananthapuram, India
e-mail: pooja_dutt@vssc.gov.in; pdisro@gmail.com

A. K. Anilkumar

Director, Directorate of Space Surveillance and Directorate of Space Situational Awareness and Management (DSSAM), ISRO - HQ, Bengaluru, India
e-mail: ak_anilkumar@isro.gov.in

1 Introduction

MGA trajectory model [1, 2] is one particular class under the vast category of space trajectory design problems where the spacecraft is assumed to thrust only during its planetocentric phases (powered flyby). The trajectory in between the planets is evaluated using Lambert's problem solver [3] (intermediate impulses/DSMs are neglected). This model's design variables include only departure epoch and times of flight between the two planets along trajectory and the constraints on the pericenter radius at each gravity assist body. This model is simpler when compared to the MGA with DSM model. Also, this may not help in finding the best trajectory model since no maneuvers are allowed in between the planets. The actual trajectory design would benefit from these impulses, also called the deep space maneuvering (DSM).

A DSM is a change in the velocity of the spacecraft during a leg/trajecotry between the planets [1, 4, 5]. This paper deals with the computation of the ΔV cost function of a MGA trajectory with one DSM allowed between each planet pair. All the flybys are taken unpowered. This MGA-1DSM model is more flexible but also more complex. Along with the variables used for MGA model, this model requires the introduction of new variables to consider the intermediate impulses. Hence, the search space is complex and highly oscillatory. The optimization of this problem becomes tedious due to the presence of many sub-optimal minima, and the objective function is extremely sensitive to the design variables. It is often difficult to compare the results, as the optimization methods are problem-dependent. A benchmark problem on MGA without DSM model is dealt with [6]. As a continuation, a comparison of the evolutionary algorithms (differential evolution, genetic algorithm, particle swarm optimization) [7, 8] considering the benchmark mission problems like Cassini–Huygens mission, Messenger mission and Rosetta mission [9–12] is studied in this paper. The data related to these missions are available in the European Space Agency (ESA) Web site under the Global Trajectory Optimization Competition (GTOC) database [12].

2 Methodology

The schematic representation of the MGA-1DSM trajectory model is as shown in Fig. 1.

A sequence of $N = n + 1$ astronomical bodies (N is length of planetary sequence) is considered P_0, \dots, P_n where P_0 is the departure planet and is usually the Earth and P_n is the destination celestial body. The planetary bodies considered need not be distinct. The sequence is to be visited by spacecraft in such a way that it is likely to minimize the overall energy consumption ΔV . It is assumed that only unpowered flybys are possible [9–11]. Some of the assumptions made for the design of MGA-1DSM trajectory are as follows [4, 5]:

- Patched conics is applied, and the perturbations are neglected.
- The planetary sequence of the mission is fixed.

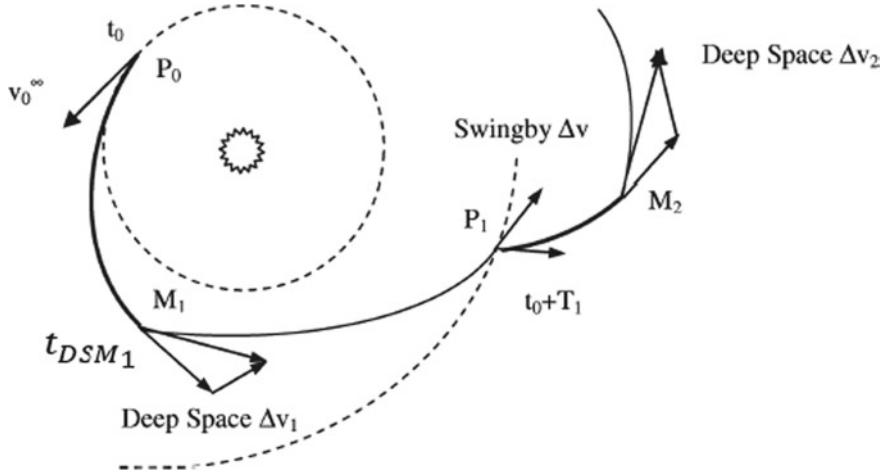


Fig. 1 Schematic representation of the MGA-1DSM trajectory model [1]

- Multiple revolutions around a planet are not considered.
- Maximum of one intermediate impulse/DSM is considered in each leg between the planets.
- Transfer directions are all counterclockwise (prograde orbits).

The magnitude of the forces of perturbations is much smaller when compared to the main gravitational force of the planets. Perturbations like atmospheric drag, solar radiation pressure, gravitational influence from bodies other than the main attracting body are neglected in the preliminary analysis. During the planetary encounters (gravity assists), only the gravitational influence of planets is considered. Modeling of the trajectory of the mission is done in the heliocentric reference frame and modeling of the gravity assists around the planet is done in the appropriate planetocentric reference frame at that instant of encounter.

The design variables considered to design the MGA-1DSM trajectory model are as follows [4, 5]:

- t_0 : The departure epoch from the Earth (P_0) in modified Julian date 2000 (MJD2000) format.
- $T_i, i = 1, \dots, n$: The times of flight (in days) of the interplanetary legs joining the body P_{i-1} to P_i .
- V_∞ : Magnitude of the relative velocity of the spacecraft at the departure planet P_0 .
- u, v : The directions (in-plane and out-of-plane angles) of the relative velocity of the spacecraft at the departure planet P_0 .
- $\eta_i, i = 1, \dots, n$: The fraction of the time of flight of the segment that has passed until the application of DSM.
- $r_{p_i}, i = 1, \dots, n - 1$: The pericenter radius at the gravity assists of the planet.
- $b_{\text{incl}_i}, i = 1, \dots, n - 1$: The rotation inclination angle in 3D (plane change angle) at the gravity assist planet.

The time instant (in MJD2000) during which the spacecraft visits the planet can be given by Eq. 1 [4].

$$t_i = t_0 + \sum_{j=1}^{i-1} T_j \quad (1)$$

Using Eq. 1 and planetary ephemerides, the position of the planets can be obtained. Analytical ephemerides are used instead of the real ones. Next the heliocentric velocities after departing from the Earth are calculated. After determining the heliocentric velocity of spacecraft after departing from the departure planet, the Kepler propagation (Theorem 1) is used to determine the trajectory leg up to the instant where the DSM has to be applied. The time instant where DSM is applied can be obtained using Eq. 2 [4].

$$t_{\text{DSM}_i} = t_0 + \sum_{j=1}^{i-1} T_j + \eta_i T_i \quad (2)$$

After obtaining the time instant of the intermediate impulse, the heliocentric velocity after application of the DSM and also the heliocentric velocity at the arrival to the next planet are calculated using Lambert's problem solver (Theorem 2). The difference between heliocentric velocity before the instant of the DSM application and the heliocentric velocity after the application of DSM gives the velocity change, ΔV required for the DSM. The final objective function can be evaluated as Eq. 3 [4].

$$\Delta V_{\text{total}} = |\Delta V_{\text{dep}}| + |\Delta V_{\text{DSM}_i}| + |\Delta V_{\text{arr}}| \quad (3)$$

where $|\Delta V_{\text{dep}}|$ is the magnitude of the velocity change required at the departure planet, $|\Delta V_{\text{DSM}_i}|$ is the magnitude of the velocity change required at the location where DSM is to be applied and $|\Delta V_{\text{arr}}|$ is the magnitude of the velocity change required to insert the spacecraft into the orbit of the destination body. Unlike the simple MGA model, here the pericenter radius is taken as the decision variable, not as the constraint separately. Finally, the evolutionary optimization technique is used to find the best MGA-1DSM trajectory among all the trajectories by minimizing the objective function given in Eq. 3, along with the bounds on design variables.

Theorem 1 (Kepler Propagation) *Propagation refers to finding the state of the spacecraft at a given time, by solving the equations of motions, given the initial states (position and velocity) of the spacecraft at some earlier initial time. To evaluate a basic problem of orbit propagation, a fast and analytic technique called Kepler propagation is used, which can be stated as follows: “given r_i , V_i and time t_i , calculate r_f , V_f at time t_f .”*

Theorem 2 (Lambert's Problem) *Lambert's theorem states that “transfer time Δt from position P_1 to position P_2 is independent of orbit's eccentricity e and depends only on the sum of the position vectors ($r_1 + r_2$), semimajor axis a and length of the*

chord joining P_1 and P_2 , c , that is, $\Delta t = f(r_1 + r_2, a, c)$.” If the time of flight Δt between P_1 to P_2 is known, the problem of finding the trajectory joining P_1 and P_2 is Lambert’s problem.

2.1 ΔV Computation

Departure: After the departure of the spacecraft from the departing planet (Earth), the heliocentric outgoing velocity of the spacecraft (V_{SC}^0) is determined by summing the velocity of the departure planet V_P to the relative velocity V_0 . The position (r_P) and velocity (V_P) of the departing planet are already calculated using planetary ephemerides at the launch time t_0 . After this calculation, Kepler propagation is used to continue the trajectory till the moment of the DSM application. The initial position of the spacecraft r_{SC}^0 is nothing but the position of the departing planet r_P . The spacecraft heliocentric velocity is also calculated. Using these initial conditions at the time instant of application of DSM t_{DSM_1} , which is calculated using Eq. 2, the heliocentric arc can be calculated from P_0 till the application of the first impulse using Kepler propagation. By this calculation, the final state of the spacecraft at time instant t_{DSM_1} , namely position of spacecraft at the time of application of DSM, r_{DSM_1} and heliocentric incoming velocity of spacecraft before the application of DSM impulse, $V_{DSM_{1,in}}$ are obtained.

The second arc connects the point r_{DSM_1} at t_{DSM_1} to the position of planet P_1 at time t_1 , that is after a time of flight of $T_1(1 - \eta_1)$ by solving Lambert’s problem. The solution of Lambert’s problem gives the velocity vectors at its boundaries, that is, $V_{DSM_{1,out}}$ and $V_{SC_{2,in}}$. Here $V_{DSM_{1,out}}$ is the heliocentric outgoing velocity of spacecraft after the application of DSM impulse and $V_{SC_{2,in}}$ is the spacecraft heliocentric incoming velocity at the first flyby planet P_1 . Now the first contribution of the ΔV can be easily computed using Eq. 4 [1, 4, 5]. Note that the magnitude of outgoing hyperbolic excess velocity V_∞ at the departure planet is also considered in this equation.

$$\Delta V_{dep} = \|V_{DSM_{1,out}} - V_{DSM_{1,in}}\| + V_\infty \quad (4)$$

Gravity Assist: The unpowered flyby model [1, 4] is used to match two consecutive legs of the hyperbola at the flyby planet. We consider the spacecraft to have same position of flyby planet $P_{i=1,\dots,n-1}$. The positions and velocities of the flyby planets are calculated using the planetary ephemerides. The hyperbolic excess velocity $V_{\infty_{in}}$ in the planetocentric reference frame during the gravity assist of that planet, is calculated by “subtracting the planet’s heliocentric velocity vector V_P from the spacecraft’s heliocentric velocity vector $V_{SC_{in}}$ at the end of the incoming Lambert arc.” The magnitude of this $V_{\infty_{in}}$ is not changed during the gravity assist, but there is a change in its orientation. The pericenter radius r_p and the plane change angle b_{incl} are used to define the flyby geometry. Although the unpowered flyby does not change the magnitude of the relative velocity vector (planetocentric), there might be a significant difference in absolute velocity (heliocentric) [5]. Subsequently, the

outgoing hyperbolic excess velocity in planetocentric reference frame, $V_{\infty_{\text{out}}}$ is calculated. Finally by summing the velocities $V_{\infty_{\text{out}}}$ and V_P , the outgoing heliocentric velocity $V_{SC_{\text{out}}}$ is calculated.

Then the spacecraft trajectory leg till the moment of application of DSM is calculated using Kepler propagation and from there till the next flyby planet is calculated using Lambert's problem solver. The procedure is similar to the one followed in the above "Departure" section. The ΔV which is required for the DSM is given by Eq. 5 [1, 4, 5].

$$\Delta V_{\text{DSM}_{i=2,\dots,n-1}} = \|V_{\text{DSM}_{i,\text{out}}} - V_{\text{DSM}_{i,\text{in}}}\| \quad (5)$$

Arrival: When the spacecraft approaches its target body, a closure burn ΔV is given so as to insert the spacecraft to the target body's orbit. In the benchmark missions, namely Cassini–Huygens, Messenger, and Rosetta missions considered in this MGA-1DSM model, a rendezvous problem to the target body is considered. After obtaining the spacecraft heliocentric incoming velocity $V_{SC_{n,\text{in}}}$ to the target body P_n , the planetocentric hyperbolic excess velocity $V_{\infty_{n,\text{in}}}$ is calculated by "subtracting the planet's heliocentric velocity vector V_P from $V_{SC_{n,\text{in}}}$." This itself is taken as the final contribution to the ΔV which is given by Eq. 6 [1, 4, 5].

$$\Delta V_{\text{arr}} = V_{\infty_{n,\text{in}}} \quad (6)$$

Thus adding all the ΔV 's, i.e., ΔV_{dep} , $\Delta V_{\text{DSM}_{i=1 \text{ to } (n-1)}}$ and ΔV_{arr} the final objective function ΔV_{total} is obtained. This is given by Eq. 7.

$$\Delta V_{\text{total}} = \|\Delta V_{\text{dep}}\| + \|\Delta V_{\text{DSM}_i}\| + \|\Delta V_{\text{arr}}\| \quad (7)$$

This objective function is used by the optimizers to find the optimal trajectory design with minimum ΔV_{total} budget. The dynamics involved in designing and optimizing the multiple gravity assist trajectory with deep space maneuver to obtain the optimized final objective function is given by the flowchart in Fig. 2.

3 Evolutionary Algorithms

The conventional methods like steepest descent method, Newton's method, Conjugate gradient method, Nelder–Mead method etc. [7] requires the objective function to be continuous and differentiable and also the solution depends on the chosen initial conditions. These methods are not so efficient in solving the discrete variables and are more likely to settle in local minima for multimodal functions.

The evolutionary algorithms like differential evolution (DE), genetic algorithm (GA), particle swarm optimization (PSO) etc. [7, 8] are some of the global opti-

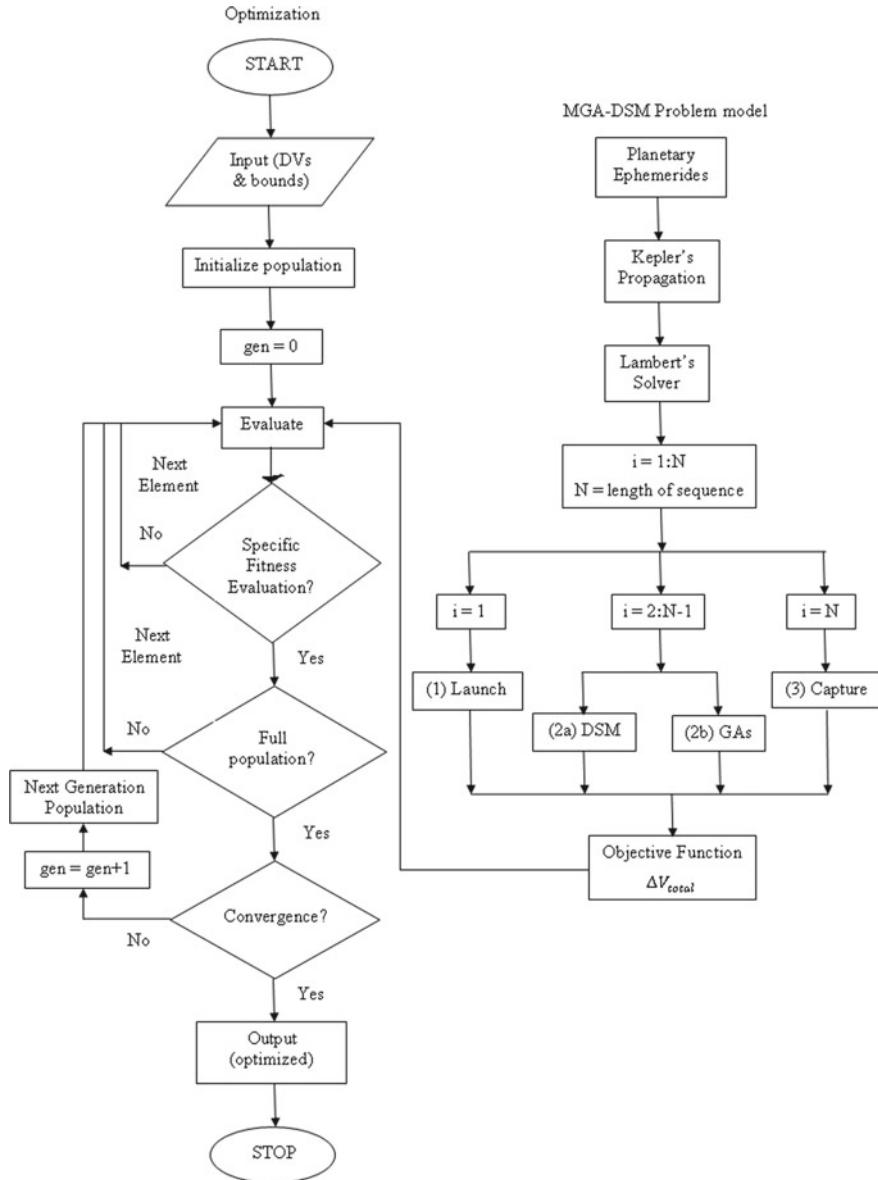


Fig. 2 Flowchart for the optimization of the “multiple gravity assist with deep space maneuver” trajectory model [5]

mization techniques that do not require the objective function to be continuous or differentiable. These are based on the nature's biological process, where, instead of using a single search point, an initial population of candidate solutions is generated; the fitness of each individual is evaluated in that population using selection, crossover, and mutation; and the next generation is continued until termination criteria are met. In this way, the chance of obtaining global optimum is higher.

4 Benchmark Missions

The benchmark mission problems considered for MGA-1DSM model here are Cassini–Huygens, Messenger, and Rosetta missions for which the data is taken from the GTOC database of ESA website [9–12]. The details of the actual missions launched which are considered as benchmark mission problems for the evaluation of MGA-1DSM design are as follows.

- **Cassini–Huygens** was launched on October 15th, 1997, and it reached Saturn's orbit on July 1st, 2004, with the help of flybys of Earth, Venus, and Jupiter. Here, Cassini–Huygens is taken as an MGA-1DSM problem.
- **Messenger** was launched on August 3rd, 2004, and it reached Mercury's orbit on March 18th, 2011, the first encounter with Mercury dated 14th January 2008, with the help of flybys of Earth, Venus, and resonant flybys of Mercury. Here Messenger is taken as an MGA-1DSM problem in its reduced version (by not considering resonant flybys around Mercury).
- **Rosetta** was launched on March 2nd, 2004, and it reached the comet 67P Churyumov–Gerasimenko (67P) on May 7th, 2014, with the help of flybys of Earth and Mars. Here Rosetta is considered as an MGA-1DSM problem.

The detailed description and data of these missions are explained further. A fair comparison of the solvers used is made based upon the results obtained. The parameters set for different optimization solvers are as follows:

- **DE**: population size: 50, scaling factor F : $[lb=0.1, ub=0.95]$, crossover function: binomial, crossover rate CR: 0.9, max. no. of iterations: 5000, strategy: DE/rand/1/bin.
- **GA**: population size: 50, function tolerance: 10^{-10} , selection function: roulette, crossover rate: 0.75, crossover function: scattered, mutation rate: 0.01, mutation function: adaptive feasible, max. no. of iterations: 5000.
- **PSO**: population size: 50, inertial weight ω : 0.65, inertial weight damping ratio: 0.99, $\eta_1 = \eta_2 = 2$, max. no. of iterations: 5000.

Changing these parameters would also give different results.

5 Results

5.1 Problem Definition and Results for Cassini–Huygens Mission Model

The objective of this mission's model is to reach the closer proximity of Saturn and rendezvous it. The planetary swing-by sequence considered is “Earth-Venus-Venus-Earth-Jupiter-Saturn.” The number of design variables considered is 22. The bounds and details of these variables are taken from [9, 10, 12].

The results obtained by applying the evolutionary algorithms DE, PSO, and GA for the benchmark mission Cassini–Huygens are given in Table 1. The reported best solution known for this problem is $x = [-779.0467, 3.2591, 0.5259, 0.3808, 167.3789, 424.0282, 53.2897, 589.7669, 2200, 0.7694, 0.5132, 0.0274, 0.2639, 0.5999, 1.3487, 1.05, 1.3073, 69.8090, -1.5937, -1.9595, -1.5549, -1.5134]$ [9, 10, 12] which gives the objective function value of 8.3830 km/s. The trajectory plot and the radial distance plot for the reported best solution in [9, 10, 12] for the Cassini–Huygens problem are given by Figs. 3 and 4, respectively.

Table 1 Performance of all the solvers (over 50 independent runs) for Cassini–Huygens problem

	Mean	Maximum	Median	Standard deviation	Best solution
DE	12.1818	15.2901	12.8579	2.4123	8.6317
PSO	21.8129	24.2906	22.3189	2.0818	15.4168
GA	20.6804	24.9361	21.1872	3.3549	10.3262

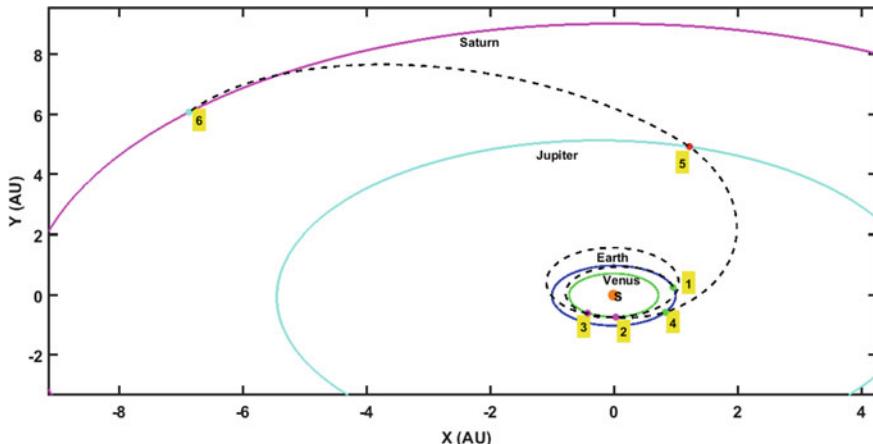


Fig. 3 2D trajectory plot for the reported best solution of the Cassini–Huygens problem

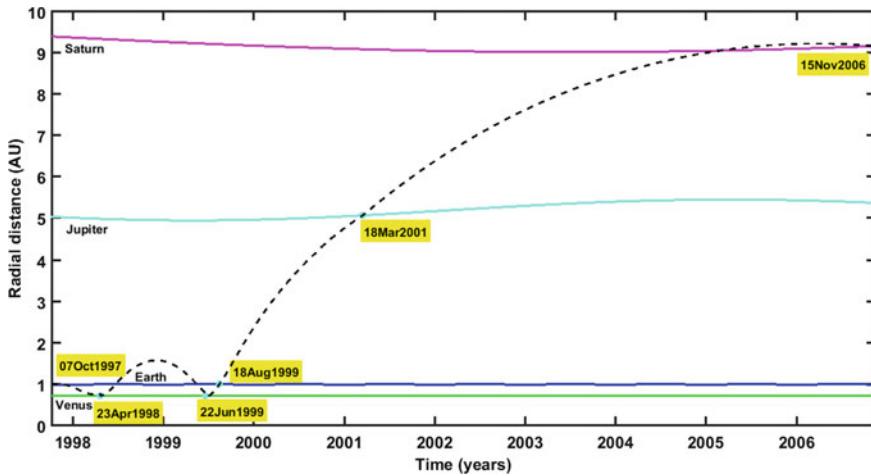


Fig. 4 Radial distance plot for the reported best solution of the Cassini–Huygens problem

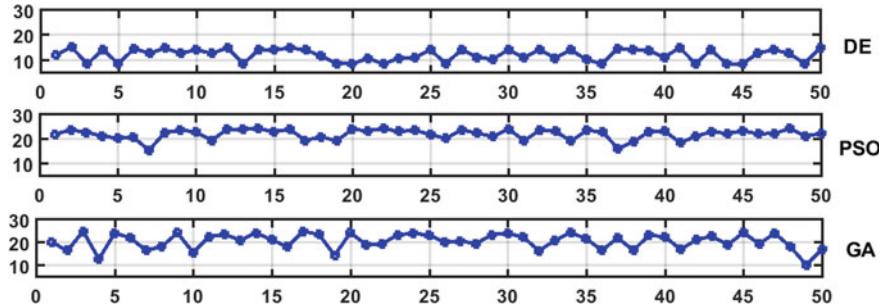


Fig. 5 Performance of DE, PSO, and GA for 50 independent runs for Cassini–Huygens problem

The best solution obtained using DE is $x = 1.0e+03 * [-0.8010, 0.0031, 0.0006, 0.0004, 0.1908, 0.4225, 0.0533, 0.5895, 2.1974, 0.0000, 0.0005, 0.0000, 0.0002, 0.0000, 0.0013, 0.0011, 0.0013, 0.0699, -0.0016, -0.0020, -0.0016, -0.0015]$ which gives the objective function value of 8.6317 km/s.

Statistics is generated for each optimization algorithm for 50 independent runs. Figure 5 gives the variation in minimum value obtained by each optimizer, i.e., DE, GA, PSO for the Cassini–Huygens problem.

The trajectory plot and the radial distance plot for the best result obtained using DE as solver for the Cassini–Huygens problem are given by Figs. 6 and 7, respectively. Here the DE algorithm gives the best minima value. Even in average performance, DE outruns the performance of PSO and GA. For this model, the performance of GA is better than that of PSO.

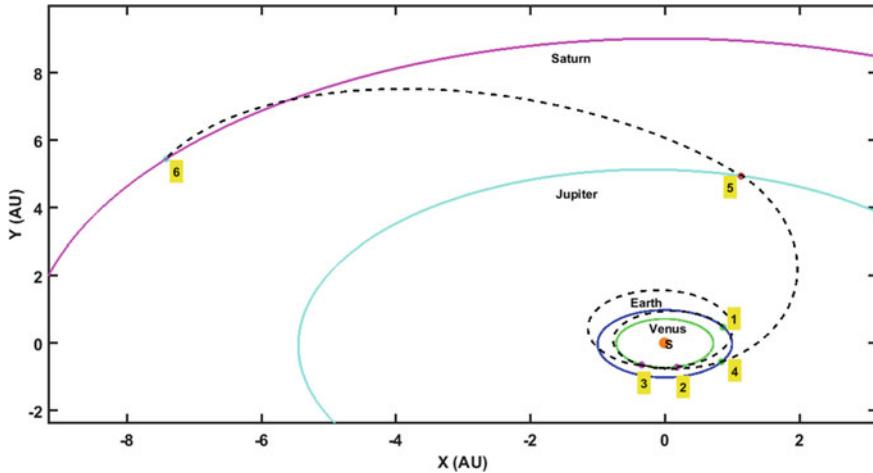


Fig. 6 2D trajectory plot for the obtained best solution of the Cassini–Huygens problem

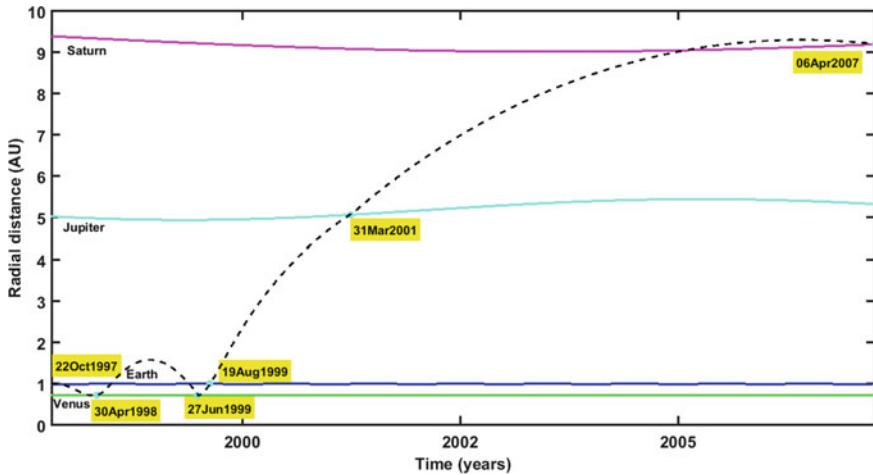


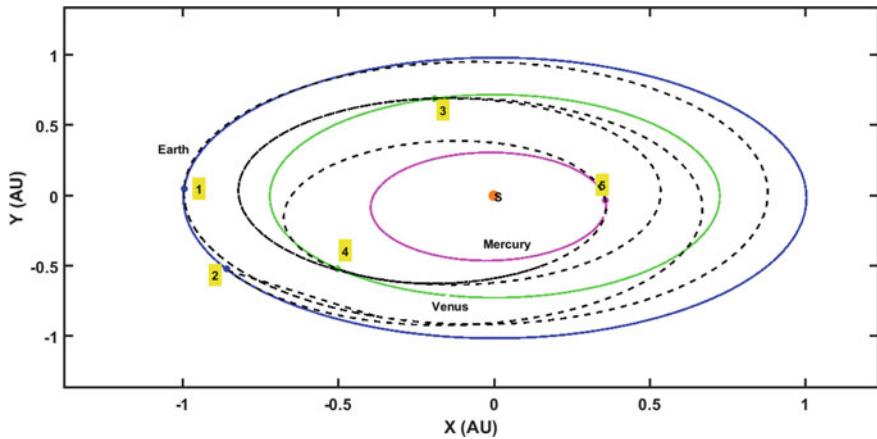
Fig. 7 Radial distance plot for the obtained best solution of the Cassini–Huygens problem

5.2 Problem Definition and Results for Messenger Mission Model

The objective of this mission's model is to represent a rendezvous mission to Mercury where the resonant flybys in the Mercury orbit are not included. The planetary swing-by sequence considered is “Earth-Earth-Venus-Venus-Mercury.” The number of design variables considered is 18. The bounds and details of these variables are taken from [9, 10, 12].

Table 2 Performance of all the solvers (over 50 independent runs) for Messenger problem

	Mean	Maximum	Median	Standard deviation	Best solution
DE	12.0662	13.4924	12.4766	1.0429	9.9972
PSO	14.0123	21.7870	13.8085	2.3862	9.2296
GA	15.0388	19.8879	15.0388	1.9513	11.0305

**Fig. 8** 2D trajectory plot for the reported best solution of the Messenger problem

The results obtained by applying the evolutionary algorithms DE, PSO, and GA for the benchmark mission Messenger are given in Table 2. The reported best solution known for this problem is $x = [1171.6450, 1.40899, 0.3799, 0.4980, 399.9999, 178.3722, 299.2231, 180.5107, 0.2345, 0.09647, 0.8299, 0.3171, 1.8062, 3.0412, 1.1000, 1.3507, 1.0955, 1.34317]$ [9, 10, 12] which gives the objective function value of 8.6310 km/s. The trajectory plot and the radial distance plot for the reported best solution in [9, 10, 12] for the Messenger problem are given by Figs. 8 and 9, respectively.

The best solution obtained using PSO is $x = 1.0e+03 * [1.2489, 0.0015, 0.0007, 0.0005, 0.3164, 0.1836, 0.3001, 0.1805, 0.0008, 0.0001, 0.0005, 0.0003, 0.0017, 0.0028, 0.0011, 0.0018, 0.0011, 0.0013]$ which gives the objective function value of 9.2296 km/s.

Statistics is generated for each optimization algorithm for 50 independent runs. Figure 10 gives the variation in minimum value obtained by each optimizer, i.e., DE, GA, PSO for the Messenger problem.

The trajectory plot and the radial distance plot for the best result obtained using DE as solver for the Messenger mission are given by Figs. 11 and 12, respectively. None of the algorithms were able to obtain these global optima because of the strong local optima at 9 km/s. Even though PSO gives the nearly best solution, in terms of

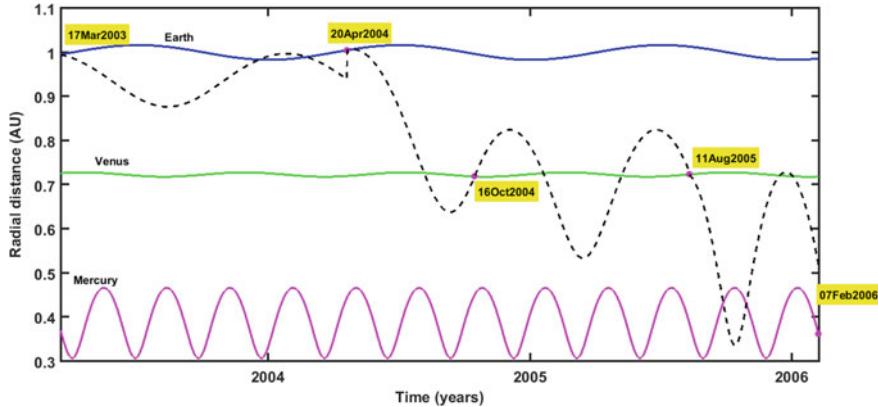


Fig. 9 Radial distance plot for the reported best solution of the Messenger problem

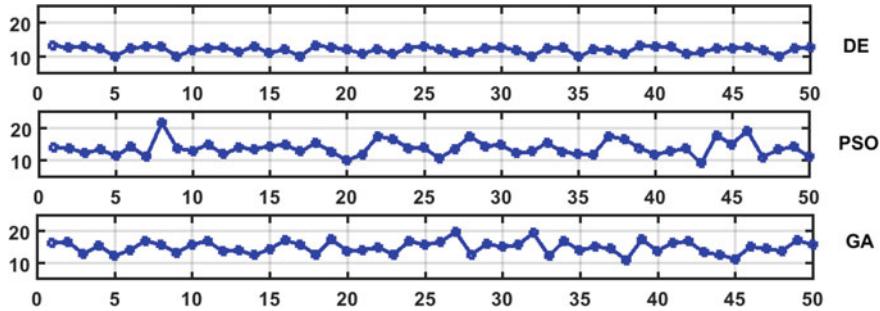


Fig. 10 Performance of DE, PSO, and GA for 50 independent runs for Messenger problem

average performance, DE again proves its effectiveness when compared to PSO and GA. But unlike the previous problem, PSO's average performance is better when compared with that of GA.

5.3 Problem Definition and Results for Rosetta Mission Model

This is an MGA-1DSM mission to the comet 67P/Churyumov–Gerasimenko (67P) where the objective function is the total mission ΔV including the launcher capabilities. The planetary swing-by sequence considered is “Earth-Earth-Mars-Earth-Earth-67P/Churyumov-Gerasimenko.” The number of design variables considered is 22. The bounds and details of these variables are taken from [9, 10, 12].

The results obtained by applying the evolutionary algorithms DE, PSO, and GA for the benchmark mission Rosetta are given in Table 3. The reported best solution known

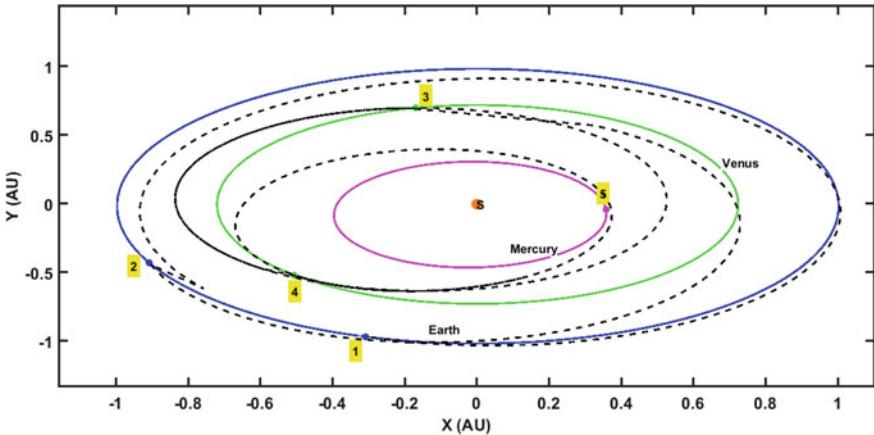


Fig. 11 2D trajectory plot for the obtained best solution of the Messenger problem

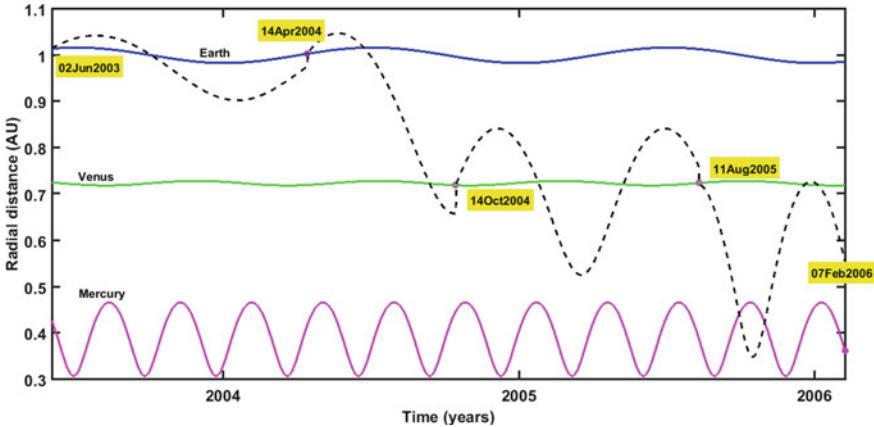


Fig. 12 Radial distance plot for the obtained best solution of the Messenger problem

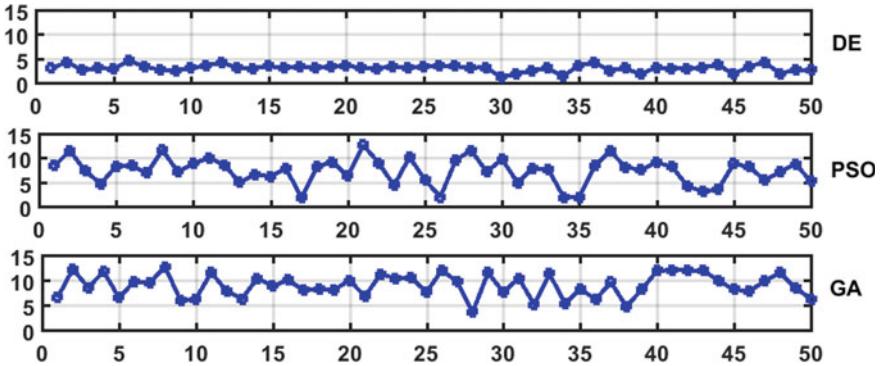
for this problem is $x = [1542.8027, 4.4784, 0.7316, 0.8782, 365.2423, 707.7546, 257.3238, 730.4837, 1850, 0.4691, 0.8103, 0.0572, 0.1233, 0.4365, 2.6576, 1.053.1978, 1.0562, -1.2538, 1.7876, -1.5946, -1.9773]$ [9, 10, 12] which gives the objective function value of 1.343 km/sec.

The best solution obtained using DE is $x = 1.0e+03 * [1.2489, 0.0015, 0.0007, 0.0005, 0.3164, 0.1836, 0.3001, 0.1805, 0.0008, 0.0001, 0.0005, 0.0003, 0.0017, 0.0028, 0.0011, 0.0018, 0.0011, 0.0013]$ which gives the objective function value of 1.3843 km/s.

Statistics is generated for each optimization algorithm for 50 independent runs. Figure 13 gives the variation in minimum value obtained by each optimizer, i.e., DE, GA, PSO for the Rosetta problem. Even for this problem, DE is able to obtain the

Table 3 Performance of all the solvers (over 50 independent runs) for Rosetta problem

	Mean	Maximum	Median	Standard deviation	Best solution
DE	3.2418	4.7923	3.2267	0.6637	1.3843
PSO	7.6077	11.8671	8.1327	2.5990	1.9442
GA	9.2446	12.9045	9.8084	2.3027	3.7713

**Fig. 13** Performance of DE, PSO, and GA for 50 independent runs for Rosetta problem

global optima as given in Table 3. Even the average performance of DE is very well, which tells that DE is more suitable for these types of problems when compared with that of PSO and GA. The performance of PSO is not so well when compared to that of DE. However, the performance of PSO is much better when compared to that of GA for this problem.

The best solution obtained by DE is close to the reported best solution [12]. For this problem, the variation of scaling factor between [0.1, 0.95] and crossover rate 0.9 seems to have provided high variation in donor and higher perturbation of parent resulting in extensive exploration of the search space.

6 Conclusion

The effectiveness of some of the evolutionary algorithms, i.e., DE, PSO, GA is compared using the benchmark interplanetary missions proposed by ESA under GTOC [12]. For the missions Cassini–Huygens and Rosetta, the DE algorithm was able to converge at the global optima given by [9, 10, 12] but for Messenger mission, where multiple revolutions around Mercury orbit are not considered, the strong local optima at around 9 km/s imposed difficulty for the optimizers to find the global optima. So, the strategy of an algorithm which works well for one problem may not go well with other problems which imply that the performances of the optimization solvers

are problem-dependent. In some cases (here, e.g., Messenger), the straightforward application of these solvers seems to fail in providing the expected global optima. The strategies and parameters set for the solvers are selected by trial and error. Fine-tuning of these parameters may yield different results. But in comparison with the reported results in [9, 10, 12], the obtained results show clearly that DE is one of the best performing global optimization algorithms for the MGA-1DSM problems.

Further, the combination of different optimization solvers can be used to find better solutions to this type of complicated trajectory optimization problems. Also, powerful local optimization techniques like monotonic basin hopping can be studied and combined with these algorithms to get better results.

Acknowledgements The first author acknowledges the support provided by Vikram Sarabhai Space Centre (VSSC), Trivandrum, and Manipal Institute of Technology, MAHE, Manipal, in carrying out this research work. The authors are thankful to Shri. Abhay Kumar, GD, AFDG, and Dr. V. Ashok, DD, VSSC (AERO) for their support and guidance.

References

1. M. Ceriotti, Global optimization multiple gravity assist trajectories. Thesis, Department of Aerospace Engineering, University of Glasgow, 2010
2. M. Scotti, Global optimization of multiple gravity assist trajectories: development of STA interplanetary module v3.0. Thesis, Politecnico Di Milano, 2013
3. H.D. Curtis, *Orbital Mechanics for Engineering Students*, 3rd edn. (Butterworth-Heinemann, an imprint of Elsevier, 2014)
4. P. Musegaas, Optimization of space trajectories including multiple gravity assists and deep space maneuvers. Thesis Report, Delft University of Technology, 2012
5. S. Molenaar, Optimization of interplanetary trajectories with deep space maneuvers—model development and application to a Uranus orbiter mission. Thesis Report, Astrodynamics and Satellite Systems, Faculty of Aerospace Engineering, Delft University of Technology, Delft, Netherlands, Aug 2009
6. H.R. Sahana, P. Dutt, A.K. Anilkumar, Optimization of Cassini trajectory using evolutionary algorithms, in *Conference Proceedings, International Conference on Numerical Analysis, Computing and Applications in Science, Engineering and Technology*, Mohandas College of Engineering and Technology, Trivandrum, 17–20 Dec 2018
7. R.K. Arora, *Optimization Algorithms and Applications*, 1st edn. (CRC Press, Taylor and Francis Group, 2015)
8. V. Arunachalam, Optimization using differential evolution. Water Resource Research Report, Department for Civil and Environment Engineering, The University of Western Ontario, July 2008
9. T. Vinko, D. Izzo, C. Bombardelli, Benchmarking different global optimization techniques for preliminary space trajectory design, in *58th International Astronautical Congress*, Hyderabad, 24–28 Feb 2007
10. T. Vinko, D. Izzo, Global optimization heuristics and test problems for preliminary spacecraft trajectory design. ESA Advanced Concepts Team Technical Report, Sept 2008
11. D. Izzo, Global optimization and space pruning for spacecraft trajectory design, in *Spacecraft Trajectory Optimization: A Book*, chap. 7, ed. by B. Conway (2008), pp. 178–201
12. European Space Agency, Global Trajectory Optimization Competition (2005). www.esa.int/gsp/ACT/collab/competitions.html

Conceptual Design Optimization of High-Altitude Airship Having a Tri-Lobed Envelope



M. Manikandan and Rajkumar S. Pant

Abstract High-altitude airships are being touted as the ideal aerial platform for long endurance applications for the next generation of communication systems. Such airships are usually powered by an electric propulsion system using solar cells mounted on upper surface of the envelope. Most existing studies related to such airships assume an axi-symmetric envelope shape, which has large drag and curved solar panels, both of which result in large size of solar panels. A possible solution to reduce solar panel area of a high-altitude airship is to use a multi-lobed envelope configuration on which flattish solar panels are mounted, which increases their power generation efficiency. This paper describes a methodology for conceptual design and sizing of high-altitude airships with a tri-lobed envelope, while incorporating the effect of parameters from four disciplines, viz envelope geometry, aerodynamics, operating environment, and solar irradiance. As a demonstration of the usefulness of this methodology, it is coupled to a particle swarm optimizer to obtain solutions that correspond to the minimum area of solar panels needed, while meeting all operating requirements and constraints for deployment on a specific day of the year. Compared to a baseline tri-lobed envelope, it is seen that the optimal solutions require 0.98–2.6% lower solar array area, while operating on four specific days of the year. Sensitivity analysis of area of solar array to a key parameter, viz fineness ratio (FR), is also carried out in this study. It is seen that as FR is increased from 3 to 6, the area of solar array required for operating on one specific day can be reduced substantially, due to reduction in the envelope drag and hence volume.

Keywords Stratospheric airship · Tri-lobed envelope · Multidisciplinary Design Optimization (MDO) · Particle Swarm Optimization (PSO)

M. Manikandan (✉) · R. S. Pant
Indian Institute of Technology Bombay, Mumbai, India
e-mail: manikandan.m@aero.iitb.ac.in

R. S. Pant
e-mail: rkpant@aero.iitb.ac.in

1 Introduction

High-altitude airships (HAAs) have been in vogue since many years as an aerial platform for long endurance applications. To meet the requirement of long endurance, most of these airships are solar-powered, since they have large sized envelopes, on top of which solar arrays can be easily mounted. Such airships are also called stratospheric airships, since they are designed to be deployed in an altitude band of 15–25 km above mean sea level, to take advantage of the low magnitude of ambient winds prevailing there, and hence, reducing the propulsive power needed for station-keeping. It may be noted that the stratospheric airships are expected to generate enough power, not only to meet the propulsive power required for their station-keeping needs, but also the power required by the payload mounted on them [1].

In the recent years, there has been an upsurge in promoting the use of unconventional multi-lobed hybrid airships. This is mainly due to their efficient aerodynamic characteristics and better flight performance, as compared to the conventional airships having a single-lobed envelope [2–5]. The key advantage of multi-lobed airships is relatively the flatter upper surface of the hull, which results in improved efficiency of the solar arrays, due to lower curvature.

The methodology described in this paper estimates the area of array, volume of the envelope, and mass breakdown of a tri-lobed airship, designed to operate at a specified geometrical location on a specific day of the year. The envelope shape consists of three conventional single-lobed bodies with some lateral overlap between them and is parameterized in terms of six geometry-related variables. The solar panel is assumed to subtend a fixed angle of 30° with the centerline. The power generated by the solar array mounted above the airship is estimated as a function of the altitude of deployment and the solar irradiance impinging on it. The magnitude and direction of ambient wind as a function of location and altitude of deployment are estimated using HWM 14 horizontal wind model developed by U.S. Naval Research Laboratory [6].

The problem is posed in a multidisciplinary design and optimization (MDO) framework, in which the optimum shape of the envelope and the location and size of the solar array of lowest area is obtained, meeting all specified requirements. Constraints are imposed to ensure that the power produced is just equal to the power needed to operate the onboard payload and the airship, and the buoyant lift produced is just equal to the total system weight. To study the effect of day of operation and operating altitude, optimal envelope shapes and layout of solar array were obtained for deployment on four specific days (viz summer solstice, winter solstice, and equinoxes) by allowing the altitude of deployment within the band of 17–25 km.

2 Disciplines Involved

The methodology involves four disciplines including geometry, aerodynamics, environment, and energy subsystems and accounts for their mutual interactions in order to achieve enhanced design solutions from the design optimization process. This section provides the details of envelope shape generation, disciplines involved, and design variables.

2.1 Geometry Model

Each of the three lobes is assumed to be of the NPL low-drag profile suitable for airship envelopes generated by National Physics Laboratory [7]. The NPL shape consists of two ellipsoids of revolution with the major axis of the rear ellipsoid is $\sqrt{2}$ times the major axis of the front ellipsoid. The 3D envelope can be described by revolving the 2D shape by 360 about the x -axis. The profile equation of NPL shape is given by:

$$\begin{cases} y = \pm b\sqrt{1 - (x - a)^2/a^2} & \text{for } x \leq a \\ y = \pm b\sqrt{1 - (x - a)^2/2a^2} & \text{for } x > a \end{cases} \quad (1)$$

Assuming each lobe to be a double ellipsoid shape, the tri-lobed envelope can be considered as a combination of three single-lobed bodies parallel to each other with a lateral overlap between them, and with solar array on the top surface as shown in Fig. 1. Figure 2 shows the details of the solar panel mounted on the top of the envelope surface.

The primary geometric parameters that define the body of multi-lobed airship are ratio of length of the airship to the maximum diameter of the envelope (ϕ), i.e., fineness ratio (FR), distance of maximum section from the nose (X_m), radius of curvature at the nose (R_0), and radius of curvature at the tail (R_1), relative distance between the centers of central lobe and outer lobes in longitudinal direction (e), relative distance between the centers of central lobe and outer lobes in lateral direction (f), and relative distance between the centers of central lobe and outer lobes in

Fig. 1 Tri-lobed airship envelope

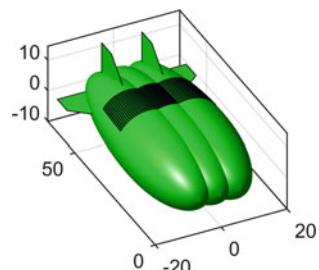
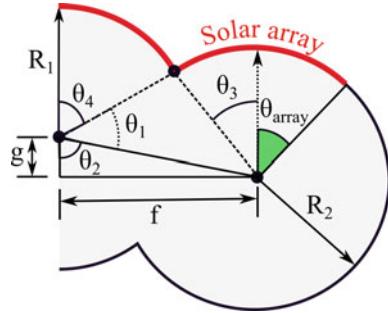


Fig. 2 Geometry of the solar array



vertical direction (g). The geometry and layout of solar array are defined by starting point of solar array (X_s), ending point of solar array (X_f), and the array angle (θ_{array}). The subtended angle of an array (θ_{array}) from the center of a lobe is fixed to be 30° . In the present study, the effect of ' e ' and ' g ' are ignored, and the value of ' f ' is assumed to be 0.3.

It is difficult to determine the exact planform and surface area of unconventional airship configurations; hence, they are usually approximated by an equivalent shape of a single ellipsoid, as suggested by Carichner and Nicolai [5]. In this study, an analytical method is used to calculate the exact planform and surface area.

The geometric model of the airship envelope is derived by the method described by Gertler [8], from a six-degree polynomial in dimensionless form, with $x = X/L$ and $y = Y/D$.

$$y^2(x) = \sum_{n=1}^6 a_n x^n \quad (2)$$

In the expansion form, Eq. (2) can be written as,

$$y^2(x) = a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5 + a_6 x^6 \quad (3)$$

The coefficients $a_1, a_2, a_3, \dots, a_6$ are determined in terms of X_m , R_0 , R_1 , and C_p . The Gertler coefficients for NPL shape are listed in Table 1.

Volume of a single lobe can be expressed as Eq. (4),

$$V = \pi D^2 L \frac{C_p}{4} \quad (4)$$

Table 1 Coefficients for baseline shape

Coefficients	a_1	a_2	a_3	a_4	a_5	a_6
Value	1.1768	-0.8689	-3.2643	6.8852	-5.5360	1.6072

The total net volume of the tri-lobed envelope can be calculated as Eq. (5)

$$V_{\text{net}} = 3V - 2V_{\text{int}} \quad (5)$$

where V_{net} = total net volume of envelope and V_{int} = intersection volume between lobes.

A total of nine design variables are used to define the geometry of the multi-lobed envelope with solar panels mounted atop it; the first six variables correspond to the envelope geometry, the next two define the geometry of the solar array, and the last one corresponding to the operating altitude. The design variables and the range of their values are given in Table 2.

Some key parameters that remain constant (called as ‘design constants’) are listed in Table 3, and their values were taken from Carichner and Nicolai [5], and Liang et al. [9].

Table 2 Design variables and the range of their values

Name of the variable	Symbol	Lower bound	Upper bound
Distance of max. diameter from the nose	X_m	0.3	0.6
Radius of curvature at nose	R_0	0	1
Radius of curvature at tail	R_1	0	1
Prismatic coefficient	C_p	0.55	0.7
Length of the envelope (m)	L	50	200
Fineness ratio	FR	3	6
Fraction of excursion of X_s between minimum (0) and maximum ($L/2$) allowable values	Y_s	0.1	1
Fraction of excursion of X_f between minimum ($L/2$) and maximum (L) allowable values	Y_f	0	1
Operating altitude (km)	h_{alt}	17	25

Table 3 Design constants and their values

Name of the parameter	Symbol	Value
Specific mass of the envelope material (kg/m^2)	ρ_{env}	0.163
Specific mass of the solar array (kg/m^2)	ρ_{array}	0.12
Mass of the payload (kg)	M_{pay}	1000
Power to control system (W)	P_{control}	1000
Power to payload (W)	P_{pay}	10,000
Solar array efficiency	η_{SA}	0.12
Airship yaw angle (\circ)	ψ	90

For the mathematical convenience, the design variables are employed as follows:

$$\begin{cases} X_m = X(1) \\ R_0 = X(2) \\ R_1 = X(3) \\ C_p = X(4) \\ \phi = 3 + (3 \times X(5)) \\ L = 50 + (200 \times X(6)) \\ X_s = (L/2) \times X(7) \\ X_f = (L/2) + ((L/2) \times X(8)) \\ h_{\text{alt}} = 17 + (8 \times X(9)) \end{cases} \quad (6)$$

The design variables $X(5)$, $X(6)$, $X(7)$, and $X(8)$ are based on length-to-diameter ratio of individual lobe (ϕ), length of the envelope (L), starting point of the solar array (X_s), and ending point of the solar array (X_f). The last design variable $X(9)$ represents the operating condition. All the design variables are scaled between 0 and 1.

2.2 Aerodynamics Model

The drag force acting on airship has significant effect on its performance, power required, and dynamics. Hence, an accurate method for evaluation of drag is necessary. It may be evaluated experimentally or using sophisticated CFD analysis; however, it is computationally very expensive. In the present study, the zero-lift drag coefficient (C_{D0}) for unconventional airships has been estimated using rough approximation method given by Carichner and Nicolai [5]. The total drag of an airship at zero angle of attack can be obtained by

$$C_{D0} = C_{Dp} + C_{Df} + C_{D,\text{int}} + C_{D,\text{misc}} \quad (7)$$

The pressure drag coefficient (C_{Dp}) is evaluated based on rough approximation as 5% of the skin friction drag coefficient (C_{Df}) when $\text{FR} > 7$ and equal to the envelope skin friction drag for $\text{FR} < 3$ [10]. Skin friction drag coefficient (C_{Df}) is a major component of the total drag, which can be computed by using component build-up method. The pressure drag coefficient (C_{Dp}) is evaluated based on rough approximation as 5% of the skin friction drag coefficient (C_{Df}) when $\text{FR} > 7$ and equal to the envelope skin friction drag for $\text{FR} < 3$ [10]. Skin friction drag coefficient (C_{Df}) is a major component of the total drag, which can be computed by using component build-up method

$$C_{Df} = \sum_{i=1}^n (C_{Df})_{\text{comp}} = \sum \left(\frac{C_f \times FF \times S_{\text{wet}}}{\text{Vol}^{2/3}} \right)_{\text{comp}} \quad (8)$$

where C_f represents skin friction coefficient for each component, FF is form factor, and S_{wet} is wetted surface area of each individual components of airship. The components that have been considered for drag evaluation are envelope, landing system, cables, and tails. $C_{D,\text{int}}$ accounts for the interference between the parts of an airship. The drag of other components other than the envelope is included in the term $C_{D,\text{misc}}$.

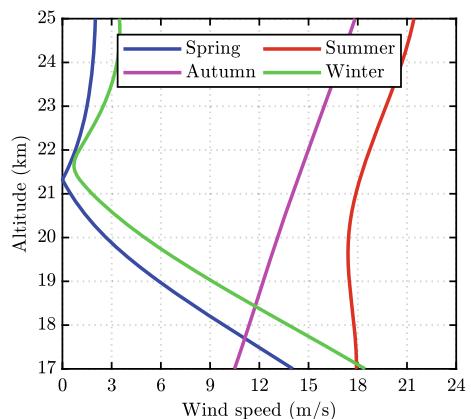
2.3 Environment Model

To investigate the effect of wind speed on the sizing of the stratospheric airship, the HWM14 routine developed by U.S. Naval Research Laboratory [6] to calculate the meridional and zonal components of the wind for one or more sets of geophysical data: latitude, longitude, and altitude is used in this study. The variation of wind speed with altitude for the position of $72^{\circ}88' \text{ E}$, and $19^{\circ}07' \text{ N}$ (viz Mumbai) on four specific days of the year is shown in Fig. 3. It can be seen from Fig. 3 that the wind speed in summer and autumn is higher than that in spring and winter. The wind speed varies from location to location. This difference in wind speed over the seasons causes significant change in the propulsive power required, and hence the area of solar array.

2.4 Irradiance Model

Solar irradiance is one of the important parameters that needs to be considered in the design of solar-powered airship. It greatly affects the temperature of the entire system and performance of the solar array on the envelope. It varies throughout a day/year which depends on various factors such as geographic location, time of day, day of

Fig. 3 Wind speed over different altitude



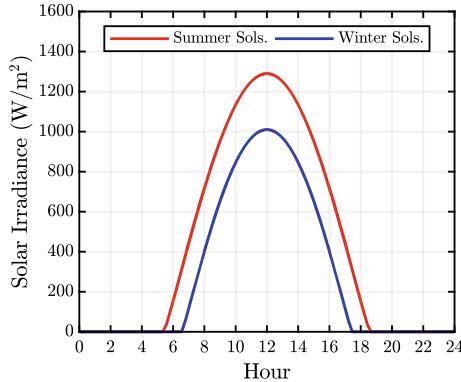


Fig. 4 Solar irradiance over a day

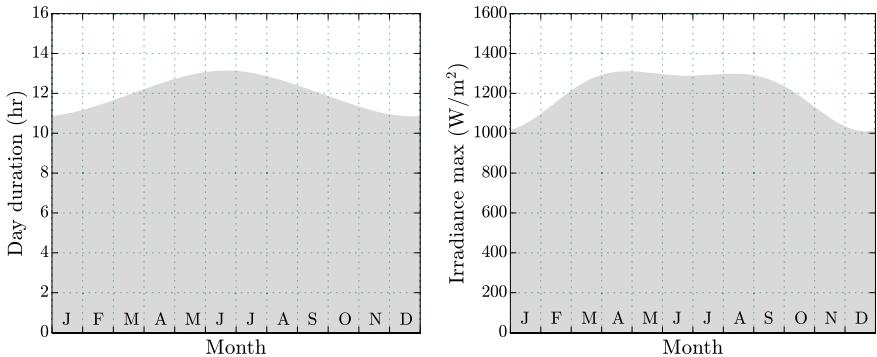


Fig. 5 Maximum irradiance and day duration throughout a year in Mumbai, India

operation, seasons, weather conditions, and altitude. In the present study, the model developed by Dai and Fang [11] is used. Figure 4 shows the variation of irradiance over a day at Mumbai, India.

The two parameters' maximum irradiance and duration of day time depends on location and day of operation. Figure 5 shows the effect of these parameters throughout the year for Mumbai, India. It can be seen that the maximum irradiance and duration of day time is less during winter due to the very low solar elevation. It is difficult to achieve long duration and continuous flight during winter season (the day lasting shorter than the night) without additional source of power.

3 Design Process

In the design process of an solar-powered airship, the area of solar array is an important parameter. It is necessary to estimate the accurate area of solar array to achieve the optimum value that meets the operating requirements and constraints.

The aim of the study is to minimize the area of solar array of a tri-lobed stratospheric airship with an user-defined design and operating requirements. The baseline airship configuration to be optimized is a combination of two semi-ellipsoids with design variables L , FR, X_s , and X_f that defines the geometry of the envelope and solar array.

The optimization problem involves four disciplines including environment, aerodynamics, energy, and geometry, and the design variables from these disciplines which are optimized to minimize the area of solar array and satisfies the constraints based on weight/lift equilibrium and energy balance. The shape of the envelope and layout of the solar array are optimized simultaneously to keep the solar array area minimum. The outputs from each discipline is coupled with other disciplines, since they are strongly associated with each other.

4 Optimization Problem Formulation

In the present study, the optimization problem is formulated to minimize the area of solar array based on penalty function method. The penalty function method helps to solve the constrained optimization problem using algorithms for unconstrained problems. The constraints are developed based on two important balances: weight-lift balance and energy balance.

Fitness function:

$$F(x) = A_{\text{array}} + \text{penalty} \sum_{i=1}^r (h_i^2(x)) + \text{penalty} \sum_{j=1}^m (g_j^2(x)) \quad (9)$$

Constraint (s.t.):

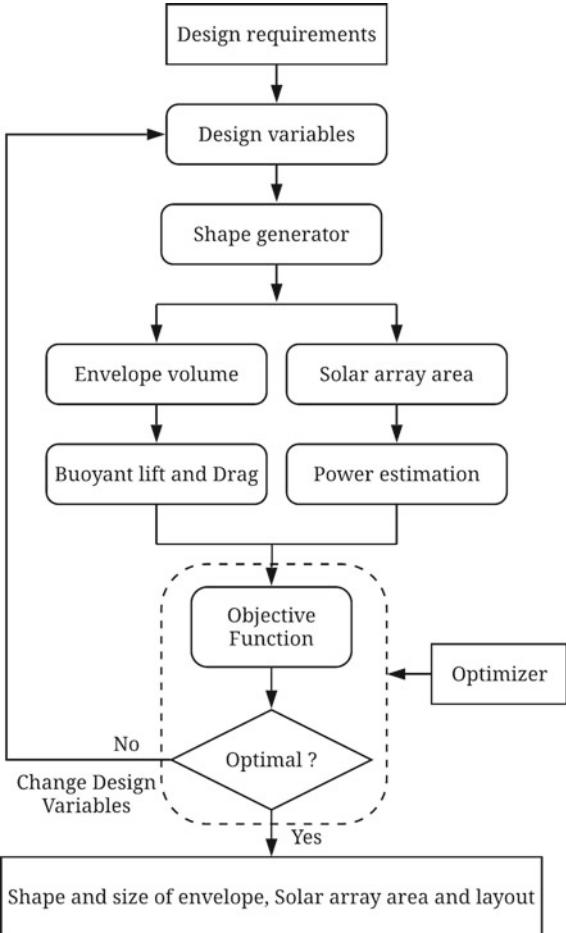
$$\begin{cases} g_1 \Rightarrow ((L_{\text{buoy}} + L_{\text{aero}}) - m_{\text{total}}g \geq 0 \\ g_2 \Rightarrow E_{\text{sup}} - E_{\text{req}} \geq 0 \end{cases} \quad (10)$$

The overview of the design optimization process followed in the present study is shown in Fig. 6.

4.1 Optimizer

Heuristic optimization techniques are widely used to find the global optima due to their efficiency and less time required for processing. These techniques do not require continuity in the problem definition or an initial input of design variables to perform the optimization, and they will search the entire design space to locate the global optima. The advantage over derivative-based optimization techniques is

Fig. 6 Optimization problem formulation



mostly they do not converge to the local optima. Among these techniques, bio-inspired population-based stochastic global optimization method, viz particle swarm optimization (PSO), is quite popular. PSO was first introduced by Eberhart and Kennedy in 1995 [12], and it mimicks the natural social behavior of flock of birds and schools of fish moving together. In this study, the PSO has been selected due to its performance over the multidisciplinary design optimization of unconventional airship configuration shown in the earlier work [13, 14]. The PSO used in the present study is based on the algorithm that is described in [12] with modifications suggested in [15, 16].

5 Results and Discussion

A multidisciplinary design optimization problem with nine design variables is posed and solved to minimize the area of array. The first six of the design variables define the shape and size of the envelope, the next two are related to the size and layout of the solar array, and the last one corresponding to the operating condition. The optimization is performed using an intelligent algorithm such as particle swarm optimization (PSO), since the problem to be optimized is multivariable with multiple constraints based on weight/lift balance and energy balance. The baseline solutions are obtained consisting of envelope of NPL profile. The values of distance of maximum diameter from the nose, radius of curvature of nose, radius of curvature of tail, and prismatic coefficient to define the NPL shape are 0.4320, 0.5884, 0.4250, and 0.6666, respectively.

Figure 7 shows that the area of solar array for final configurations is lower than that for the corresponding baseline solutions and is the lowest for configuration designed for operation on Spring equinox, i.e., March 20. As a result of sensitivity analysis, Fig. 8 shows the variation in area of solar array with fineness ratio (FR) for operation

Fig. 7 Optimal solar array area

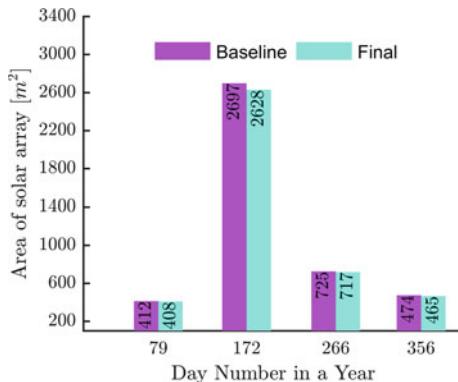
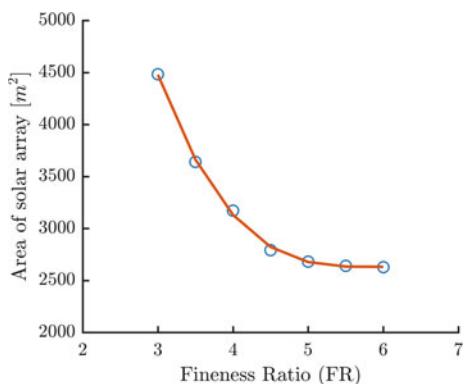


Fig. 8 Area of solar array versus FR



on summer solstice, i.e., June 21. As FR increases, there is a decrease in area of solar array due to change in frontal area and volume of the envelope results in lower drag, and hence, lesser area of solar array.

6 Conclusions

A methodology for conceptual sizing of a tri-lobed airship has been proposed, which takes into consideration the effect of four disciplines, viz envelope geometry, aerodynamics, environment, and energy. The methodology is coupled to an optimizer to minimize the area of solar array needed, while meeting all the specified requirements and constraints, for operating a stratospheric airship at a specific altitude and day of the year. Particle swarm optimization (PSO) technique is employed to obtain a series of feasible solutions with optimal geometry and layout of the solar array for different days of operation. It is seen that the area of solar array needed for the airship to meet the given requirements and constraints are 0.98–2.6% lower than that for the baseline airship envelope. It is also seen that as envelope fineness ratio (FR) is increased from 3 to 6, the area of solar array required for operating on one specific day can be reduced substantially.

References

1. M.I. Alam, R.S. Pant, Multidisciplinary approach for solar area optimization of high altitude airships. *Energy Convers. Manag.* **164**, 301–310 (2018)
2. J. Agte, T. Gan, F. Kunzi, A. March, S. Sato, B. Suarez, B. Yutko, Conceptual design of a hybrid lift airship for intra-regional flexible access transport, in *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition* (2010), p. 1391
3. B. Buerge, The suitability of hybrid vs. conventional airships for persistent surveillance missions, in *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition* (2010), p. 1014
4. A. Ceruti, P. Marzocca, Conceptual approach to unconventional airship design and synthesis. *J. Aerosp. Eng.* **27**(6), 04014035 (2013)
5. G.E. Carichner, L.M. Nicolai, *Fundamentals of Aircraft and Airship Design. Airship Design and Case Studies*, vol. 2 (American Institute of Aeronautics and Astronautics, 2013)
6. D.P. Drob, J.T. Emmert, J.W. Meriwether, J. Jonathan, M.E. Doornbos, M. Conde, G. Hernandez, J. Noto, K.A. Zawdie, S.E. McDonald, J.D. Huba, J.H. Klenzing, An update to the Horizontal Wind Model (HWM): the quiet time thermosphere. *Earth Space Sci.* (2015)
7. G.A. Khouri, J.D. Gillett (eds.), *Airship Technology. Cambridge Aerospace Series*, vol. 10 (Cambridge University Press, Cambridge, 1999), p. 33. ISBN: 0-521-430-747
8. M. Gertler, Resistance experiments on a systematic series of streamlined bodies of revolution—for application to the design of high-speed submarines. DTMB Report, Washington, DC, 1950
9. H. Liang, M. Zhu, X. Guo, Z. Zheng, Conceptual design optimization of high altitude airship in concurrent subspace optimization, in *50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition* (2012), p. 1180
10. A. Ceruti, D. Gambacorta, P. Marzocca, Unconventional hybrid airships design optimization accounting for added masses. *J. Aerosp. Sci. Technol.* **72**, 164–173 (2018)

11. Q. Dai, X. Fang, A simple model to predict solar radiation under clear sky conditions. *Adv. Space Res.* **53**(8), 1239–1245 (2014)
12. J. Kennedy, R. Eberhart, Particle swarm optimization, in *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia (1995), pp. 1942–1945
13. A. Ceruti, V. Voloshin, P. Marzocca, Heuristic algorithms applied to multidisciplinary design optimization of unconventional airship configuration. *J. Aircraft* **51**(6), 1758–1772 (2014)
14. M. Manikandan, R.S. Pant, Design optimization of a tri-lobed solar powered stratospheric airship. *Aerospace Sci. Technol.* **91**, 255–262 (2019)
15. E. Mezura-Montes, C.A. Coello Coello, Constraint-handling in nature-inspired numerical optimization: past, present and future. *Swarm Evol. Comput.* 173–194 (2011)
16. M.E. Pedersen, *Good Parameters for Particle Swarm Optimization* (Hvass Laboratories, Luxembourg, 2010)

Multi-objective Aerodynamic Optimization of a Hypersonic Scramjet Inlet



Lipi Roy and Pankaj Priyadarshi

Abstract The flow path of a three ramp, external compression, 2D, hypersonic scramjet inlet is optimized through multi-objective aerodynamic optimization. The two objectives of the study are maximization of the total pressure recovery (TPR) and minimization of exit Mach number. Constraints on minimum mass flow ratio (MFR) are enforced. The design variables include the lengths and the angles of the three ramps. A novel parameterization strategy has been used to maximize the feasible design space. Numerical simulations of viscous flow at Mach 6 are carried out using the open-source RANS CFD solver SU2. Validation studies have been attempted using NASA wind tunnel test data. CFD simulations were carried out on 400 configurations obtained through design of experiments. The results were postprocessed to obtain the TPR, MFR, and exit Mach number. Surrogate models of these objectives and the mass flow ratio constraint were constructed using the regression. These regression models were used in the optimization using the *fmincon* function of MATLAB by optimizing one objective while keeping the other objective as a constraint for multiple values of the second objective. It was seen that the total pressure recovery and the exit Mach number are conflicting objectives. The Pareto optimal set is obtained and the characteristics of the optimal solution are analyzed.

Keywords Scramjet · Optimization · Hypersonic inlet · Aerodynamics

Supported by Vikram Sarabhai Space Center, ISRO.

L. Roy

Indian Institute of Space Science and Technology, Thiruvananthapuram, India
e-mail: lipiroydel@gmail.com

P. Priyadarshi (

Vikram Sarabhai Space Center, Thiruvananthapuram, India
e-mail: pankaj.priyadarshi@gmail.com

1 Introduction

Supersonic Combustion Ramjet (Scramjet) engines are required to produce thrust at hypersonic Mach numbers. So an optimal design of each of its components (inlet, combustor, and nozzle) is essential to ensure maximum thrust and efficiency. In the present work, an optimal shape of an external compression scramjet inlet is arrived at. In an external compression inlet, the shock on cowl lip ensures no spillage [1]. So the efficiency of a supersonic inlet is primarily defined by:

1. Total pressure recovery
2. Mass flow recovery
3. Decrease in Mach number.

In a real-flight scenario, viscous/boundary effects, turbulence, and shock-boundary layer interactions come into play and flow field becomes too complicated to resolve analytically [2]. Hence, computational fluid dynamics (CFD) simulations are done to model these effects effectively.

In the present work, a novel parameterization strategy is employed and a large number of air intake designs are simulated to obtain total pressure recovery (TPR), mass Flow recovery (MFR), and exit Mach number and then, the optimization process is carried out.

2 Literature Review

Hypersonic air breathing propulsion offers the potential for reliable and economical transport for access to space and high-speed atmospheric cruise [3]. Hence, it has become a popular area of research in the past few decades. Many studies [4–6] have been carried out to understand the flow characteristics of a scramjet air intake.

In one of the earliest studies in scramjet optimization , Zha et al. [7] use a computer-aided automated optimization technique to maximize the total pressure recovery (TPR) of the scramjet inlet. Carrier et al. [8] elaborate a multi-criteria design optimization technique using two different solvers, to maximize the mass capture ratio in accelerated flight and TPR in cruise and maneuver. The genetic algorithm GADO is used for optimization in this case. Chen et al. [9] use a GA-based parabolized Navier Stokes solver and SSPNS for flow visualization to optimize TPR, static pressure, and drag coefficient. Since there are conflicting objectives, Pareto fronts are obtained as a trade off between these. With the scientific advancements in optimization techniques, researchers sought to using evolutionary algorithms like non-dominated sorting genetic algorithm (NSGA II) to make the optimized geometry better and the process less time consuming. While Ogawa et al. [3] use NSGA II to optimize compression efficiency, drag, adverse pressure gradient while executing a constraint on the exit temperature, Wu et al. [10] use this to optimize TPR, L/D ratio of the engine, specific impulse of the engine and maximum temperature on the surface of the vehicle.

Although these methods prove to be quite efficient even when dealing with a time-consuming computation of this nature, there has been no effort in the direction of eliminating the infeasible region from the design space. This has been attempted in this paper. A novel parameterization strategy has been employed to maximize feasible designs for the sampled configurations in the design space. This has been done in an effort to further reduce the CPU time and make the optimization process more efficient.

3 Methodology

3.1 CFD Solver and Its Parameters

The open-source CFD solver SU2 [11, 12] is employed for flow computation. For flow simulation, the CFL number taken is 5, numerical methods are JST (first 30k iterations) and the ROE (next 30k iterations). The SA turbulence model has been used and a residual reduction of 5 orders of magnitude is set as the convergence criteria.

3.2 Validation

Validation studies were carried out using NASA experimental results of flow in Inlet-combustor isolators [13]. The geometry and mesh of the given configuration [13] was made in Gmsh [14]. Different resolution grids were generated and final results were compared to establish grid independence. Figure 1a shows the grid and boundary conditions employed. CFD simulations were carried out using three different grid sizes of (1) 125×103 , (2) 150×123 , (3) 188×155 cells, respectively. It was seen that grid 2 showed 4.1% deviation of result from grid 1 where as grid 3 showed only 1.4% deviation from grid 2. So it was decided to use grid 2 for further simulations. Figure 1b shows the results of the grid independence study. Tables 1 and 2 show the grid details and the flow conditions, respectively.

The results of the CFD simulations were compared with the experimental results and are shown in Fig. 2. The plot used for comparison depicts the ramp pressure variation for thick boundary layer. For this purpose, a 12 in. plate was placed before the ramp, in the experiment, is simulated by extending the ramp wall by the same amount, in the simulation. It is seen that CFD simulation captures the trend of pressure variation quite well. However, there is a difference of 6.5% in the peak values of pressure measured.

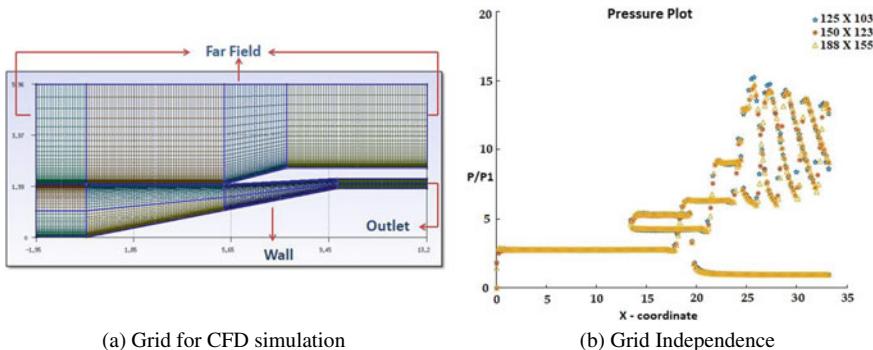


Fig. 1 Validation case simulation

Table 1 Grid details

Parameter	Value
Resolution	150 × 123
y+	1
Boundary conditions	Farfield, outlet, wall

Table 2 Flow conditions

Parameter	Value
Mach number	4.03
Reynold's number	21×10^6
Total pressure	200 psia
Total temperature	500 °R
Static pressure	1.266 psia

Table 3 Design variable definition

S. No.	Design variable	Limits	Definition of geometric parameter
1	R_{x1}	0–0.2	$x_1 = -((R_{x1} * 0.236) + 2.16)$
2	R_{x2}	0–1	$x_2 = x_{2,\text{low}} + R_{x2} * (x_{2,\text{high}} - x_{2,\text{low}})$
3	R_{x3}	0–1	$x_3 = x_{3,\text{low}} + R_{x3} * (x_{3,\text{high}} - x_{3,\text{low}})$
4	R_{x4}	0–1	$x_4 = x_3 + R_{x4} * (-x_3)$
5	R_{x5}	0.2–0.6	$x_5 = R_{x5}$
6	R_{x6}	0.2–0.6	$x_6 = x_5 + (R_{x6} * 2.1)$
7	$R_{\delta 1}$	0–1	$\delta_2 = \delta_{2,\text{low}} + R_{\delta 1} * (\delta_{2,\text{high}} - \delta_{2,\text{low}})$
8	$R_{\delta 2}$	0–1	$\delta_3 = \delta_{3,\text{low}} + R_{\delta 2} * (\delta_{3,\text{high}} - \delta_{3,\text{low}})$
9	R_{y4}	0.3–0.8	$y_5 = R_{y4} * y_4$

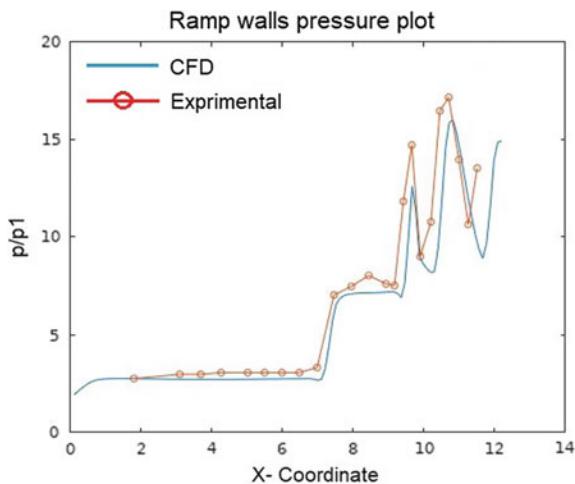


Fig. 2 Experimental and CFD result comparison

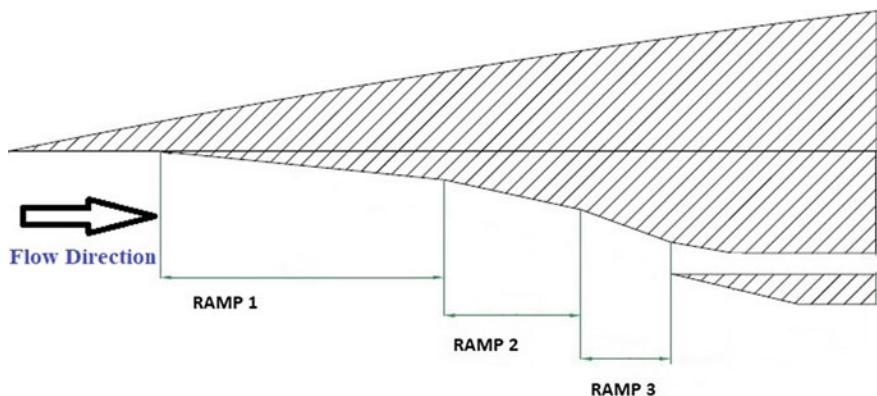


Fig. 3 Reference geometry

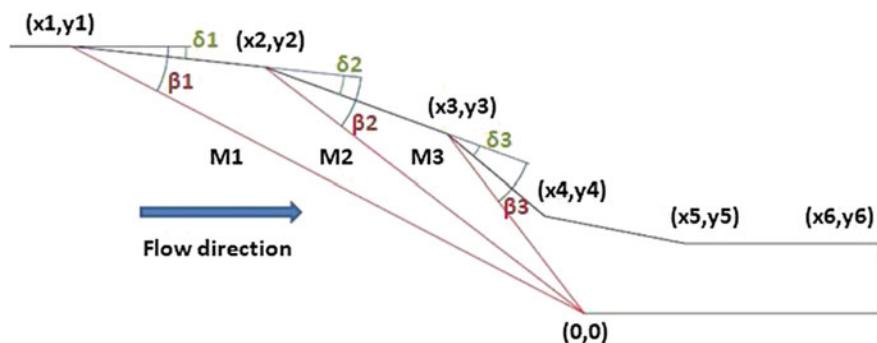


Fig. 4 Schematic of the supersonic intake

3.3 A Novel Parametrization Strategy

Figure 3 shows the schematic of the reference geometry used for optimization. Optimization requires searching the feasible design space till the optimum point is reached. The basic design variables are the ramp starting locations x_i and the ramp angles δ_i (Fig. 4). To create the design space, if these variables are varied between arbitrary intervals, many infeasible geometries can be produced. Simulating these will be computationally expensive and not useful. To rule this out in the initial stage itself, a novel methodology is proposed which allows the design variables, only near the configurations with shock on lip condition estimated from inviscid computation.

Consider Fig. 4. All the given co-ordinates need to be parameterized using appropriate variables. The inlet Mach is taken as 6, cowl lip as origin, and the height of the inlet (y_1) is fixed. Also, before the first ramp, the flow can be considered to be inviscid since the wall effects are minimal. Due to this imposition, the initial ramp angle δ_1 can be calculated from compressible flow relations if β_1 is known. β_1 can be easily calculated from geometry if x_1 is known, since it is known that ramp shocks interact with the cowl lip for optimal configuration [15]. So the first variable R_{x1} is used to fix x_1 in the interval $[-2.2072, -2.16]$ which defines the neighborhood of x_1 in the reference geometry. Details pertaining to R_{x1} are given in Table 3.

Since x_1 , β_1 and hence δ_1 is now known, the calculation of the starting point of the next ramp only requires the x-coordinate x_2 since y_2 can be automatically calculated. Location of x_2 depends upon the next shock angle β_2 . Hence, a minimum and maximum value of β_2 is required to set the range for x_2 . To find the minimum value of β_2 , the condition of Mach wave is invoked and the maximum value of β_2 is taken to be around 30° , which is a few degrees above the inviscid case β_2 . Range of x_2 is calculated from this range of β_2 , using geometry and the second design variable R_{x2} is used to fix x_2 within this range, i.e., as R_{x2} varies between 0 and 1, x_2 varies from its minimum to maximum value. Using this x_2 , β_2 is calculated and an inviscid δ is calculated from theory. But since the flow is no longer inviscid, the real δ_2 is selected from a range of $\pm 4^\circ$ around this inviscid δ . A third variable, $R_{\delta 1}$ spans the space between $[\delta - 4^\circ, \delta + 4^\circ]$ as it itself varies from 0 to 1. With the δ_2 now known, y_2 can be calculated from geometry. Similar procedure is followed to define the variables R_{x3} and $R_{\delta 2}$ associated with the point 3.

For x_4 , variable R_{x4} is used to span the distance between x_3 and the cowl lip. Variables R_{y4} , R_{x5} , R_{x6} are used to locate co-ordinates y_5 , x_5 and x_6 within an estimated interval. Details of all the design variable limits and how they are used to define the given geometric parameters are shown in Table 3.

3.4 Grid Generation and CFD Simulations

Structured grid with clustering near the walls was created similar to the validation case. The domain size extension, discretization, boundary conditions, and $y+$ calcu-

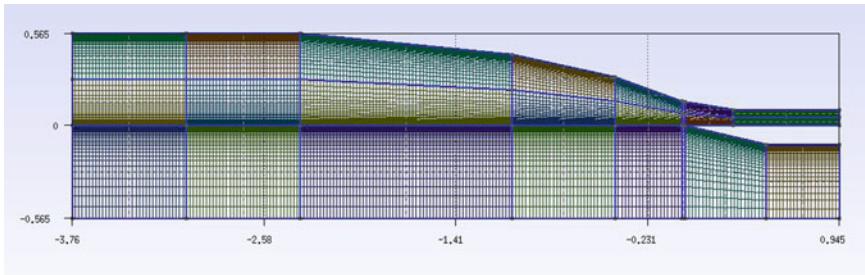


Fig. 5 Structured grid of the test case with clustering near the walls

Table 4 Flow parameters for test case

Parameter	Value
Mach number	6
P_∞	4490 Pa
T_∞	218 K
Reynolds number	8.1e6

lation were done similar to the validation case, using the free stream conditions for this case. A grid of 236×213 elements was created using the open source solver Gmsh as shown in Fig. 5.

The solver parameters were kept the same as the parameters of the validation case. The flow parameters pertaining to the test case are tabulated in Table 4. Once the variables were selected and their bounds were finalized, *Latin Hypercube sampling* was used to create 400 design points. Flow simulation was done for each of them. Out of 400, only 325 results converged and could be used for post processing. Python scripting in Paraview was done to extract TPR, MFR, and exit Mach number from each of the 325 simulations.

4 Surrogate Model

Surrogate models of total pressure recovery (TPR), mass flow recovery (MFR) and Mach numbers were created using the regression learning app in MATLAB [16]. The model fitting results are shown in Fig. 6. It was observed that the Mach number and the MFR model fitted quite well with average error percentage less than 1%.

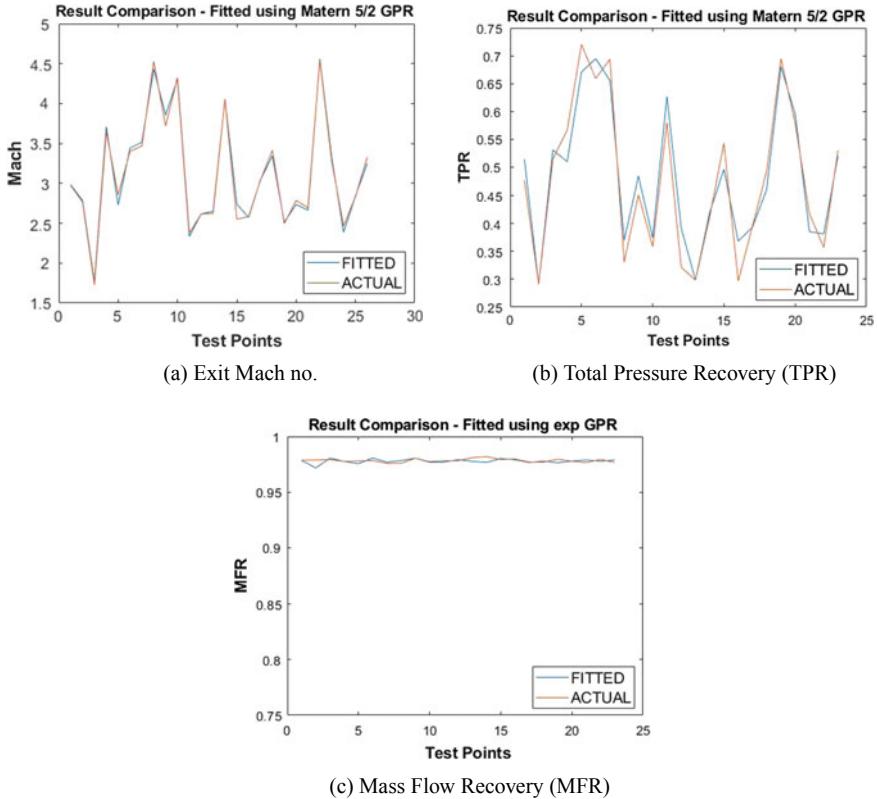


Fig. 6 Comparison of surrogate model results and actual results for exit Mach, TPR, and MFR

However, the average error slightly increased to 6.67% for the TPR model. Alternate surrogate models like the Artificial Neural Networks (ANN) or the radial basis functions (RBF) would be attempted in the future.

4.1 Multiobjective Optimization

The **Optimization problem** was defined as follows: Maximize TPR and minimize exit Mach number such that $MFR > 97\%$. Optimization, done using the *fmincon* function of MATLAB [16], was carried out using the ϵ -constraint method, that is, the TPR was maximized multiple times, with a different constraint on the exit Mach number ($M \in [2.5, 3.5]$). Alternate optimization strategies like multiobjective evolutionary algorithms, SNOPT, etc., will be tried in the future.

5 Results and Discussion

5.1 CFD Simulations

Flow visualization of the CFD simulations on the reference configuration was done using Paraview [17]. Figures 7 and 8 show the variation in flow quantities as the flow passes through the air intake. As can be seen from Fig. 7, the pressure variation is according to the expected results. There is a sharp increase in pressure across the oblique shock waves. Beyond the cowl, there are multiple shock wave reflections to satisfy the wall boundary condition. In Fig. 8, the formation of boundary layer can be clearly observed. The increase in the thickness of boundary layer, and its interaction with the cowl shock wave can be clearly seen in the simulation result. A small spillage of mass flow rate can be observed at the inlet. This is because the shock wave does not exactly touch the lip. This is what is to be optimized for, as the best inlet condition is when the oblique shock touches the cowl lip.

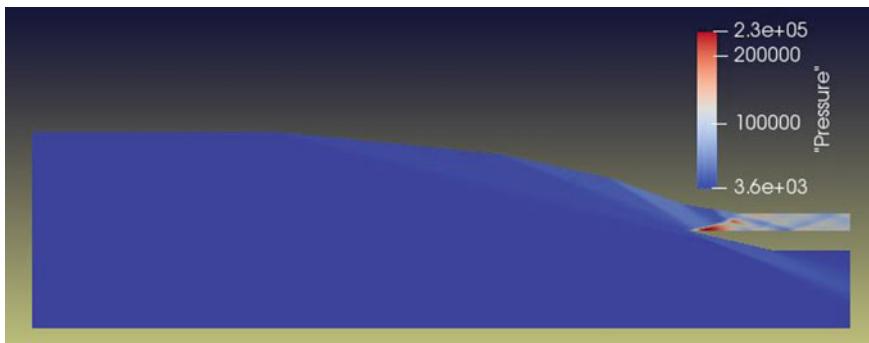


Fig. 7 Pressure variation

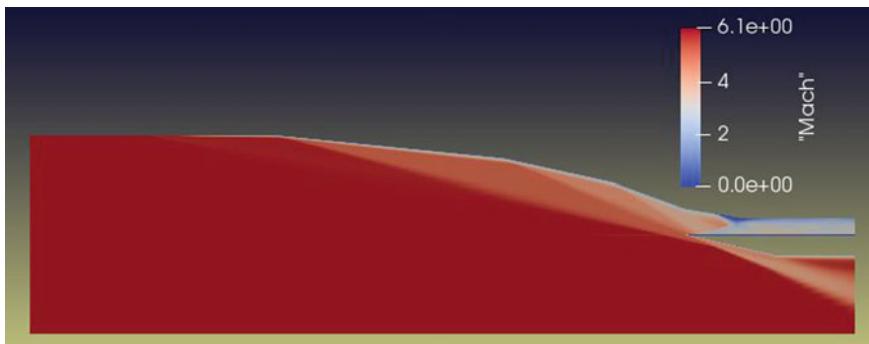


Fig. 8 Mach number variation

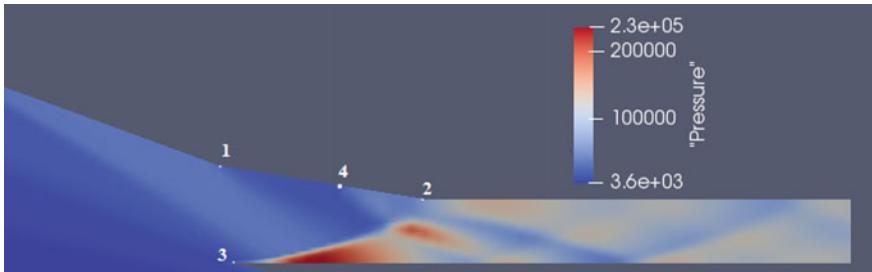


Fig. 9 Pressure variation in the cowl

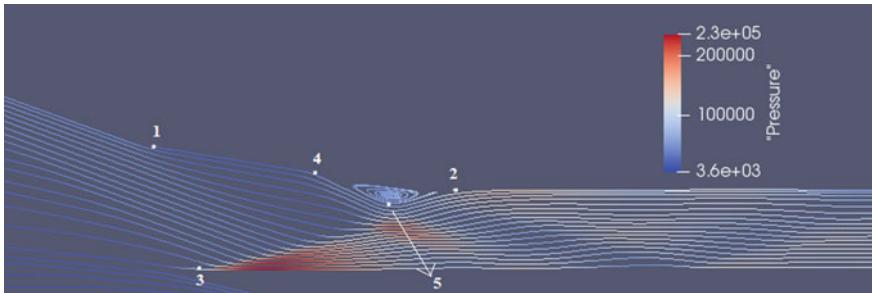


Fig. 10 Streamlines in the cowl

Pressure variation close to the cowl region can be clearly observed in Fig. 9. The shock waves that almost touch the cowl lip (marked as 3) reflect and become slightly curved due to their interaction with expansion waves emanating from the corner (marked as 1). Additional compression is seen at point 4 which lies on the ramp between points 1 and 2. The source of this was investigated and streamlines are plotted as shown in Fig. 10.

It can be seen that the separated flow region exists between point 4 and point 2. This separation of boundary layer occurs due to the adverse pressure gradient from the reflected shock wave at point 3. Due to the formation of this separation bubble, the streamlines get curved inside and resemble the shape of a convergent divergent nozzle with its throat marked as point 5. Therefore, when the supersonic flow is between point 4 and point 5, compression takes place and when the flow is between point 5 and point 2, expansion takes place. Now, to satisfy the wall boundary condition at 2, another shock wave originates and hits the cowl wall where it gets reflected again. These waves are again propagated downstream and reflected multiple times.

5.2 MultiObjective Optimization

Pareto front obtained from the multiobjective constrained optimization is shown in Fig. 11a. The optimization problem requires the maximization of TPR and minimization of exit Mach number, which are conflicting objectives, as can be seen from the Pareto front. To increase the value of optimal TPR, the optimal exit Mach number will also have to increase, and vice versa. Thus, one of optimum points should be chosen from the Pareto front which satisfies the particular design requirements.

An exercise was carried out to check the MFR at 3 selected points from the Pareto front: those with constraint on exit Mach as 2, 2.5, and 3. It was noted that MFR > 97% for all these cases. This rendered the MFR constraint inactive. This was concluded to be a result of careful parameterization of the geometry which eliminated the infeasible region and reduced the feasible region to the vicinity of optimal points.

Figure 9b and c show how the design variables vary along the Pareto front. The initial point x_1 (Fig. 11b) and initial δ_1 (Fig. 11c) seem to be fixed for each optimal TPR. It was noted that they were fixed at the maximum possible x coordinate allowed

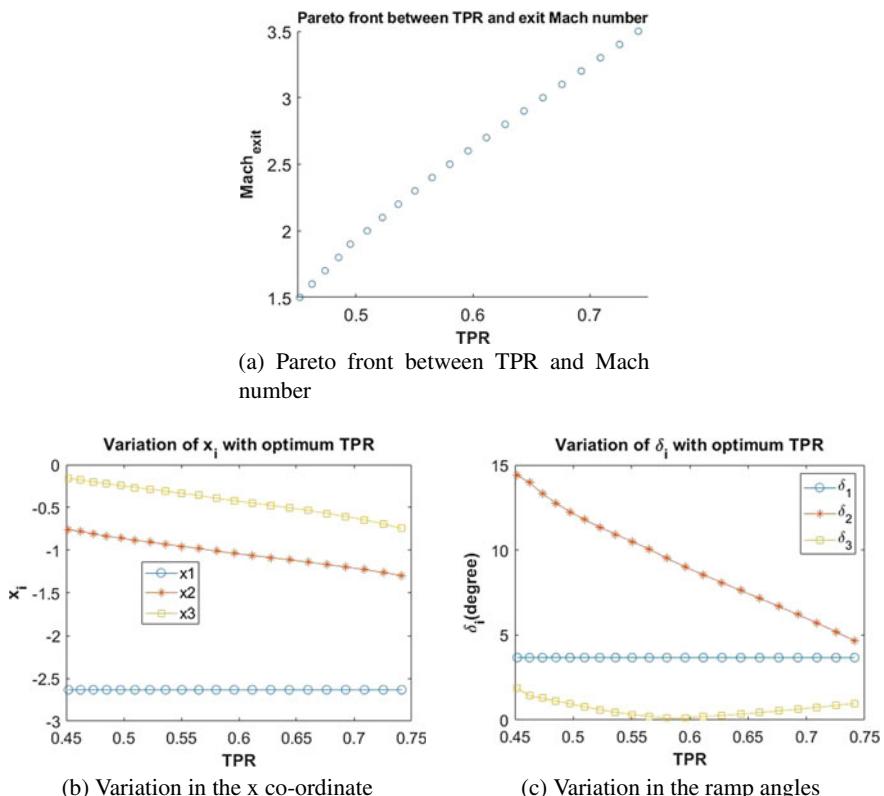


Fig. 11 Multiobjective optimization results

by the given range of parameterization. This means a requirement of searching the design space in the neighborhood of this point to further refine the optimal value, as a next step. As the optimum TPR value increases, x_2 and x_3 become more negative (Fig. 11b), hence moving away from the cowl. Also, δ_2 decreases sharply as the optimum TPR value increases (Fig. 11c). This is because as ramp angles reduce, weaker shocks would form and the loss in total pressure will be less. This comes naturally at the cost of exit Mach number. The angle of the third ramp (δ_3) does not show a monotonic trend with increasing optimum TPR.

6 Conclusion

Multiobjective flow path optimization of a hypersonic scramjet air inlet has been carried out. The flow simulations were carried out using SU2 and the results were validated against NASA wind tunnel data. A novel parameterization strategy was employed to minimize the infeasible points in the design space. Four hundred configurations were sampled in the design space and simulated using SU2. Results of TPR, MFR, and exit Mach number were extracted from each of the 325 converged simulations and the three surrogate models were created. These surrogate models were used to optimize the air intake. The TPR was maximized and the exit Mach number was minimized. It was seen that these objectives are conflicting, i.e., for higher optimal TPR, the exit Mach number is also higher. Owing to the novel inviscid oblique shock-based parameterization, the MFR exceeded 97% for most of the sampled configurations. Variation of design variables x_i and δ_i were obtained and analyzed with increasing value of optimum TPR.

References

1. S. Farokhi, *Aircraft Propulsion*, 2nd edn. (Wiley, 2014)
2. W.H. Heiser, D.T. Pratt, *Hypersonic Airbreathing Propulsion* (AIAA, 1994)
3. Hideaki Ogawa, Russell R. Boyce, Physical insight into scramjet inlet behavior via multi-objective design optimization. *AIAA J.* **50**(8), 1773–1783 (2012)
4. C. Herrmann, W. Koschel, Experimental investigation of the internal compression inside a hypersonic intake, in *38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit* (2002). <https://doi.org/10.2514/6.2002-4130>
5. B.U. Reinartz, C.D. Herrmann, J. Ballmann, W.W. Koschel, Aerodynamic performance analysis of a hypersonic inlet isolator using computation and experiment. *J. Propul. Power* **19**(5), 868–875 (2003)
6. J. Haberle, Ali Gulhan, Investigation of two-dimensional scramjet inlet flowfield at Mach 7. *J. Propul. Power* **24**(3), 446–459 (2008)
7. G.-C. Zha, D. Smith, M. Schwabacher, K. Rasheed, A. Gelsey, D. Knight, M. Haas, High-performance supersonic missile inlet design using automated optimization. *J. Aircr.* **34**(6), 697–705 (1997)

8. G. Carrier, D. Knight, K. Rasheed, X. Montazel, Multi-criteria design optimization of two-dimensional supersonic inlet, in *39th Aerospace Sciences Meeting and Exhibit* (Aerospace Sciences Meetings)
9. B. Chen, X. Xu, G. Cai, Single- and multi-objective optimization of scramjet components using genetic algorithms based on a parabolized Navier-Stokes solver, in *Collection of Technical Papers—AIAA/ASME/SAE/ASEE 42nd Joint Propulsion Conference* (2006). DOIurl<https://doi.org/10.2514/6.2006-4686>
10. P. Wu, Z. Tang, J. Sheng, Multidisciplinary optimization of airbreathing hypersonic vehicle. *Int. J. Mod. Phys. Conf. Ser.* **42**
11. T.D. Economon, F. Palacios, S.R. Copeland, T.W. Lukaczyk, J.J. Alonso, SU2: an open-source suite for multiphysics simulation and design. *AIAA J.* **54**(3), 828–846 (2016)
12. H.L. Kline, T.D. Economon, J.J. Alonso, *Multi-Objective Optimization of a Hypersonic Inlet Using Generalized Outflow Boundary Conditions in the Continuous Adjoint Method* (2016). <https://doi.org/10.2514/6.2016-0912>
13. S. Emami, C.A. Trexler, A.H. Auslender, J.P. Weidner, *Experimental Investigation of Inlet-Combustor Isolators for a Dual-Mode Scramjet at a Mach Number of 4*, NASA Technical paper 3502 (1995)
14. C. Geuzaine, J.F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built in pre and post processing facilities. *Int. J. Numer. Methods Eng.* **79**(11), 1309–1331 (2009)
15. J. Seddon, E. Goldsmith, *Intake Aerodynamics* (Collins, London, 1985)
16. Matlab opptimisation toolbox, Regression learner app, Version, *The MathWorks* (Natick, MA, USA, 2018)
17. J. Ahrens, B. Geveci, C. Law, ParaView: an end-user tool for large data visualization, in *Visualization Handbook* (Elsevier, 2005) (ISBN-13: 978-0123875822)

Study of Wing Flutter at Preliminary Stages of Design



A. T. Sriram and H. K. Narahari

Abstract Study of problems associated with aircraft flutter requires understanding of aerodynamics and structural dynamics. As the aircraft speed increases, the energy coming to structure causes the wing to deflect and in turn the deflection is in such way that it adds energy back to the structure. The positive coupling between wing deflection and aerodynamic forces enables self-sustained oscillation, which is generally referred as flutter, and it can cause structural failure. A simple spring mass system for vibration with external source describes the basic phenomena. Linear model of the problem was well studied by various authors. With the current capabilities of MATLAB/Simulink, it is possible to visualize the oscillations. The problem was simulated in the previous study and verified with available data in the literature. In the present study, the mass, moment of inertia and spring stiffness in heave and pitch motions have been varied. A 5% reduction in mass from the baseline mass 51.5 kg reduces the flutter speed of 1 m/s from the baseline speed 188.65 m/s. The basic premise is to shift the flutter speed upward and beyond the aircraft flight envelope. This will minimize the probability of aircraft entering into flutter. A surrogate meta-model is developed based on Kriging method. Genetic algorithm is used on the meta-model to optimize the design variables. The optimum value of flutter speed in the given range of design variables is about 7.4 m/s more than the baseline case. This study describes a systematic approach on implementation of multidisciplinary design optimization on a simple problem.

Keywords Flutter · Surrogate model · Genetic algorithm · Optimization

A. T. Sriram (✉) · H. K. Narahari
Ramaiah University of Applied Sciences, Bangalore 54, India
e-mail: atsriram.aae.et@msruas.ac.in

H. K. Narahari
e-mail: [nahari.aae.et@msruas.ac.in](mailto:narahari.aae.et@msruas.ac.in)

1 Introduction

A system-level design requires knowledge of various disciplines and the interconnecting mechanism. In the recent times, significant numbers of studies were carried out on design optimization to demonstrate its capabilities. Currently, available tools and techniques are quite easy to implement for a particular problem of interest and also visualize the results. In order to demonstrate the current capabilities, wing flutter problem is considered. A detailed simulation of wing flutter requires two-way coupling between computational fluid dynamics and computational structural dynamics solvers on moving grid [1–3]. Such simulations require more computing power and computational time. This technique will be useful on important design points and final design. In the preliminary stages of design, a simplified approach is necessary to study the influence of design parameter and arriving optimum design. Simplified methods were reported well [4, 5] for advanced learning purpose. The very early work of Theodorsen [6] forms the foundation for estimating flutter speed. Later, modified strip analysis method was developed from Theodorsen's method. For aerodynamic calculations, doublet lattice method (DLM) was introduced in the early 70s. Frequency-based methods, such as p - k and U - g methods, were developed immediately after Theodorsen's analysis of the flutter problem. These methods generally solve the eigenvalue problem.

In most of the cases, the pitch and heave motions of airfoil are considered to represent the flutter problem. A simple sketch of the airfoil is shown in Fig. 1. The equation of motion relates aerodynamic forces and moments. As a next step from wing configuration, leading and trailing edge control surfaces were added in the

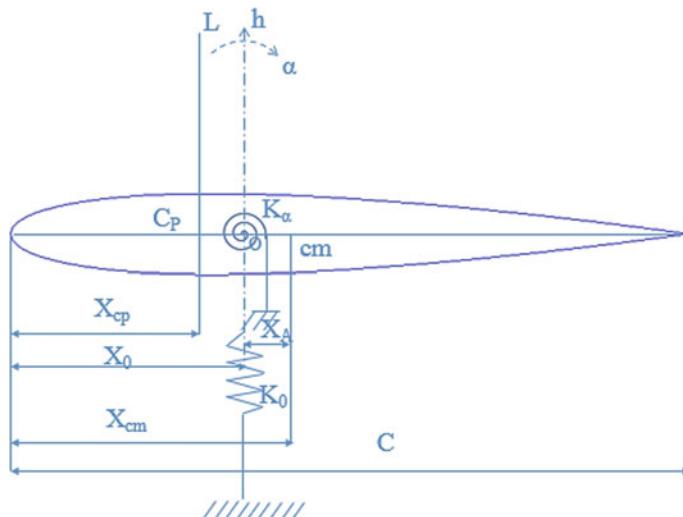


Fig. 1 Schematic representation of airfoil with two degrees of freedom

system to study their effect. It is also possible to include fuselage and develop spring mass balance system for entire aircraft. As a first step, wing alone configuration is considered for optimization of flutter speed within the small variation in major design parameters.

2 Problem Formulation and Solution Methods

The heave and pitch equation of motion of the airfoil serves as basic equations. In the earlier study [7], flutter speed was estimated based on eigenvalue problem [4] and also developed Simulink model to simulate the problem on time domain method. The design parameters of mass, moment of inertia, and stiffness in heave and pitch motion are varied for 5%. The optimum value of flutter speed in the design space obtained with standard methods is described below.

2.1 General Framework for Solution Method

The optimization problem is solved in the framework of surrogate model. The well-known Kriging method is considered in this study. This method was considered in other aerodynamic shape optimization problems. For the case of optimization,

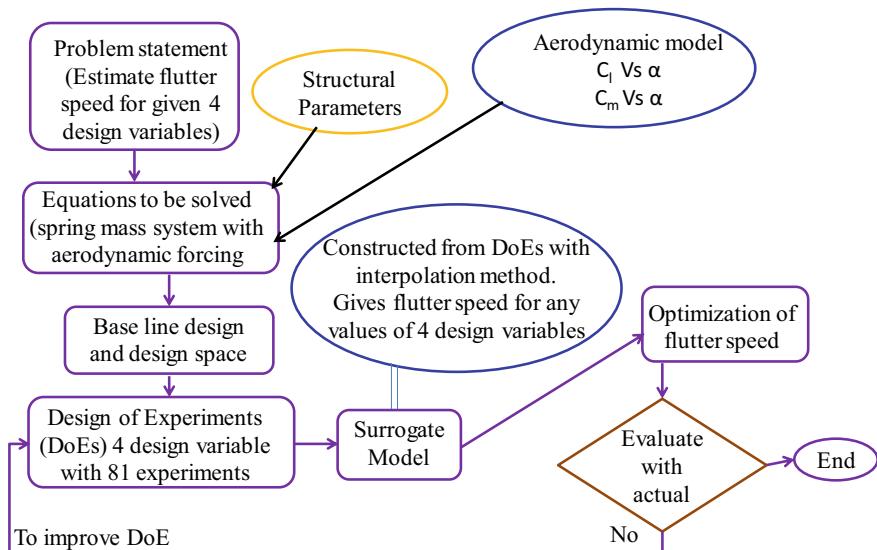


Fig. 2 Flowchart to describe the solution procedure

Table 1 Details of various cases

Sl. No.	Case	M (kg)	I_α (kg m ²)	K_h (N/m)	K_α (Nm/rad)	Flutter speed (m/s)	
						Frequency domain	Time domain
1	Baseline	51.500	2.275	50,828.00	35,923.00	188.65	184.6
2	Optimized with 50 generations	53.656	2.162	48,675.12	37,517.16	195.75	191.5
3	Optimized with 100 generations	53.632	2.164	48,348.32	37,621.89	196.26	192.0

genetic algorithm is used. The flowchart in Fig. 2 provides the steps involved in this approach.

2.2 Equations to Be Solved

The equations of motion describing heave and pitch motions are written as follows.

$$m\ddot{h} - mX_A\ddot{\alpha} + C_h\dot{h} + K_hh = L \quad (1)$$

$$mX_A\ddot{h} - I_\alpha\ddot{\alpha} + C_\alpha\dot{\alpha} + K_\alpha\alpha = M \quad (2)$$

The geometric and structural properties are considered from the literature [8]. In the case of aerodynamic model, the linear approximation for coefficient of lift curve slope with respect to angle of attack is used. Hence, the lift force is estimated by equating $dC_l/d\alpha$ to the value of 2π . This assumption is valid due to the fact that most of aircraft operations takes place in the range. It is also used in literature [4, 5]. Table 1 in Sect. 3.3 provides baseline values for design parameter.

2.3 Surrogate Modeling

From the four design variables, with three variations, 81 sample points (3^4) are created and the flutter speed is obtained from eigenvalue approach. Now the flutter velocity is written in Kriging method as

$$p(X) = \mathcal{F}(X) + Z(X)$$

where X is a vector, $\mathcal{F}(X)$ is a regression function, and Z is a random process with mean 0, variance σ^2 , and a correlation matrix R . The predictor flutter velocity for the Kriging method is

$$\hat{p}(X) = f(X)^T \beta^* + r(X)^T \gamma^*$$

The additional relations are explained in Ref. [9] with details. It tries to provide zero error at the sample points.

2.4 Genetic Algorithm for Optimization

For the case of optimization, genetic algorithm is used in this study. Though other traditional methods are applicable for this problem, the applicability of genetic algorithm in other problems associated with aerospace engineering was the motivation for selecting this approach. Hence, this approach serves as general platform for optimization.

3 Results and Discussions

3.1 Baseline Case and Design of Experiments (DoEs)

The flutter speed predicted from frequency domain approach is 188.65 m/s. The solution from time domain approach is shown in Fig. 3 for the verification of results. It is noticed that the flutter speed from time domain approach is slightly lesser than that of frequency domain approach. Even though small variation in speed, there is significant changes near the flutter speed. The flutter speed at various DoEs conditions is estimated from frequency domain approach. The time domain approach is used at the end for verification of the prediction.

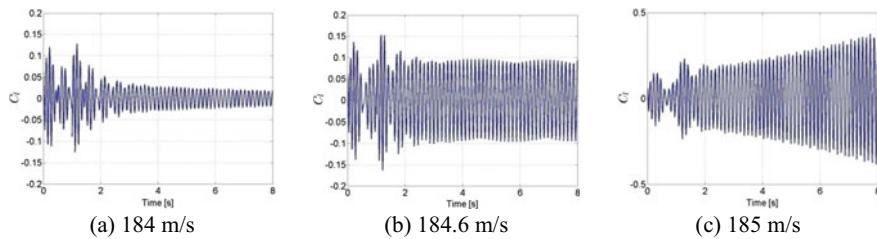


Fig. 3 Time histories of aerodynamic lift coefficient at three different inflow velocities for the baseline case [7]

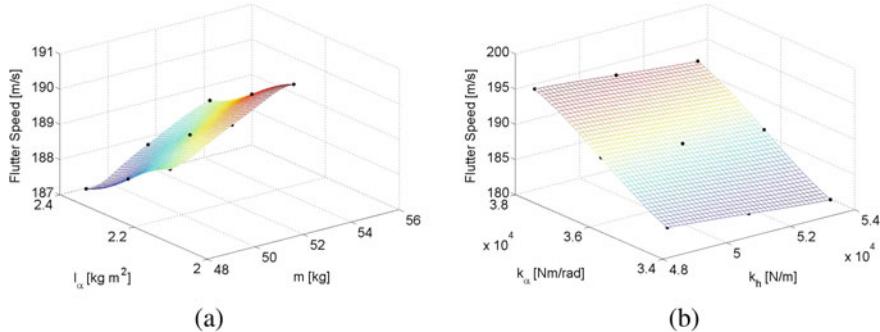


Fig. 4 Response surface generated from Kriging method to describe the influence of design parameters by keeping other parameters at baseline value. The symbols represent actual values from design of experiments. **a** Effect of mass and moment of inertia. **b** Effect of spring stiffness due to heave and pitch motion

3.2 Kriging Surrogate Model to Develop Response Surface

The 81 DoEs with four design variables and the estimated flutter speed at the DoE points from frequency domain approach serve as input from surrogate model. Figure 4 shows the response surface for mass and moment of inertia variation as well as spring stiffness variation. It is noticed that the variation in spring stiffness has larger influence of flutter speed.

3.3 Optimization of Design Variable with the Help of Genetic Algorithm

MATLAB tool box on genetic algorithm [10] is used in this study to optimize the design variables. The lower and upper bounds of design variable are specified. The population size of 20 is chosen. In order to assign minimization problem, the inverse of flutter speed is considered. Figure 5 shows the fitness value for various generations. The optimized design variables and the flutter speed for 50 generations and 100 generations are listed in Table 1. Figure 6 shows the flutter speed in time domain simulation with Simulink. Additional studies on improving the surrogate model with a greater number of points near the optimum value were conducted. As shown in Fig. 4, the surrogate model is a smooth surface; hence, no significant improvement on flutter was observed. Hence, it is not further discussed here.

Fig. 5 Convergence of inverse of fitness function with generation in genetic algorithm

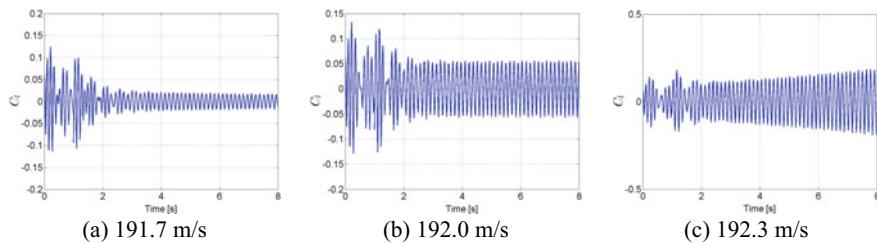
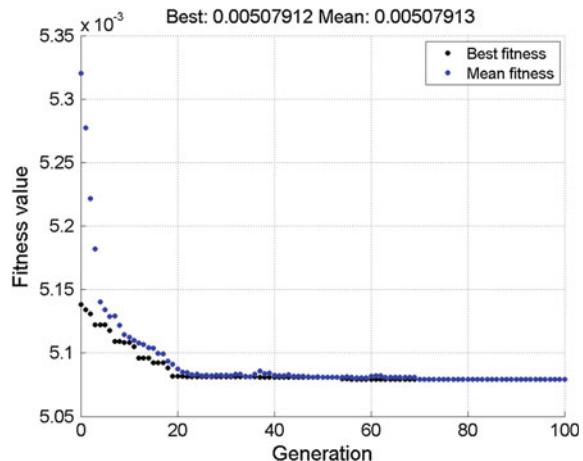


Fig. 6 Time histories of aerodynamic lift coefficient at three different inflow velocities for the optimized case

4 Conclusions

The classical problem of heave and pitch motion of the wing is considered in this study for optimization of flutter speed. The structural parameter of mass, moment of inertia, and spring stiffness for heave and pitch motions is varied from -5 to $+5\%$ of baseline value. A set of 81 DoEs is prepared. A frequency domain-based approach is used to estimate the flutter speed on DoEs. A surrogate response surface is created with the help of Kriging interpolation method to estimate flutter speed at various points in the design space. The response surface for varying mass and moment of inertia and keeping other variable at baseline value shows little improvement over varying the spring stiffness value for heave and pitch motion. The flutter speed in the design space is optimized with the help of genetic algorithm. The flutter speed for the baseline case is 188.65 m/s, and the optimized value is 196.26 m/s with frequency domain approach. In order visualize the oscillations in time domain, the Simulink environment is utilized. The flutter speed for baseline case is 184.6 m/s. The Simulink model has shown little lower value on flutter speed as compared with frequency domain approach. The Simulink model predicted the flutter speed for the optimized

design condition which is 192.0 m/s. There is an increase in flutter speed of about 7.4 m/s from the baseline value from both the methods. This work demonstrates few techniques toward multidisciplinary optimization. Further extension of incorporating full aircraft will be much useful at early stages of aircraft design.

Acknowledgements The authors thank publication committee of Ramaiah University of Applied Sciences for allowing to publish this work.

References

1. X. Chen, G. Zha, M. Yang, Numerical simulations of 3-D wing flutter with fully coupled fluid–structural interaction. *Comput. Fluids* **36**, 856–867 (2007)
2. P.P. Shedambi, A.T. Sriram, R.R. Pathan, Numerical simulations of onset of flutter for AGARD 445.6 wing Planform. in *Proceedings of the 14th National Conference on Fluid Mechanical and Fluid Power* (NIT Hamirpur, India, 2013)
3. D. Ramdenee, A. Ilinca, I.S. Minea, Aeroelasticity of wind turbines blades using numerical simulation. *Adv. Wind Power INTECH* (2012). <https://doi.org/10.5772/52281>
4. J.R. Wright, J.E. Cooper, *Introduction to Aircraft Aeroelasticity and Loads* (Wiley, England, 2007)
5. M. Nikbay, P. ACAR Integrating analytical aeroelastic instability analysis into design optimization of aircraft wing structures. *J. App. Eng. Math.* V.1, N.2, 237 (2011)
6. T. Theodorsen, General theory of aerodynamics instability and the mechanism of flutter, NACA Report 496 (1934)
7. M. Karthik, A.T. Sriram, H.K. Narahari, Modeling and simulation studies on aeroelasticity of an airfoil with simulink. *SASTECH J.* **17**(1), 13–16 (2018)
8. D. Rana, S. Patel, A.K. Onkar, M. Manjuprasad, Time domain simulation of airfoil flutter using fluid structure coupling through fem based CFD solver. in *Symposium on Applied Aerodynamics and Design of Aerospace Vehicle (SAROD 2011)* (Bangalore, India, 2011)
9. S.N. Lophaven, H.B. Nielsen, J. Søndergaard, DACE A MATLAB Kriging Tool-box. *Technical Report IMM-TR-2002-12* (2002)
10. Matlab, Genetic algorithm guidelines at <https://in.mathworks.com/discovery/genetic-algorithm.html>, viewed on Dec 2018

Coupled Aero-structural Simulation Techniques Using High-Fidelity Analysis and Design Tools



Ravi Kumar Kovvali , Beerinder Singh, Srivatsan Lakshminarasimhan, and Sukumar Chakravarthy

Abstract An important component in the aerodynamic design and analysis of wing structures is the accurate modeling of the aeroelasticity in the problem. Along with the need for high-fidelity fluid and structural solvers, this multi-physics problem also requires a tight coupling strategy between the solvers and a robust methodology to handle the deformation of the CFD mesh. For partitioned solvers, it is also crucial that the coupling methodology caters for dissimilar meshes at the fluid–structure interface in order to fully leverage the capabilities of the individual solvers. In this paper, such a partitioned approach to modeling fluid–structure interaction (FSI) is demonstrated. Our methodology can accurately transfer loads and displacements at the wetted boundaries even when working with dissimilar meshes, such as a 3D CFD boundary interfacing with a finite element mesh comprising of 1D and 2D structural elements like beams and shells. The CFD mesh is deformed using a mesh morphing technology that only modifies the node locations while keeping the mesh connectivity unchanged. The effectiveness of our fluid structure interaction (FSI) methodology is demonstrated by conducting a static FSI analysis of the NASA Common Research Model (CRM) (modified with winglets) using CFD++ and CSM++ while maintaining a constant C_L . Our mesh morphing technology is also used to parameterize the mesh to conduct shape optimization studies of the winglet.

Keywords CFD · FSI · Winglet design · Optimization · Mesh morphing

R. K. Kovvali · B. Singh · S. Chakravarthy
Metacomp Technologies Inc., Agoura Hills, CA 91301, USA
e-mail: ravi@metacomptech.com

S. Lakshminarasimhan
Metacomp Technologies Pvt. Ltd, Chennai, India

1 Introduction

Airplane structures are required to be light, and consequently, the interplay of aerodynamics with structural dynamics is of paramount importance in their design and analysis. For this reason, with increasing computational power, high-fidelity fluid–structure interaction (FSI) analyses are becoming more and more important in the aircraft computer-aided design (CAD) industry. This paper is a demonstration of a partitioned approach to high-fidelity FSI applied to the design of winglets for drag reduction on an existing aircraft. For the purposes of this study, high fidelity refers to an analysis that can accurately capture skin friction drag in addition to the other flow physics.

FSI analyses of full aircraft configurations using hi-fidelity CFD tightly coupled with full FEA structural dynamics solvers are still not ubiquitous. The last AIAA CFD Drag Prediction Workshop (DPW) included a task for conducting coupled aero-structural simulations. Only three teams submitted solutions for this case with two teams using linear FEA and one using linear modal analysis [1]. These studies utilized the NASA Common Research Model (CRM) for demonstrating the drag prediction capabilities of the solvers [2, 3]. The authors were part of the only team from a commercial CFD company that showed a close agreement between the FSI analysis and the experimental wing deflections. This validated FSI methodology is the starting point for the winglet studies in this paper.

Winglets were first introduced by Whitcomb [4] who showed significant drag reduction over planar wingtip extensions using theory and wind tunnel tests. The wing itself was not modified in this study. Jones and Lasinsky [5] showed that if the wing shape was also optimized, wingtip extensions could provide induced drag reduction comparable to winglets. In this study, we demonstrate the optimization of a baseline winglet that is retrofit to an existing airplane wing such as the Boeing 737–800 [6]. In view of this, we restrict shape changes to only the winglet without changing the wing planform or cross section. In addition, we assume that a certain amount of span increase is allowed just like the B737-800, which has a larger span than the baseline aircraft.

Multidisciplinary Design Analysis and Optimization (MDAO) studies of aircraft wings or wingtips/winglets have traditionally utilized low-fidelity aerodynamics such as vortex lattice codes coupled with FE models of the structure [7]. Ning and Kroo [8] used low-fidelity aerodynamic models and stall speed constraints to optimize drag and weight for both retrofit and new wing designs. They showed that winglets slightly outperform wing tip extensions in maneuvering conditions and vice versa in cruise flight. Jansen et al. [9] showed that when a spanwise constraint is present, winglets provide the optimal performance.

In recent years, high-fidelity analyses with CFD codes are becoming more commonplace for MDAO of wings and winglets. Takenaka et al. [10] used Euler CFD in conjunction with an FE analysis of the structure to optimize winglet root chord, taper ratio, length, cant angle, and toe angle with a Kriging model. They also validated the drag reduction with wind tunnel tests. Kenway and Martins [11] studied

the wing planform optimization of the NASA CRM to reduce Max Take-Off Weight (MTOW) and fuel burn. The wing shape was optimized with hundreds of design variables using fast gradient calculations via the adjoint method. They did not include a winglet or any span constraint and showed fuel burn reductions of 6.6% for the MTOW objective and 11.2% for the cruise fuel burn objective. Khosravi and Zingg [12] used a similar procedure to optimize a wing and winglet in cruise and maneuver conditions. They found that a downward pointing winglet reduces the induced drag by a larger amount since the effective span is larger in the deformed state. Hwang and Martins [13] used adjoint-based optimization with thousands of design variables to minimize fuel burn over the entire mission profile including route allocation in order to maximize airline profits. They showed a 27% increase in profits going from a route allocation only optimization to an allocation and mission design optimization. However, the above studies ran the CFD codes in Euler mode to reduce computational time because of the large number of simulations required. For this reason, these studies utilize viscous drag estimation techniques to predict the drag which may bring in errors due to inaccurate representation of skin friction. There are optimization studies that use Reynolds averaged Navier–Stokes (RANS) solvers, but these typically do not include FSI [14]. Also, since these studies on wings include optimization of the wing shape using a large number of design variables they may be more suited for conceptual design as opposed to retrofitting.

Optimization methodologies used in these MDAO studies range from Design of Experiments (DoE) to Genetic Algorithms with surrogate models and full gradient-based optimization with a large number of design variables. In this paper, we use a solve-to-wall CFD grid in an FSI analysis to study various winglet design parameters with a DoE approach. We use a general-purpose CFD solver (CFD++) with a pitch trim procedure coupled to a structural solver (CSM++) to optimize a retrofit winglet design for cruise flight conditions. We also restrict this study to an upward pointing winglet since this is the most common type of winglet seen on production aircraft. Even though a downward winglet may produce lower induced drag, it may be ruled out because of other considerations such as a tip strike on landing.

2 Methodology

2.1 Computational Fluid Dynamics (CFD) Solver

CFD++ is a general-purpose, unstructured computational fluid dynamics solver in widespread commercial use around the world. The combination of modern numerical methods and modern turbulence closures enables increasingly reliable prediction of both internal and external, steady and unsteady aerodynamic flow problems across the Mach number range. In CFD++, structured, unstructured, hybrid, and complex overset grids can be handled within a single unified framework. Further flexibility and computational power are realized through the use of massively parallel computers.

Both these attributes are crucial for successful prediction of real-life flow problems of engineering interest.

2.2 Computational Structural Mechanics (CSM) Solver

The general-purpose structural dynamics solver CSM++ is designed around existing finite element analysis (FEA) libraries with the ultimate goal of FSI in mind from the outset. It can be used for static or dynamic structural analysis including large deformations and material nonlinearity. A variety of specialized elements such as beams, shells, rigid body elements (RBEs), etc., are available in addition to various topologies of solid elements. Both corotational and large strain formulation elements are available for large deformation analyses. In addition, thermal stress computations can be carried out including temperature-dependent material properties. Standard FE models from other commercial software can also be easily imported into CSM++.

2.3 MetaFSI—Coupling Tools for Fluid–Structure Interaction

Two important capabilities are required for a partitioned approach to FSI: (1) a methodology to transfer pressure/stresses from the CFD boundaries to the structure boundaries and (2) a technique to propagate the structural displacements from the structure into the fluid volume grid. For a partitioned approach, both of these capabilities must cater for vastly different fluid and structure topologies. For example, the best structural representation for a wing may be a beam model, especially in the preliminary design stage, but the CFD representation must have the full 3D wing surface.

MetaFSI is a suite of tools to facilitate FSI analysis using CFD++ and a structural solver such as CSM++. These tools enable energy-conserving load transfer from the CFD grid to a topologically matching or non-matching CSD grid. Loads are transferred to the closest point of the structural grid keeping in mind that this may be in the middle of an element.

A diagram showing the way CFD++, CSM++ and MetaFSI interact with one another is shown in Fig. 1. Once the structural deformations are available after application of fluid loads, these deformations need to be propagated into the CFD volume grid. This is achieved by using a 3D interpolation based on radial basis functions (RBFs). As shown in Fig. 2a, the interpolant is built using a cloud of points, some of which define zero-displacements (anchor points) and some define the displacement received from the CSM solver (moving points). Since the CFD grid points are moved without changing the cell connectivity, we refer to this process as morphing.

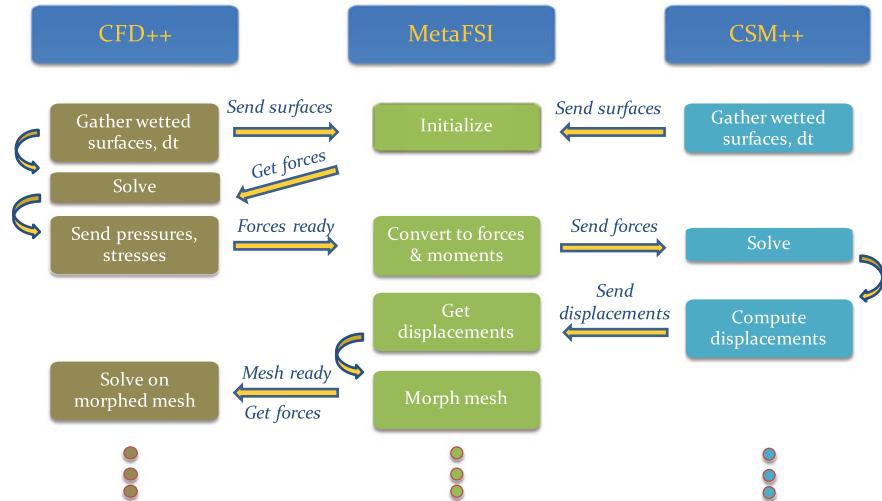


Fig. 1 Flowchart depicting FSI methodology using mesh morphing

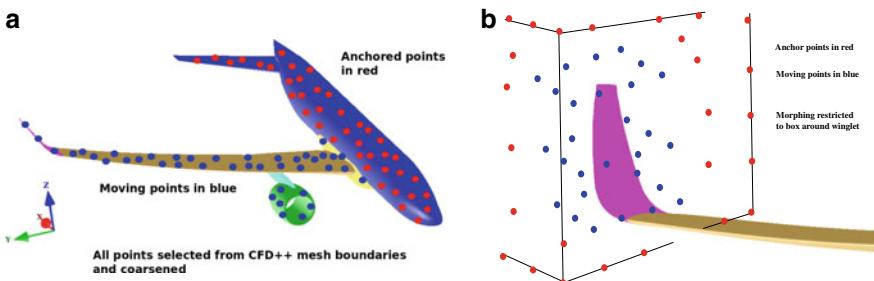


Fig. 2 Illustration of RBF control points for: **a** FSI morphing on the CRM geometry and **b** parameterized winglet morphing

An added benefit of the morphing technique is that it can also be used to morph the grid for optimization studies without having to revert to the CAD software to generate separate geometries. Figure 2b shows the way in which this is applied to the winglet design problem.

3 Computational Model

The computational model used in this study is based on the NASA Common Research Model (CRM) [2]. This is based on a generic airliner as shown in Fig. 2. Geometry and FEM files for several variants are available on the NASA Web site. We have used

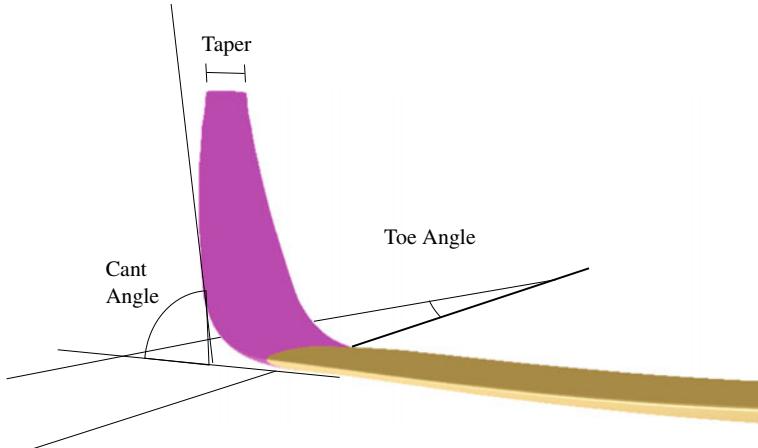


Fig. 3 Parameterization for winglet design

the variant that includes the engine nacelle and pylon since this is the most general case. We modified the geometry to include a generic winglet at the tip. This winglet, dubbed configuration 0, was measured to have a cant angle of 68.1° , a toe angle of -3.5° (toe-in), and a taper ratio of 0.29. Figure 3 shows the baseline winglet with the parameterization used to vary the winglet design.

A solve-to-wall grid was generated on the baseline configuration with about 31 million cells, and this grid was used for all analyses in this paper. Analysis for each design configuration consists of a rigid, steady-state CFD++ run including a C_L driver to adjust the angle of attack (AoA) for $C_L = 0.5$, followed by a coupled, static FSI run also including the C_L driver. After each CSM++ computation and grid morphing cycle, CFD++ was run 60 steps to allow for convergence and the C_L driver intervals were set such that AoA changes occurred in between the FSI coupling steps. A convergence tolerance of $1.0\text{e-}04$ for the wing deflections at steady state was used as the termination criterion. All the CFD++ simulations were run using a 2-equation realizable $k-\varepsilon$ turbulence model with the compressible RANS equations.

For the finite element model, a NASTRAN bulk data file of the CRM wingbox model provided by NASA was imported into CSM++. As provided, the model consists of spars, ribs, and skin modeled using shells, beams, rigid elements, and point masses. This was modified slightly within CSM++ by appending a rigid beam at the wing tip to model the winglet.

4 Results

Table 1 shows the C_D values predicted for the various design configurations sorted based on the elastic C_D . Configurations 1 and 2 vary only the cant angle to 60°

Table 1 Winglet design results (ordered by increasing C_D elastic)

Config.#	Cant angle	Toe angle	Taper ratio	C_D rigid ($\times 10^{-2}$)	C_D elastic ($\times 10^{-2}$)	AoA rigid	AoA elastic	$\Delta C_D (\times 10^{-4})$
1—"136"	60.0°	0.0°	0.35	2.45517	2.47469	2.13°	2.36°	-4.515
2—"13"	60.0°	0.0°	0.29	2.45472	2.48336	2.13°	2.36°	-3.648
3—"135"	60.0°	0.0°	0.25	2.45472	2.48341	2.13°	2.36°	-3.643
4—"35"	68.1°	0.0°	0.25	2.45654	2.48517	2.14°	2.36°	-3.467
5—"3"	68.1°	0.0°	0.29	2.45682	2.48519	2.14°	2.36°	-3.464
6—"36"	68.1°	0.0°	0.35	2.45717	2.48529	2.14°	2.36°	-3.454
7—"23"	75.0°	0.0°	0.29	2.46182	2.48949	2.14°	2.36°	-3.035
8—"235"	75.0°	0.0°	0.25	2.46172	2.48954	2.14°	2.36°	-3.030
9—"236"	75.0°	0.0°	0.35	2.46172	2.48958	2.14°	2.36°	-3.026
10—"16"	60.0°	-3.5°	0.35	2.46093	2.49926	2.15°	2.37°	-2.058
11—"1"	60.0°	-3.5°	0.29	2.46216	2.50026	2.15°	2.37°	-1.957
12—"15"	60.0°	-3.5°	0.25	2.46305	2.50108	2.15°	2.37°	-1.875
13—"6"	68.1°	-3.5°	0.35	2.46588	2.50332	2.15°	2.37°	-1.651
14—"0" (baseline)	68.1°	-3.5°	0.29	2.46720	2.50440	2.16°	2.37°	-1.544
15—"5"	68.1°	-3.5°	0.25	2.46803	2.50529	2.16°	2.37°	-1.454
16—"26"	75.0°	-3.5°	0.35	2.47142	2.50736	2.16°	2.37°	-1.247
17—"2"	75.0°	-3.5°	0.29	2.47260	2.50849	2.16°	2.37°	-1.134
18—"25"	75.0°	-3.5°	0.25	2.47349	2.50932	2.16°	2.37°	-1.051
19—"14"	60.0°	3.5°	0.29	2.51140	2.51087	2.11°	2.35°	-0.897
20—"146"	60.0°	3.5°	0.35	2.51140	2.51182	2.11°	2.35°	-0.802
21—"145"	60.0°	3.5°	0.25	2.51257	2.51228	2.12°	2.35°	-0.756
22—"46"	68.1°	3.5°	0.35	2.51751	2.51900	2.12°	2.35°	-0.083
23—"4"	68.1°	3.5°	0.29	2.51973	2.52087	2.12°	2.35°	+0.103
24—"45"	68.1°	3.5°	0.25	2.52086	2.52268	2.12°	2.35°	+0.284
25—"246"	75.0°	3.5°	0.35	2.52458	2.52693	2.13°	2.35°	+0.709
26—"24"	75.0°	3.5°	0.29	2.52946	2.53114	2.13°	2.35°	+1.130
27—"245"	75.0°	3.5°	0.25	2.53355	2.53503	2.13°	2.35°	+1.519
No winglet (CRM)	-	-	-	2.51652	2.51984	2.16°	2.34°	-

and 75°, respectively, configurations 3 and 4 vary only the toe angle to 0° and +3.5°, respectively, and finally configurations 5 and 6 vary only the taper ratio to 0.25 and 0.35, respectively. Two- and three-digit identifiers signify a combination of the single-digit configurations described above. In all, 27 different combinations of the winglet geometry are considered. The drag coefficients and the respective angles of attack for the rigid and elastic runs are also shown. The configuration labeled 136 provides a reduction in C_D by nearly 3 drag counts (a reduction of 1.19%) over the

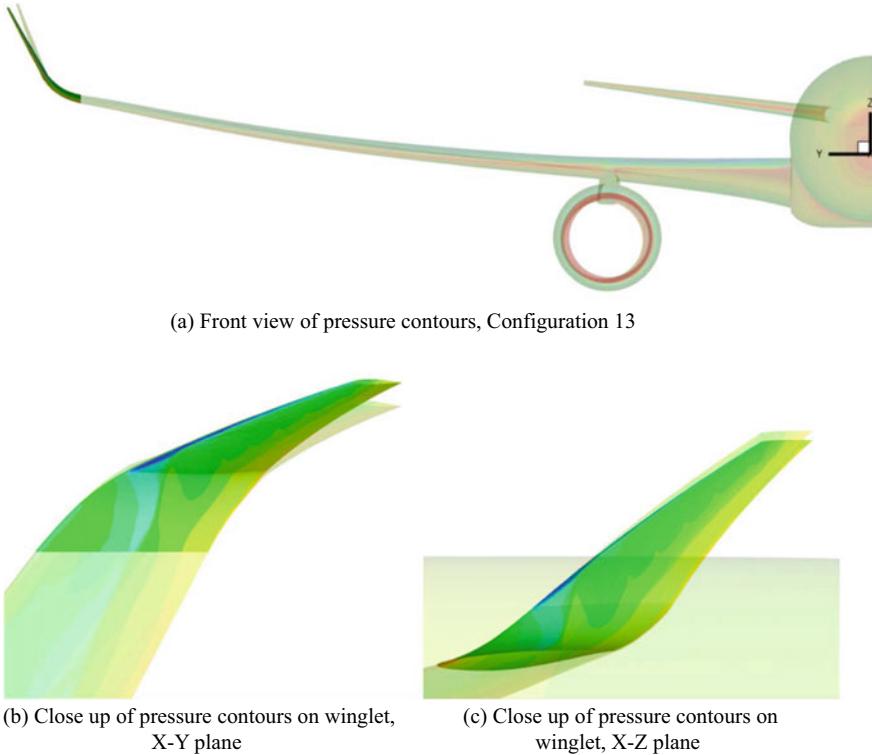


Fig. 4 Pressure contours for winglet configuration 136 (opaque) and baseline configuration (translucent)

baseline winglet. In all these configurations, the computed angle of attack to achieve a C_D of 0.5 is fairly constant. Contour plots of the pressure distribution on the winglet from this configuration are presented in Fig. 4, overlaid on corresponding results from the analysis of the baseline configuration.

Finally, results from a configuration without a winglet are also included in Table 1 to help highlight the benefits of adding a winglet. Looking at the reduction in drag achieved, it is clear that adding a winglet to the Common Research Model with a geometry described by configuration 136 achieves a notable reduction in C_D of over 4.5 drag counts (a reduction of 1.79%) that can potentially translate to meaningful improvements in, say, fuel consumption or payload capacity. It can also be seen that certain configurations (mostly those with outward toe angles) can be very poor by resulting in an increase in overall drag, up to 1.5 drag counts, compared to having no winglet at all. This would happen when the reduction in induced drag is unable to offset the increase in viscous drag caused by the mere presence of a winglet.

A plot of the reduction in drag, expressed in drag counts, is shown in Fig. 5 as it varies with the configuration numbers listed in Table 1. Qualitatively, we can first see that the consideration of aeroelastic effects results in a noticeably smoother trend

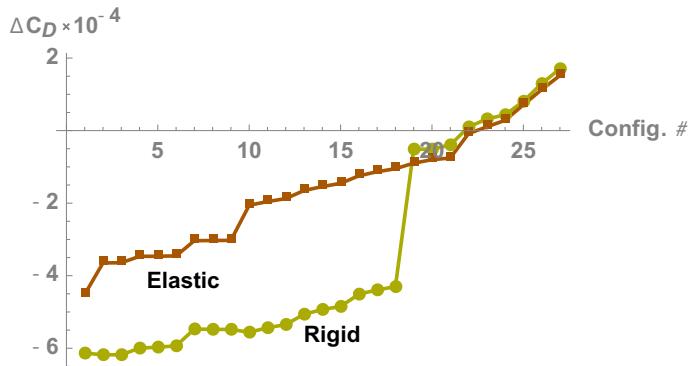


Fig. 5 Comparison of predicted $\Delta C_D (\times 10^{-4})$, with and without modeling aeroelastic effects

observed regarding the effect of the various geometric parameters of the winglet. This is as opposed to the analyses that ignored aeroelastic effects where, say, a drastic jump of 4.5 drag stops is predicted when going to configurations with a toe out angle of 3.5° . Quantitatively, the rigid wing results drastically overestimate the benefits of the addition of a winglet—by up to 70% for certain configurations—making it all the more important to include aeroelastic effects to accurately gauge the benefits of adding a specific winglet configuration.

It is worth noting that while the goal of this exercise is to be able to quickly identify the significant geometric parameters of a winglet that help in reducing the total drag, it may not always be possible to vary certain parameters by much. For example, a retrofitted winglet design might have constraints on the maximum allowable wingspan that can be practically accommodated which places a restriction on the lowest cant angle that can be allowed, despite a lower angle typically resulting in a more effective design as is evident from the results.

References

1. S. Keye, D. Mavriplis, Summary of case 5 from sixth drag prediction workshop: coupled aerostructural simulation. *J. Aerogr.* **55**(4), 1380–1387 (2018). <https://doi.org/10.2514/1.C034427>
2. J.C. Vassberg, M.A. DeHaan, S.M. Rivers, R. Wahls, A development of a common research model for applied cfd validation studies. in *26th AIAA Applied Aerodynamics Conference* (Honolulu, Hawaii, 2008). <https://doi.org/10.2514/6.2008-6919>
3. S. Keye, R. Rudnik (2015) Validation of wing deformation simulations for the NASA CRM model using fluid structure interaction computations. in *53rd AIAA Aerospace Sciences Meeting*. (Kissimmee, Florida, 2015). <https://doi.org/10.2514/6.2015-0619>
4. R. Whitcomb, A design approach and selected wind tunnel results at high subsonic speeds for wing-tip mounted winglets. *NASA TND-8260* (1976)
5. R.T. Jones, T.A. Lasinsky (1980) Effects of winglets on the induced drag of ideal wing shapes. *NASA TM-81230* (1980)

6. Faye R., Laprete R., Winter M., Blended winglets for improved airplane performance. Boeing Commercial Airplanes AERO Magazine, 17 (2002). http://www.boeing.com/commercial/aeromagazine/aero_17/winglets.pdf
7. B. Grossman, R.T. Haftka, P.J. Kao, D.M. Polen, M. Rais-Rohani, Integrated aerodynamic-structural design of a transport wing. *J. Aircr.* **27**(12), 1050–1056 (1990). <https://doi.org/10.2514/3.45980>
8. S.A. Ning, I. Kroo, Multidisciplinary considerations in the design of wings and wing tip devices. *J. Aircr.* **47**(2), 534–554 (2010). <https://doi.org/10.2514/1.41833>
9. P. Jansen, R.E. Perez, J.R.R.A. Martins, Aerostructural optimization of nonplanar lifting surfaces. *J. Aircr.* **47**(5), 1490–1503 (2010). <https://doi.org/10.2514/1.44727>
10. K. Takenaka, K. Hatanaka, W. Yamazaki, K. Nakahashi, Multidisciplinary design exploration for winglet. *J. Aircr.* **45**(5), 1601–1611 (2008). <https://doi.org/10.2514/1.33031>
11. G.K.W. Kenway, J.R.R.A. Martins, Multipoint high-fidelity aerostructural optimization of a transport aircraft configuration. *J. Aircr.* **51**(1), 144–160 (2014). <https://doi.org/10.2514/1.C032150>
12. S. Khosravi, D.W. Zingg, Aerostructural perspective on winglets. *J. Aircr.* **54**(3), 1121–1138 (2017). <https://doi.org/10.2514/1.C033914>
13. J.T. Hwang, J.R.R.A. Martins, Allocation-mission-design optimization of next-generation aircraft using a parallel computational framework. in *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference* (San Diego, California, 2016). <https://doi.org/10.2514/6.2016-1662>
14. Z. Lyu, J.R.R.A. Martins, Aerodynamic shape optimization of a blended-wing-body aircraft. in *21st AIAA Computational Fluid Dynamics Conference*, (San Diego, California, 2013). <https://doi.org/10.2514/6.2013-283>

Optimization of B-Spline Launch Vehicle Payload Fairing



K. Nihal and Pankaj Priyadarshi

Abstract Payload fairing (PLF) provides protection for the payload carried by a launch vehicle from external flow and aerodynamic heating. Not only the PLF should be able to accommodate the required payload volume inside it, it should have desirable aerodynamic features like low drag, low transonic shock strength, low base bending moment, etc. The PLF shape design problem is posed as a multiobjective optimization problem with minimization of drag coefficient and maximization of PLF volume. A fairly general parameterization of the PLF shape has been constructed by using B-Splines. The shape parameters constitute the design variables. The control points of the B-Splines formed the design variables. The design space also included the Mach number in addition to the control point co-ordinates. The design space was sampled using Random Latin Hypercube Sampling (RLHS). For the study, 160 PLF designs were constructed and CFD simulations were carried out. CFD studies were carried out to evaluate the aerodynamics of these configurations. Validation studies were done for a particular configuration at various Mach numbers. An open-source mesh generation software, Gmsh, was used along with the open-source CFD solver, SU2, and the Paraview flow visualization software for postprocessing. Grid independence, domain independence, scheme independence, and turbulence model independence studies were performed for a reference geometry before getting into the detailed CFD simulations. Axisymmetric Reynolds Averaged Navier Stokes (RANS) CFD simulations were carried out using SST turbulence model. Second-order spatial discretization was employed and the ROE solver with Venkatakrishnan Limiter was used. An ANN surrogate model was constructed to estimate the drag coefficient as a function of the design variables. The optimization studies brought out that the conical configurations was optimal for lower volumes at supersonic Mach numbers, whereas

VSSC Trivandrum.

K. Nihal

Department of Aerospace Engineering, Indian Institute of Space Science and Technology,
Valiyamala, Trivandrum, Kerala, India

e-mail: nihalsuresh@gmail.com

P. Priyadarshi

ACDD, VSSC, ISRO, Trivandrum, Kerala, India

e-mail: pankaj.priyadarshi@gmail.com

© Springer Nature Singapore Pte Ltd. 2020

R. R. Salagame et al. (eds.), *Advances in Multidisciplinary Analysis and Optimization*, Lecture Notes in Mechanical Engineering,
https://doi.org/10.1007/978-981-15-5432-2_8

blunt configuration was optimal for all cases of transonic regime for the objectives considered in the present study.

Keywords Drag coefficient · Optimization · Design space · RLHS · CFD

1 Introduction

Launch vehicles are used to carry payloads to space. Payloads are protected from the external flow by payload fairing (PLF). The design of the PLF should take care of minimizing the inflight noise levels in the transonic regimes and minimize drag and weight. As the external profile of nose cone of most existing configurations is fairly simple consisting of blunt cap followed by either cones or flares (for conical nose cone configurations) or segments of a circle (for ogive configurations), these parameterizations allow rather restricted shapes. Hence, arises the need to explore the use of B-spline payload fairing.

The present study tries to obtain the optimal configurations in terms of minimizing drag and maximizing volume.

2 Methodology

2.1 B-Splines for Configuration Parameterization

A curve made of one polynomial gets usually restricted in various forms. Bézier curve is a perfect example wherein the degree gets fixed with the number of control points. Any alteration in these points would lead to a change in the entire curve. Also, complex shapes are difficult to be fitted with low degree curves [1]. Hence, arises the importance of B-Splines which overcomes all these shortcomings. Let \mathbf{U} be a knot vector which is a nondecreasing sequence of real numbers called knots. The knot values are assigned as follows [1]:

$$u_i = \begin{cases} 0 & i < k \\ i - k + 1 & k \leq i \leq n \\ n - k + 2 & i > n \end{cases} \quad (1)$$

The i th basis function of degree j can be given by:

$$N_{i,j}(u) = \frac{u - u_i}{u_{i+j} - u_i} N_{i,j-1}(u) + \frac{u_{i+j+1} - u}{u_{i+j+1} - u_{i+1}} N_{i+1,j-1}(u) \quad (2)$$

$$N_{i,0}(u) = \begin{cases} 1 & u_i \leq u \leq u_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

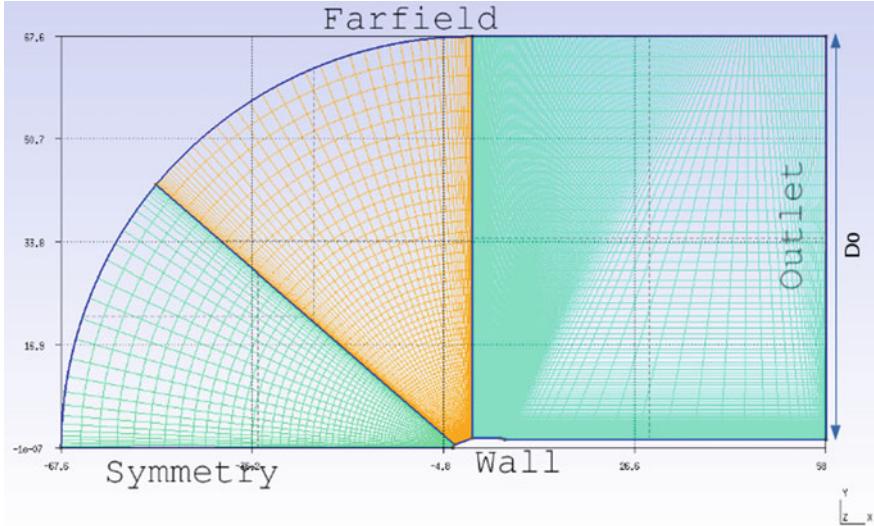


Fig. 1 Mesh generated for reference configuration

Then the B-spline curve of degree k is given by

$$\mathbf{C}(u) = \sum_{i=0}^n N_{i,k}(u) P_i \quad 0 \leq u \leq n - k + 2 \quad (4)$$

A few properties of B-splines are that there will be a total of $n - k + 2$ segments in the curve, where in each segment of the curve is influenced by only k control points. Here k defines the continuity of the curve. Curve will be C_{k-2} continuous.

2.2 Mesh Generation

The mesh generation for the problem was done in the open-source software Gmsh. A configuration of payload fairing with inter-stage was meshed with four boundary markers, namely *Wall*, *Symmetry*, *Farfield* and *Outlet* (Fig. 1). A grid independence test was also conducted on this reference configuration.

2.3 Governing Equations

The Navier–Stokes equations describing the behaviour of a viscous fluid is given by [2].

$$\text{Continuity Equation : } \frac{\partial}{\partial t} \int_{\Omega} \rho \, d\Omega + \oint_{\partial\Omega} \rho (\mathbf{v} \cdot \mathbf{n}) \, dS = 0 \quad (5)$$

$$\text{Momentum Equation : } \frac{\partial}{\partial t} \int_{\Omega} \rho \mathbf{v} \, d\Omega + \oint_{\partial\Omega} \rho \mathbf{v} (\mathbf{v} \cdot \mathbf{n}) \, dS = \int_{\Omega} \rho \mathbf{f}_e \, d\Omega - \oint_{\partial\Omega} p \mathbf{n} \, dS + \oint_{\partial\Omega} (\bar{\tau} \cdot \mathbf{n}) \, dS \quad (6)$$

$$\text{Energy Equation : } \frac{\partial}{\partial t} \int_{\Omega} \rho E \, d\Omega + \oint_{\partial\Omega} \rho H (\mathbf{v} \cdot \mathbf{n}) \, dS = \oint_{\partial\Omega} k (\nabla T \cdot \mathbf{n}) \, dS + \int_{\Omega} (\rho \mathbf{f}_e \cdot \mathbf{v} + q_h) \, d\Omega + \oint_{\partial\Omega} (\bar{\tau} \cdot \mathbf{v}) \cdot \mathbf{n} \, dS \quad (7)$$

where,

Ω : Control volume of finite element, $\partial\Omega$: Closed surface, ρ : Density, \mathbf{n} : Outward pointing unit normal vector, dS : Surface element, k : Thermal diffusivity coefficient, \mathbf{f}_e : Body force per unit volume, $\bar{\tau}$: Viscous stress tensor, E : Total energy per unit volume, H : Total enthalpy, T : Absolute static temperature, q_h : Heat transfer per unit mass.

2.4 CFD Solver

CFD simulations were carried out using ROE, AUSM and JST numerical schemes. The solver used in the current study is the open-source software named SU2 [3]. Both Euler and RANS simulations were carried out. The reference length for coefficient of moment is taken to be as that of cylinder length, whereas the reference area as the cross-section area of the same. Axisymmetric runs were performed as the results were required for zero degree angle of attack.

The viscous cases were run with a zero heatflux boundary condition as well as isothermal boundary condition with a constant temperature of 300 K on the wall. Apart from these, the symmetry boundary condition was used at $y = 0$ and farfield boundary condition on the rest of boundaries. For supersonic flows as the outer boundary characteristics come from inside, a supersonic outlet boundary is given for supersonic flow regime. Second-order scheme with Venkatakrishnan limiter was used. CFL number ranging from 2 to 10 was used depending on the convergence trend. Convergence of the solutions was studied based on the C_D value. The postprocessing was done in Paraview [4].

2.5 Solution Quality and Validation Studies

A series of CFD simulations were conducted on a reference geometry (Fig. 2a) to study independence with respect to *scheme, grid, domain and turbulence* model being

used. The simulations were conducted for free stream conditions, $P_\infty = 101,325.0$ Pa, $T_\infty = 288.15$ K and $\rho_\infty = 1.2252$ kg/m³ for $\alpha = 0^\circ$.

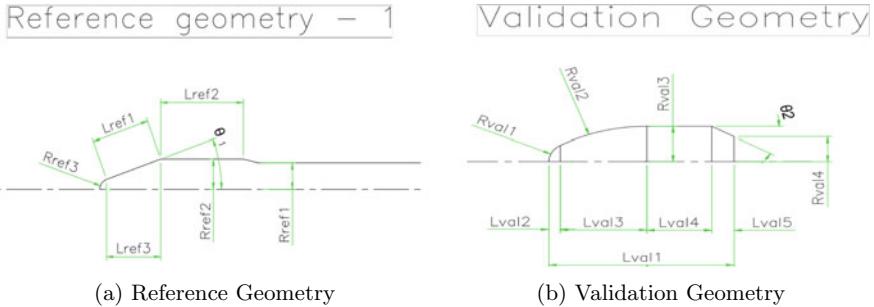


Fig. 2 The geometries used for studies

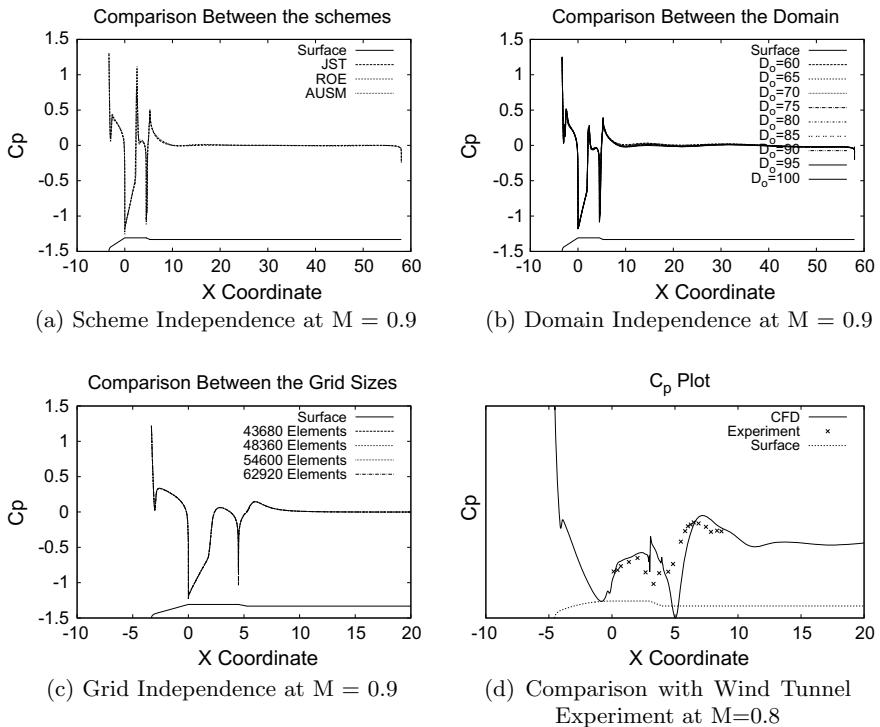


Fig. 3 C_p plots

2.5.1 Scheme Independence

Results of the CFD simulation at $M = 0.9$, on a mesh with a grid size of 51,000 elements for various schemes are shown in Fig. 3a. The C_p variation was seen to be within ± 0.05 between the three schemes, namely JST, ROE and AUSM.

2.5.2 Domain Independence

The domain defined by the size of the outlet connector as in Fig. 1 was varied from 60 to 100 in intervals of 5 and the corresponding C_p plots were plotted which was seen to be within ± 0.005 (Fig. 3b).

2.5.3 Grid Independence

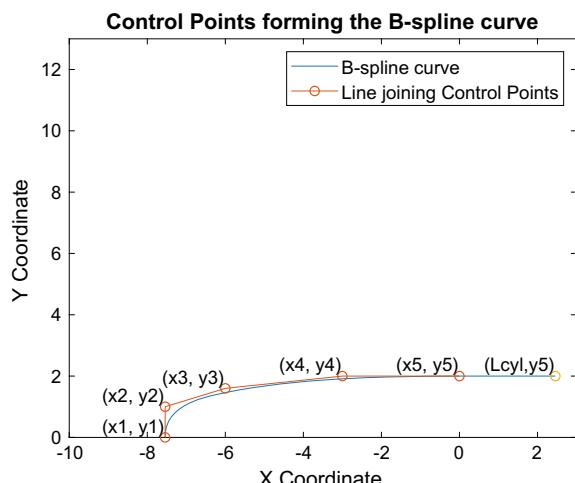
Among four grid sizes used, insignificant variation in C_p was observed as seen in Fig. 3c.

2.5.4 Validation Studies

A validation study was done on a double ogive configuration (Fig. 2b) at a Mach number of 0.8 with SST turbulence model and rest of the conditions same as given above.

The simulation results were validated with available wind tunnel test results. An error of less than ± 0.05 is observed from Fig. 3d, which is within tolerable limits as

Fig. 4 Plot showing the control points forming the B-spline curve



the measurement uncertainty is of the same order. Deviation in the boat tail region is larger, which points to poor capture of the flow physics by CFD simulations in this region.

2.6 Geometric Parametrization

The variables associated with the design are the x and y coordinates of the control points of the B-spline curve, length of the cylinder and the Mach number making it a total of twelve variables as follows.

$(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), L_{\text{cyl}}$, and M . The control points making up the B-spline are marked in Fig. 4.

The abscissa of fifth point and the ordinate of first point are fixed to zero. To ensure smooth transition from B-spline curve to cylinder, the ordinate of the fourth point should be same as of the fifth. The same applies to the abscissa of first and second point, where the configuration should be blunt from aerodynamic heating considerations. Hence,

$$x_5 = y_1 = 0 \quad (8)$$

$$y_4 = y_5 \quad (9)$$

$$x_1 = x_2 \quad (10)$$

Therefore, the variables reduce to eight in number. We define L' as the total length of the PLF, $L' = (L_{\text{cyl}} + |x_1|)$. Apart from the variables L' and y_5 , the rest of the coordinate points are represented as fractions of these two, which are given as

$$fx_1 = \frac{-x_1}{L'} \quad (11)$$

$$fx_3 = \frac{x_3}{x_1} \quad (12)$$

$$fx_4 = \frac{x_4}{x_3} \quad (13)$$

$$fy_2 = \frac{y_2}{y_5} \quad (14)$$

The y coordinate of a point on line joining (x_2, y_2) and (x_4, y_4) at $x = x_3$ is given by

$$y_{\text{inter}} = y_2 + (x_3 - x_2) \times \frac{y_4 - y_2}{x_4 - x_2} \quad (15)$$

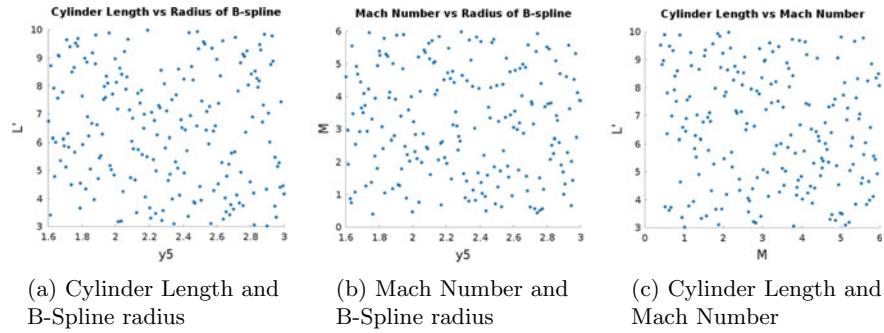


Fig. 5 Plot showing scatter plot of various sampled data

Now to maintain y_3 as part of convex hull, the coordinate should lie between the lines $y = y_5$ and $y = y_{\text{inter}}$. Hence

$$f y_3 = \frac{y_3 - y_{\text{inter}}}{y_5 - y_{\text{inter}}} \quad (16)$$

2.7 Design of Experiments

Random Latin Hypercube Sampling (RLHS) was used to ensure the design space was sampled with least correlation. A few of them are plotted and shown in Fig. 5. A total of 186 samples were drawn in the 8-D design space.

2.8 Surrogate Modelling

An artificial neural network model was trained using MATLAB with one hidden layer consisting of five sigmoid neurons and one linear output neuron to form a surrogate model out of 160 data points. The number was chosen based on the accuracy of the model formed. The ANN-based surrogate model calculates the coefficient of drag at fraction of time as compared to CFD simulation, making the whole optimization process faster.

Figure 6 shows a comparison of the training as well as testing data with the C_D predicted by the surrogate model for corresponding data points, labelled as data number. It was seen that the RMS error was 14%. The forebody C_D for all sampled configurations was obtained after integration of pressure drag as well as skin friction drag upto the start of the boat tail.

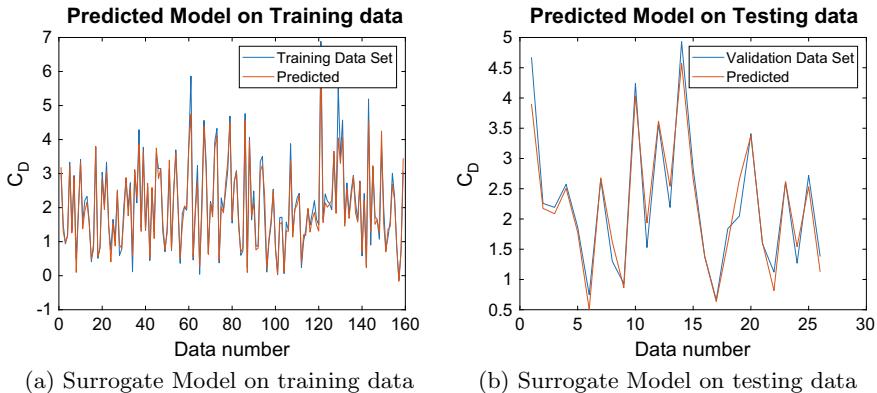


Fig. 6 Surrogate model

2.9 Optimization Problem

2.9.1 Objectives

Many objectives can be considered for this problem, e.g. maximizing the volume, minimizing the drag, minimizing the shock strength at transonic Mach numbers, minimizing the bending moment due to the PLF, minimizing the mass of the PLF, etc. However, in the present study, the following two objectives are considered:

$$\text{Minimize } C_D \text{ and} \quad (17)$$

$$\text{Maximize Volume} \quad (18)$$

2.9.2 Constraints

There are a few geometric constraints, which have been removed by intelligently defining appropriate geometric parameterization. This is described in Sect. 2.6.

2.9.3 Design Variables

The design variables for this problem are primarily geometric in nature. They are described in Sect. 2.6. The ranges of these design variables are given below:

$$3 < L' < 10 \quad (19)$$

$$1.6 < y5 < 3 \quad (20)$$

$$0.1 < fx1 < 0.95 \quad (21)$$

$$0.01 < fx3 < 0.99 \quad (22)$$

$$0.01 < fx4 < 0.99 \quad (23)$$

$$0.01 < fy2 < 0.99 \quad (24)$$

$$0 < fy3 < 0.99 \quad (25)$$

$$0.4 < M < 6 \quad (26)$$

2.9.4 Optimization Algorithm

The optimization problem was solved using the fmincon function of MATLAB by employing multiple initial points. The multistart option of MATLAB runs the optimisation program while attempting to find multiple local optimum solutions by starting from various initial points in the design space to target a globally optimum solution. In order to get the optimal solutions for this multiobjective problem, ϵ constraint method was employed and the optimization problem was posed as follows:

$$\text{minimize } C_D \quad (27)$$

such that

$$\text{Volume} \geq V_i \quad (28)$$

where $V_i \in \{50, 55, \dots, 95, 100\}\text{m}^3$ and under the design variable bounds given above.

3 Results & Discussion

Three typical configurations were taken from the DOE as shown in Fig. 7. The reference configuration is also plotted along with it. The plot of C_D with Mach numbers for these configurations (Fig. 8) indicates a significantly lower dras coefficient for

the configuration 1 design especially in the transonic region. C_P distributions were plotted for each configurations and the reference geometry for Mach numbers 0.95 and 2.0 (Fig. 9a, b). The flow characteristics on configuration 1 for the same Mach numbers are depicted in Fig. 9c, d.

Optimization studies were carried out for $M = 0.4\text{--}6.0$. Figure 10 shows the results of multiobjective optimization to minimize C_D and maximize volume. It is seen from the Pareto optimal front that the two objectives are conflicting. That is, higher volume results in higher C_D . It is seen that the drag coefficient at lower Mach numbers is very small and can even become -ve due to -ve C_p on most of the forebody region. Only forebody drag coefficient is considered up to the boat-tail start and the base drag on the boat tail is not accounted.

Figure 11a-d shows optimal configurations for four typical Mach numbers for different volumes. It is seen that all optimal configurations have longest permissible length, primarily to maximize the volume and reduce pressure drag. An interesting observation is that the optimal configurations of transonic Mach numbers are blunter as compared to the optimal supersonic and subsonic configurations. Conical configurations came out to be the optimal ones for lower volumes for supersonic Mach numbers, as here attached oblique shocks are formed which are weaker than the detached shocks. This reduces the pressure drag.

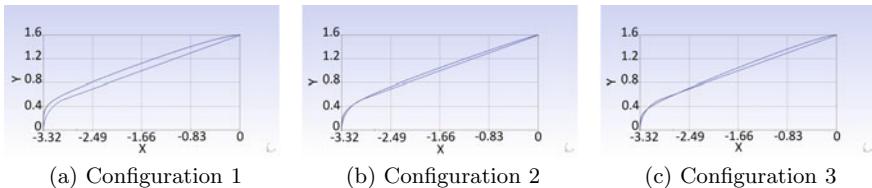


Fig. 7 Typical configurations plotted along with the reference configuration

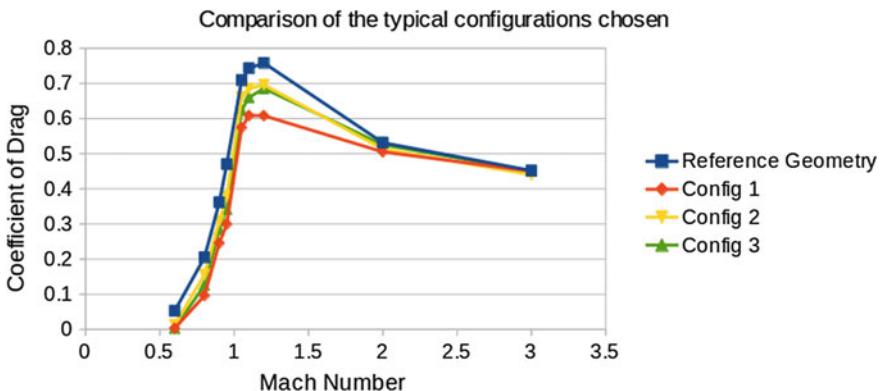


Fig. 8 C_D versus Mach Number for the typical configurations

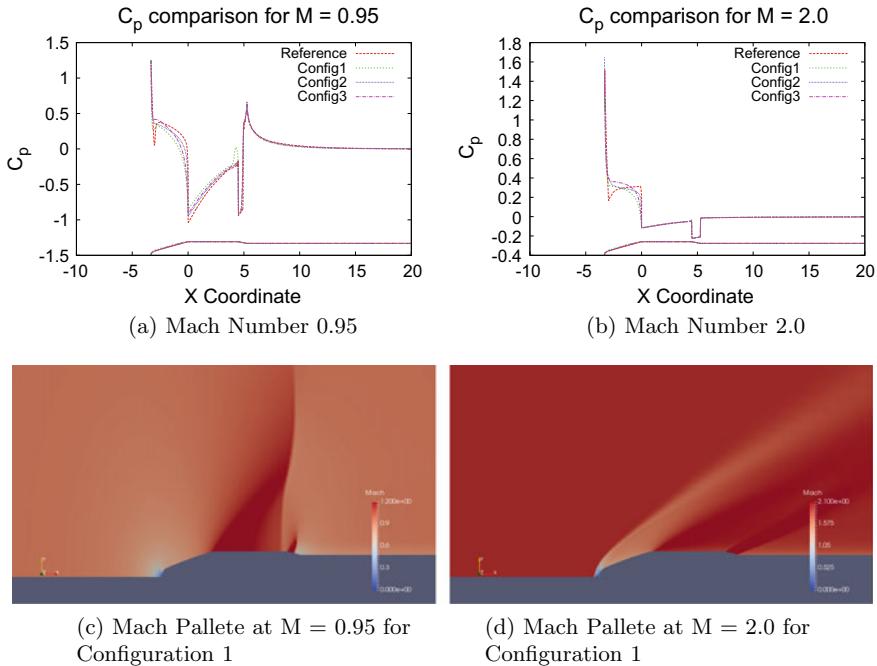


Fig. 9 C_p comparison plots for the three configurations along the Mach number palletes for Configuration 1

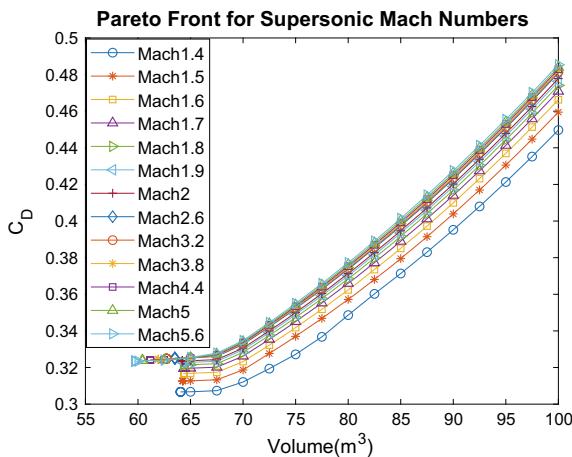


Fig. 10 Results of the multiobjective optimization studies to minimize forebody C_D and maximize PLF volume. Pareto optimal fronts are shown for different Mach Numbers

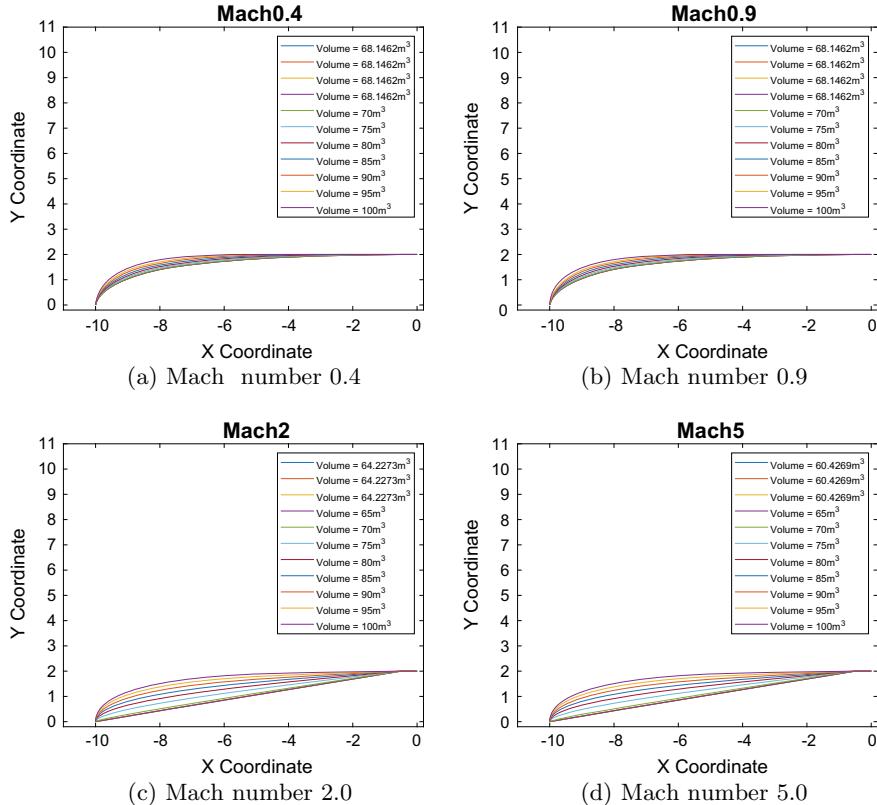


Fig. 11 Optimal configurations plotted for various volume for subsonic, transonic and supersonic Mach numbers

4 Conclusions

Extensive CFD simulations were carried out on various B-spline PLF configurations. It was seen that the C_D increases till transonic Mach numbers followed by a drop for supersonic Mach numbers. A surrogate model was created to model forebody C_D as a function of the configuration design variables and Mach number. A multiobjective optimization study was carried out to minimize the forebody drag coefficient C_D and maximize the volume. It was seen that the two objectives were conflicting and the configurations with lower optimal C_D had lower volume and vice versa. Also, another observation was that the optimal configurations had longest permissible length in the design space. Another notable observation was that the optimal transonic configurations were blunter than the optimal supersonic configurations with the same volume. Also for supersonic Mach numbers, it was seen that the conical configurations were optimal for lower volumes. However, the optimal transonic configurations were always seen to be blunt.

References

1. L. Piegl, W. Tiller, *The Nurbs Book* (1997)
2. J. Blazek, *Computational Fluid Dynamics: Principles and Applications* (Joe Hayton, 2005)
3. T.D. Economon, F. Palacios, S.R. Copeland, T.W. Lukaczyk, J.J. Alonso, *SU2: An Open-Source Suite for Multiphysics Simulation and Design* (2016)
4. U. Ayachit, J. Ahrens, A. Bauer, *The Paraview Guide* (Kitware Inc., 2019)

Design of Vortex Flaps for Reducing Approach Speed of a Supersonic Naval Fighter Aircraft



Soumyajit Behera[✉], C. Muthuraj, Balmiki Kumar, U. R. Srikanth, Sathish Sunnam, and R. Jolly

Abstract A naval fighter aircraft has conflicting demands for achieving low speed for landing on an aircraft carrier with minimum compromise on high-speed and supersonic capabilities. A sharp leading edge surface called LEVCON that can be deflected upwards for reducing approach speed has been implemented on an indigenous naval aircraft. Another solution explored in this paper is to vertically extend a panel near the wing apex called vortex flap to achieve increase in trim lift coefficient. It shows enhanced performance and improved aerodynamic characteristics for better handling qualities, without impacting high-speed flight. CFD simulations are run for a number of flap profiles, and the preferred family of shapes is identified for optimization by parameterizing six design variables. Some selected vortex flap geometries were then tested in a wind tunnel, and the results indicate that the approach speed target is met within all identified constraints. It is now planned to flight test the preferred shape for confirming the practical application of this concept on future naval aircraft designs.

Keywords Carrier-based fighter aircraft · Vortex flaps · CFD · Wind tunnel · Sensitivity analysis

1 Introduction

The design of a carrier-compatible supersonic fighter aircraft is a challenging multi-disciplinary problem. The design solutions for carrier landing and high-speed operations contradict each other. A solution for achieving desired carrier landing performance is in the form of a leading-edge vortex controller (LEVCON) implemented on a naval aircraft currently under development in the country. The authors propose an alternative solution with the potential of improved performance and handling qualities, referred to as vortex flaps.

S. Behera (✉) · C. Muthuraj · B. Kumar · U. R. Srikanth · S. Sunnam · R. Jolly
Aeronautical Development Agency, Bangalore, India
e-mail: soumyajitbehera01@gmail.com

The work on vortex flaps was started by Dr. DM Rao, and its earliest references are about four decades old. Thus, the vortex flaps are also referred to as Rao's vortex flaps (RVF) in this document. They are mentioned in the seminal paper by Polhamus [1] where he discusses the use of vortex flaps for reducing induced drag associated with delta wings. Rao, however, proposed their deployment to enhance subsonic performance of supersonic fighter aircraft [2, 3]. Research by Luckring [4], contemporary to that of Rao, was focussed on wing–strake interaction towards achieving solutions for naval aircraft. The outcomes of his research are applicable to the design and development of a new aircraft with fixed strakes. Retractable vortex flaps, on the contrary, can be incorporated into the aircraft at a later stage in the design and development process. Further, the design process is also simplified as the designer needs to be concerned with vortex flow interactions only in the low-speed regimes.

This is important as the flow structure over simple delta wings is complicated, as discussed by Visbal [5]. Visbal discusses the computational aspects of vortex interactions when computational fluid dynamics (CFD) tools were restricted by the availability of computational resources. Significant advancements have since been seen in the capabilities of numerical tools and the availability of computational resources. There are, however, problems faced by analysts while attempting to predict aircraft behaviour in flight regimes where combinations of large flow angles and high Mach numbers are present. Recent studies, such as those by Hitzel and Osterhuber [6], bring out the difficulty in accurately capturing fighter aircraft forces and moments at manoeuvre conditions using Reynolds-averaged navier–Stokes (RANS) solvers.

The aspects of vortex control via experimental methods are discussed by Gursul [7] in his review paper. Vortex flaps have been identified as effective vortex control devices. The review paper is a good source to earlier work carried out by Gursul as well as other authors. These authors have explored various aspects of delta wing aerodynamics in the presence of vortex flaps but their results broadly agree with that of Rao.

Ghee [8, 9] from NAVAIR, USA, has shown the use of vortex flaps to control flow separation over a moderately swept unmanned combat air vehicle (UCAV) configuration. Parametric studies were carried out with the effects of serration and tandem deployment studied via experimental techniques. Atkinson and Ferguson [10] carried forward the work via numerical tools.

Vortex flaps have featured in Boeing X-32A concept demonstration aircraft [11]. This aircraft was intended to operate from shore as well as aircraft carrier deck. According to the official web site of the manufacturer, this aircraft completed flight tests in 2001 and demonstrated match between expected and achieved performance.

Vortex flaps have thus been identified as versatile flow control devices. In this study, they are used to seek solutions for a tightly constrained naval supersonic fighter aircraft design problem.

2 Objective and Constraints

The main objective is to achieve the lowest approach speed below a specified target with minimum impact on the high subsonic and supersonic capabilities and with the least mass addition for its implementation. The constraints include:

1. The angle of attack which is restricted by pilot's vision requirement below the horizon for conducting a carrier approach [12].
2. The target approach speed is constrained by the maximum speed allowed for the hook to engage the arresting gear system on the carrier at the design wind on deck (WOD) that can be generated by the movement of the carrier.
3. The size and location of this surface is restricted by space available on the wings and from internal layout considerations.
4. A limit on maximum downward deflection of the elevons for trim so as to maintain residual deflection margin for adequate pitch and roll control authority.
5. Minimum L/D_Trim ratios for adequate acceleration and climb gradient in case of waveoff/bolter (aborted/missed carrier landing).
6. Maximum longitudinal instability to avoid abrupt pitch-up and wing drop around the approach AOA.
7. Acceptable lateral–directional characteristics.

Since this surface will be deployed at low speeds during carrier approach phase, the aerodynamic loads and hinge moments are relatively small and are thus considered redundant objectives.

3 Methods

The analyses were carried out using CFD++ solver with high-fidelity RANS CFD methods. The grid sizes varied between 35 and 40 million elements, and the first layer of viscous padding was such that a target y^+ of 1.0 is met. Two-equation $k\omega$ -SST method [13] was used for turbulence modelling. Various shapes were studied in the primary analysis. The design decisions were manually made to shortlist six geometries which were then tested on a 1:10 scale model in a low-speed wind tunnel for sensitivity analysis and validation of CFD simulations.

4 CFD Evaluation of Vortex Flap Shapes

The different shapes studied in the primary analysis are shown in Fig. 1 (left). In Fig. 1 (right), all shapes are evaluated in terms of the approach speed (V_{app}) achieved vs. their own true area. A least square fit to all data points defines the mean trend of vortex flap effectiveness. It was found that the gothic-rectangle shapes utilize more of

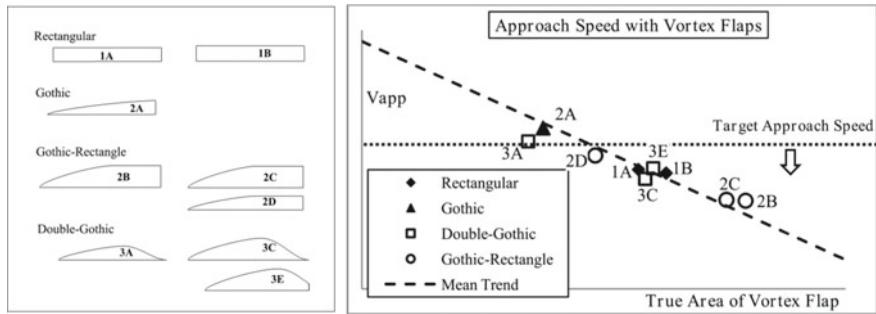


Fig. 1 Families of vortex flap shapes (left) and their effectiveness (right)

the available area on the wings and result in maximum reduction in approach speed. Thus, the gothic-rectangle family of shapes was selected for sensitivity analysis.

5 Sensitivity Analysis with Two Constraints

The six variables that define the design space for gothic-rectangle shapes are shown in Fig. 2. A sensitivity analysis was carried out with these parameters using CFD evaluation for two constraints, V_{app} and elevon deflection (δ_e). For the defined datum

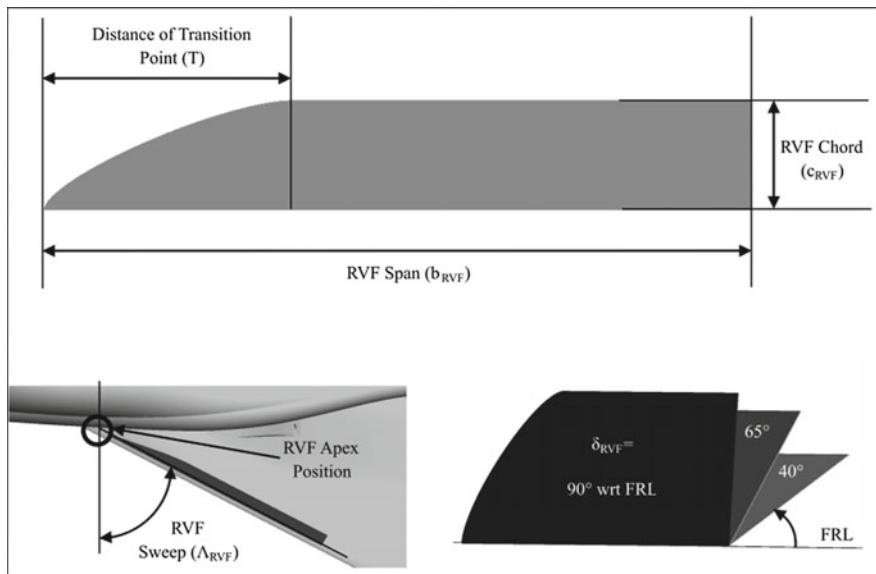


Fig. 2 Design variables for sensitivity analysis on vortex flaps

Table 1 Results of sensitivity analysis

Variable name	Variable normalized with	Variable datum value	Normalized Vapp gradient	Normalized δe gradient	Remarks
T	Nominal value	1.00	-0.01	0.21	Little sensitivity
b_{RVF}	Max limit	1.00	0.02	-0.18	Not desirable
c_{RVF}	Max limit	0.64	-0.07	0.68	Sensitive
$(X, Y)_{\text{RVF}}$	Min value	1.00	0.05	-0.13	Not desirable
Λ_{RVF}	Min value	0.31	0.07	-1.54	Not desirable
δ_{RVF}	Nominal value	1.00	-0.06	0.53	Sensitive

geometry, it was found that $V_{\text{app}} = 0.99V_{\text{app,target}}$ and $\delta e = 0.73\delta e_{\text{limit}}$. Normalized gradients of these two outputs wrt to the six variables are listed in Table 1.

6 Wind Tunnel Tests

The L/D ratios and lateral-directional stability parameters need to be evaluated over a range of AOA and AOS, and thus, their evaluation was considered computationally too expensive. Earlier experience indicates that the local longitudinal instability at these conditions is not consistently captured using CFD methods. For both these reasons, low-speed wind tunnel tests with vortex flaps were required to take further design decisions.

A wind tunnel model for the baseline configuration was not readily available. However, a 1:10 low-speed model for a similar variant of a flying aircraft (henceforth referred to as Baseline-II) was available and used for testing in HAL's low-speed wind tunnel. This configuration also features a LEVCON.

To decide the vortex flap configurations to be tested on Baseline-II, CFD studies were carried out on a number of vortex flap shapes, sizes, orientations and deflections, culminating in shortlisting of six promising vortex flap configurations for tunnel tests. As the tests progressed, three more shapes were added to the list.

Reduction in Vapp is achieved with placement of vortex flaps on Baseline-II due to:

1. Increase in basic CL
2. Increase in trim lift resulting from positive offset in CPM that requires further downward deflection of the elevons for trim, which then replicate the function of a flap on a wing's trailing edge.

The CL-AOA and CPM-AOA behaviour for a typical vortex flap is presented in Fig. 3. In Fig. 4 (left), the approach speeds achieved without and with various vortex flap configurations tested are shown. Based on Vapp considerations alone, all vortex flaps seem suitable for the aircraft. However, another criterion is dynamic directional stability parameter $Cn\beta$, dyn of the aircraft during carrier landing. An important contributor to $Cn\beta$, dyn particularly with increasing AOA, is the lateral stability derivative $Cl\beta$ shown here in Fig. 4 (right) from the wind tunnel test data. The larger chord vortex flap RVF-E suffers rapid loss in lateral stability closer to the approach AOA and is hence not a preferred choice.

Further, from aircraft handling quality consideration, the pitching moment derivative $CPM\alpha$ should not exceed a limit, particularly around the AOA identified for carrier approach, to avoid abrupt pitch-up and wing drop behaviour during this critical flight phase. In Fig. 5 (left), it is seen that vortex flap H1 has unacceptable $CPM\alpha$ characteristics which eliminates this option. Of the remaining, flap B has a better margin as compared to flap C and is therefore chosen as the most preferred vortex flap option for Baseline-II.

As a next step, a check is carried out on the level of (L/D) trim achieved with the deployment of the selected vortex flap on Baseline-II. The constraint on this arises from the capability to achieve adequate acceleration and climb gradient during waveoff and bolter manoeuvres. It can be seen in Fig. 5 (right) that within the AOA range of relevance, the selected vortex flap exceeds the minimum (L/D) trim constraint.

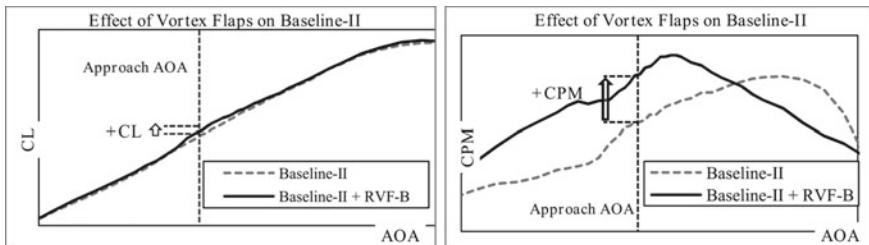


Fig. 3 General behaviour of a vortex flap

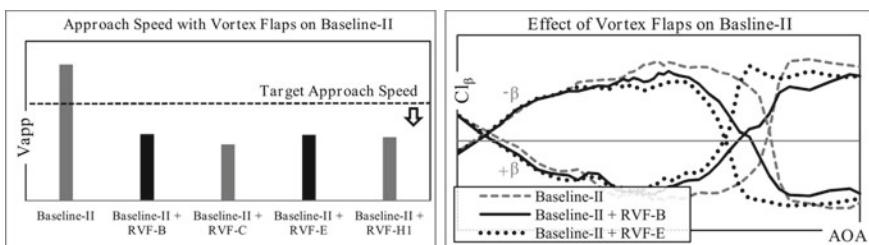


Fig. 4 Approach speed for all vortex flaps (left) and effect of chord increase on rolling moment derivative (right)

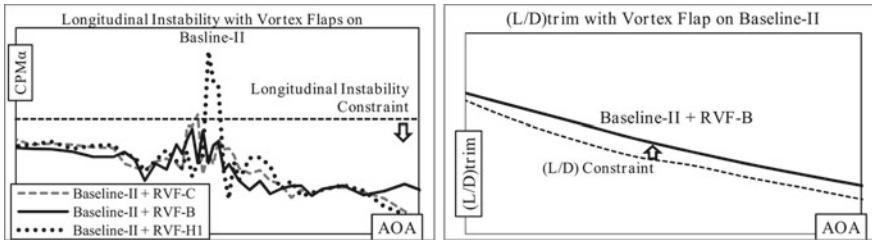


Fig. 5 Effect of flap deflection on $CPM\alpha$ (left); checking (L/D) constraint for chosen vortex flap (right)

7 Flow Visualization from CFD

CFD simulations carried out for Baseline-II and Baseline-II + RVF-B configurations at approach AOA, zero AOS and tunnel running conditions including model scale give insight on the flow phenomena. In Fig. 6, the C_p contours are plotted around the vortex cores (dashed black lines).

It can be seen that the vortex flap strengthens the core in the forward portion of the wing and shifts it inboards. The strengthened core imparts lower pressure throughout the wing surface but the effect is more pronounced in the forward portion ahead of

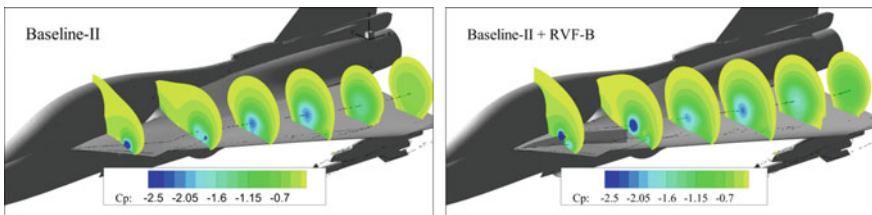


Fig. 6 Flow visualization from RANS CFD results



Fig. 7 Boeing X-32A CTOL concept demonstrator with vortex flaps deployed prior to landing.
Source [14]

the aircraft CG, resulting in an increase in basic CL as well as CPM as seen in Fig. 3. Further, the inboard movement of the vortex core in the presence of vortex flaps in these zero AOS cases could be correlated to the decrease in rolling moment as seen in Fig. 4 (right).

8 Further Optimization

An optimized solution is sought in the neighbourhood of parameters defining RVF-B. The objective function and constraints are approximated by 2nd order polynomials. Interior point algorithm with a basic nonlinear solver yields a solution that is better than that with RVF-B by less than 1 kt reduction in trim approach speed. The difference is within the error band of approach speed estimation methods. Further, another campaign to ascertain the benefits of this optimized solution would cost the program time and money while promising little improvement. Thus, the vortex flap RVF-B is accepted as the optimized solution and recommended for implementation on the aircraft.

9 Conclusions

CFD studies, supported by wind tunnel tests, show that a vortex flap deployed on the upper surface of the wing close to its leading edge and approximately aligned with the vortex system can significantly reduce the trim speed at constant AOA. Six geometric parameters define the design space for varying size, shape, orientation and deflection of the vortex flap. CFD simulations are carried out to shortlist a few promising designs that were then tested on a 1:10 scale wind tunnel model at low speeds. Based on these results, the most preferred design is identified for the minimum approach speed objective and meeting a number of other criteria defined as constraints for its acceptable implementation. It is planned to flight test this shape for confirming the practical application of this concept on future naval aircraft designs.

Acknowledgements It is acknowledged that the inspiration for this work is drawn from Dr. DM Rao's work and its application on X-32A concept demonstrator (as shown in Fig. 7). Efforts put in by HAL's low-speed wind tunnel test team for an innovative but effective technique to quickly realize and test multiple vortex flap shapes are also acknowledged.

References

1. E.C. Polhamus, Applying slender wing benefits to military aircraft. *J. Aircr.* **21**(8) (1984). <https://doi.org/10.2514/3.45023>
2. K.D. Höffler, D.M. Rao, M.C. Frassinelli, Low-speed aerodynamics of apex fences on a tailless delta configuration. *J. Aircr.* **25**(4) (1988). <https://doi.org/10.2514/3.45563>
3. D.M. Rao, J.F. Campbell, Vortical flow management techniques. *Prog. Sci.* **24**, (1987). [https://doi.org/10.1016/0376-0421\(87\)90007-8](https://doi.org/10.1016/0376-0421(87)90007-8)
4. J.M. Luckring, Aerodynamics of strake-wing interactions. *J. Aircr.* **16**(11) (1979). Article No. 78-1202R
5. M.R. Visbal, Computational and physical aspects of vortex breakdown on delta wings. in *33rd AIAA Aerospace Sciences Meeting and Exhibit*. Jan (1995). <https://doi.org/10.2514/6.1995-585>
6. S.M. Hitzel, R. Osterhuber, Enhanced maneuverability of a delta-canard combat aircraft by vortex flow control. *J. Aircr.* **55**(3) (2017). <https://doi.org/10.2514/1.c034473>
7. I. Gursul, Z. Wang, E. Vardaki. Review of flow control mechanisms of leading-edge vortices. *Prog. Aerosp. Sci.* **43**, (2007). <https://doi.org/10.1016/j.paerosci.2007.08.001>
8. T.A. Ghee, A deployable serrated flap for air vehicle control of a representative UCAV configuration. in *1st Flow Control Conference*, June 2002. <https://doi.org/10.2514/6.2002-3276>
9. T.A. Ghee, Parametric evaluation of deployable serrated flaps on a representative UCAV configuration. in *21st Applied Aerodynamics Conference* June (2003). <https://doi.org/10.2514/6.2003-3928>
10. M. Atkinson, F. Ferguson, A computational fluid dynamics investigation of the 1303 UCAV configuration with deployable Rao vortex flaps. in *44th AIAA Aerospace Sciences Meeting and Exhibit*. Jan (2006). <https://doi.org/10.2514/6.2006-1262>
11. Joint Strike Fighter: One Aircraft, Many missions. *Aviat. Week Space Technol.* 25 May 1998, pp. S3–S18
12. AS-5263(US Navy), Naval air systems command specifications, guidelines for the preparation of standard aircraft characteristic charts and performance data, piloted aircraft (Fixed Wing), Oct 1986
13. F.R. Menter, Two-equation eddy-viscosity turbulence model for engineering applications. *AIAA J.* **32**, (1994). <https://doi.org/10.2514/3.12149>
14. <https://www.cavok.com.br/blog/em-tempo-de-crise-no-programa-do-f-35-que-tal-relembra-do-seu-concorrente/>

Optimization Applications: Automotive

Performance Comparison of Discrete Kalman Filter and Dynamic Programming Technique for Pavement Roughness Identification



O. A. Shereena, C. G. Krishnanunni, and B. N. Rao

Abstract Roughness profile identification is indispensable in vehicle dynamics research since it is the major source of excitation to the vehicle body. The present paper deals with pavement roughness identification based on the dynamic response of a quarter car vehicle model. The forward problem involves computing the vehicle responses using Newmark's beta method. Noise in the measured data poses a serious challenge to the roughness identification problem. In the present paper, the computed dynamic responses are contaminated with Gaussian white noise in order to simulate actual field measurements. Two techniques are employed for identification, and their efficiency is compared. Roughness identification based on dynamic programming technique involves the formulation of an objective function that computes the least square error between measured and identified vehicle responses. Tikhonov regularization is incorporated to deal with the ill-posed inverse problem. Bellman's principle of optimality is employed to minimize the objective function and compute the pavement roughness profile. In case of discrete Kalman filter-based identification, an optimal filter considering roughness as an unknown input rather than as a state variable is adopted. The efficiency of both methods in dealing with measurement error and estimation accuracy is compared. The computational efficiency and the feasibility of the two methods in tackling different scenarios in the identification problem are also demonstrated.

Keywords Dynamic programming · Discrete Kalman filter-unknown inputs · Roughness identification · Tikhonov regularization

O. A. Shereena · C. G. Krishnanunni · B. N. Rao (✉)
Indian Institute of Technology Madras, Chennai, India
e-mail: bnrao@iitm.ac.in

O. A. Shereena
e-mail: ce14d020@mail.iitm.ac.in

C. G. Krishnanunni
e-mail: researchchunni@gmail.com

1 Introduction

Pavement roughness serves as the major source of excitation to the vehicle body and as such influences the ride comfort and vehicle dynamics [1]. Also, it has been proved that the deterioration of pavement is a function of the road roughness profile as it induces high vehicle axle loading on the pavement. The roughness profiles used in such studies must reflect the true situation of the pavement under consideration as it influences the vehicle design and pavement maintenance [1, 2]. Consequently, the pavement roughness identification has turned out to be an active area of research in the last decade.

It is a common practice to classify pavement roughness for evaluating road roughness. The International Roughness Index [3] and the International Standards Organization (ISO) [4] are the two popular methods for classifying road roughness. Since local roughness features affect the ride comfort, pavement stresses, and vehicle fatigue damage [5, 6], it is necessary to obtain accurate roughness profile estimates in the time domain. The earliest method for road profile identification is based on static methods such as the use of rod and level equipment, which proved to be time-consuming [7]. The development of inertial profilometer in the early 1960s paved the way for accurate high-speed road profiling methods [7]. The technique makes use of a laser-based technology mounted on a vehicle for measuring pavement elevations. The effects of dynamic vehicle response on the roughness measurements are mitigated utilizing accelerometer readings from the vehicle [7]. However, the cost associated with the laser-based technology is high and is a disadvantage. More recently, efficient and low-cost approaches for road roughness identification based on the dynamic response of an instrumented vehicle has been attempted by various researchers [8, 9]. Fauriat et al. [10] proposed a data processing algorithm based on Kalman filter for roughness identification.

In this work, the dynamic responses (acceleration response) of a quarter car vehicle model are utilized for roughness identification. In order to simulate actual field measurements, Gaussian white noise is added to the computed dynamic responses. The identification procedure based on dynamic programming technique defines an objective function that calculates the least square error between measured and identified vehicle responses and is minimized based on Bellman's principle of optimality [11]. The effect of measurement noise is handled by incorporating a regularization parameter in the objective function [12]. In this case, choosing an appropriate regularization parameter for different noise levels holds the key to fast and accurate road profiling. Identification based on a filtering framework uses the recently developed unbiased minimum variance estimator [13] for roughness estimation. Contrary to the existing methods, the road roughness is not assumed as a state variable in this approach. For this technique, it is mandatory that the measurement equation should include the unknown inputs information. These constraints can be a limitation to choose the sensors for designing the filter. In this paper, the two techniques for roughness characterization are compared in terms of accuracy, efficiency, and any constraints that might impair the respective applicability in different scenarios.

2 System Model and Identification

In the present study, a quarter car model moving with velocity v is shown in Fig. 1 is employed.

The equations of motion of the quarter car model are as follows:

$$m_b \ddot{x}_b + c_s (\dot{x}_b - \dot{x}_w) + k_s (x_b - x_w) = 0 \quad (1)$$

$$m_w \ddot{x}_w + c_s (\dot{x}_w - \dot{x}_b) + k_s (x_w - x_b) + k_t x_w = k_t x_r \quad (2)$$

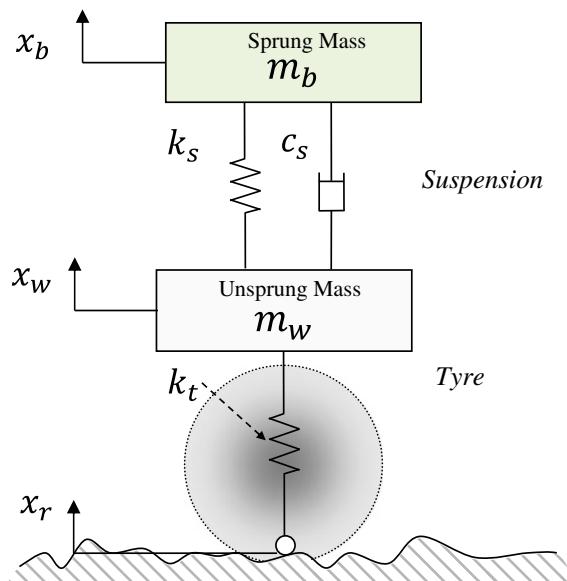
where x_w and x_b represent the unsprung mass displacement and sprung mass displacement, respectively. x_r represents the road roughness.

2.1 Minimum Variance Unbiased Filter for Roughness Identification [13]

The discretized state-space equation based on Euler's method is given by [14]:

$$x_{k+1} = (I + hA_c)x_k + hB_c^* u_k^* \quad (3)$$

Fig. 1 Quarter car model



where h is the sampling time. The $*$ is used to denote the unknown input. The general form of the measurement model is given as:

$$y_k = Cx_k + D^*u_k^* \quad (4)$$

For quarter car model, the state x consists of vertical vehicle displacements and velocities. The unknown input vector is the discrete point roughness, x_r . The state vector x is given by:

$$x = [x_b \dot{x}_b x_w \dot{x}_w]^T \quad u^* = x_r \quad (5)$$

The matrix A_c and the vector B_c^* are given as follows:

$$A_c = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_s}{m_b} & -\frac{c_s}{m_b} & \frac{k_s}{m_b} & \frac{c_s}{m_b} \\ 0 & 0 & 0 & 1 \\ \frac{k_s}{m_w} & \frac{c_s}{m_w} & -\frac{k_s+k_t}{m_w} & -\frac{c_s}{m_w} \end{pmatrix}, \quad B_c^* = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\frac{k_t}{m_w} \end{pmatrix} \quad (6)$$

The measurement vector y_k consists of the sprung mass acceleration and unsprung mass acceleration.

$$y_k = [\ddot{x}_b \ddot{x}_w]^T \quad (7)$$

The terms C and D^* are:

$$C = \begin{pmatrix} -\frac{k_s}{m_b} & -\frac{c_s}{m_b} & \frac{k_s}{m_b} & \frac{c_s}{m_b} \\ \frac{k_s}{m_w} & \frac{c_s}{m_w} & -\frac{k_s+k_t}{m_w} & -\frac{c_s}{m_w} \end{pmatrix}, \quad D^* = \begin{pmatrix} 0 \\ -\frac{k_t}{m_w} \end{pmatrix} \quad (8)$$

With the above coefficient matrices, the measurement model satisfies the observability condition. One measurement quantity, i.e., unsprung mass acceleration containing the unknown input component is included to satisfy the constraint, $D^* \neq 0$. The developed matrices are fed to the recursive filtering algorithm [13] for identification of x_r .

Minimum variance unbiased estimator used in this work has the structure of a Kalman filter and is designed for discrete linear systems with unknown inputs. This recursive filter performs joint estimation of states and inputs, where the states and inputs are interconnected [13]. For the application of the filter, no prior knowledge regarding the unknown inputs is required.

General description of the system is as follows:

$$x_{k+1} = A_k x_k + B_k u_k + w_k \quad (9)$$

$$y_k = C_k x_k + D_k u_k + \vartheta_k \quad (10)$$

where $x_k \in R^n$ is the state vector, $u_k \in R^p$ is the unknown input vector and $y_k \in R^m$ is the measurement vector. $w_k \in R^n$ and $\vartheta_k \in R^m$ are the process noise and the measurement noise, respectively, uncorrelated white noises with mean zero and with covariance matrices $E[w_k w_k^T] = Q_k$ and $E[\vartheta_k \vartheta_k^T] = R_k$. The coefficient matrices A_k , B_k , C_k and D_k are assumed to be known. The rank of the matrix D_k is assumed to be equal to p , the size of the unknown input vector. The condition ($\text{rank } D_k = p$) is the necessary and sufficient condition for the unbiasedness of the estimator and is critical in determining the observability of the system.

The summary of the filtering equations is given below [13]:

Initialization:

$$\begin{aligned}\hat{x}_0 &= E[x_0] \\ P_0^x &= E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T]\end{aligned}\quad (11)$$

Time update:

$$\begin{aligned}x_{k+1|k} &= A_k \hat{x}_{k|k} + B_k \hat{u}_k \\ P_{k+1|k}^x &= \begin{bmatrix} A_k & B_k \end{bmatrix} \begin{bmatrix} P_{k|k}^x & P_k^{xu} \\ P_k^{ux} & P_k^u \end{bmatrix} \begin{bmatrix} A_k^T \\ B_k^T \end{bmatrix} + Q_k\end{aligned}\quad (12)$$

Estimation of unknown input:

$$\begin{aligned}\tilde{R}_k &= C_k P_{k|k-1}^x C_k^T + R_k \\ M_k &= (D_k^T \tilde{R}_k^{-1} D_k)^{-1} D_k^T \tilde{R}_k^{-1} \\ \hat{u}_k &= M_k (y_k - C_k \hat{x}_{k|k-1}) \\ P_k^u &= (D_k^T \tilde{R}_k^{-1} D_k)^{-1}\end{aligned}\quad (13)$$

Measurement update:

$$\begin{aligned}K_k &= P_{k|k-1}^x C_k^T R_k^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k (y_k - C_k \hat{x}_{k|k-1} - D_k \hat{u}_k) \\ P_{k|k}^x &= P_{k|k-1}^x - K_k (\tilde{R}_k - D_k P_k^u D_k^T) K_k^T \\ P_k^{xu} &= (P_k^{ux})^T = -K_k D_k P_k^u\end{aligned}\quad (14)$$

2.2 Dynamic Programming Technique for Roughness Identification [15]

The profile roughness identification can also be posed as an optimization problem. The state-space equation and the measurement model for this case are written as:

$$x_{k+1} = Ex_k + Ff_k \quad (15)$$

$$y_k = Gx_k \quad (16)$$

where the force $f_k = k_t u_k^*$. The state vector is given as: $x = [x_b \ x_w \ \dot{x}_b \ \dot{x}_w \ \ddot{x}_b \ \ddot{x}_w]$. Measurement vector is given by: $y = [\ddot{x}_b \ \ddot{x}_w]$. In this case, since only the acceleration measurements are used, G is a 2×6 matrix. Based on Newmark discretization, the matrices E and F can be computed as follows [15]:

$$\begin{aligned} E &= \begin{pmatrix} I & 0 & -\beta h^2 I \\ 0 & I & -\gamma h I \\ K_v & C_v & M_v \end{pmatrix}^{-1} \begin{pmatrix} I & hI & (0.5 - \beta)h^2 I \\ 0 & I & (1 - \gamma)hI \\ 0 & 0 & 0 \end{pmatrix} \\ F &= \begin{pmatrix} I & 0 & -\beta h^2 I \\ 0 & I & -\gamma h I \\ K_v & C_v & M_v \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 0 \\ I \end{pmatrix} \end{aligned} \quad (17)$$

where β and γ are the parameters of Newmark's scheme. I is the identity matrix. K_v , C_v , M_v are the stiffness matrix, damping matrix, and mass matrix of the vehicle based on Eq. (1). In this case, the goal is to find the roughness u^* that causes the system to best match the measurements (\hat{y}_k). The objective function is defined as:

$$\text{Error} = \sum_{k=1}^N (y_k - \hat{y}_k)^T (y_k - \hat{y}_k) + (f_k)^T \lambda_2 (f_k) \quad (18)$$

The effect of noise is undertaken by the regularization parameter λ_2 so that the problem becomes a well-posed one. The above objective function is minimized for f_k based on dynamic programming and Bellman's principle of optimality. A set of recursive relations can be derived [15] for determining the optimal solution of Eq. (8). The recursive equations for determining f_k are given below [15]:

$$L_{k-1} = G^T G + E^T (L_k - H_k^T V_k H_k / 2) E \quad (19)$$

$$S_{k-1} = -2G^T \hat{y}_{k-1} + E^T (I - H_k^T V_k F^T) S_k \quad (20)$$

$$f_{k-1} = -V_k F^T S_k - V_k H_k E x_{k-1} \quad (21)$$

Table 1 Parameters of the quarter car vehicle model

m_b (kg)	m_w (kg)	$k_s(\frac{N}{m})$	$c_s(\frac{Ns}{m})$	$k_s(\frac{N}{m})$
18,000	500	6.528×10^5	2×10^5	7.50^5

where $V_k = (2\lambda_2 + 2F^T L_k F)^{-1}$, $H_k = 2F^T L_k$. The solution is computed by starting at the end of the process, and working backwards to $k = 1$.

3 Numerical Example

3.1 Parameters of the Vehicle Model and Simulated Road Roughness

For the present study, a quarter car model with the parameters mentioned in Table 1 is adopted. The roughness profile of a 30 m stretch of road is considered in the present study.

The roughness profile used in this study is simulated based on the power spectral density of road roughness provided by ISO 8608. A class road roughness profile is considered. The generated road roughness profile serves as an input to Eq. (15) that computes the dynamic vehicle responses based on Newmark's beta method. To simulate actual field measurements, Gaussian white noise is added to the simulated responses. In this study, the effect of 10% and 15% noise in measurement on the identified roughness profile is compared.

3.2 Results for Identification Based on Dynamic Programming Technique

In this method, the regularization parameter λ_2 is tuned for the corresponding noise levels so that the least square error in the estimation of roughness profile is minimum. The L-curve method [16] is employed for this purpose. For 10% noise in the measurement, the corner of the L-curve occurs at $\lambda_2 = 1.4 \times 10^{-11}$ with least square error of 6.25×10^{-4} m. For 15% noise in the measurement, λ_2 is obtained as 3.5×10^{-11} with least square error of 7.921×10^{-3} m. The identified roughness profile is shown in Fig. 2. It can be observed that the magnitude of regularization to be applied has slightly increased due to an increase in the noise level.

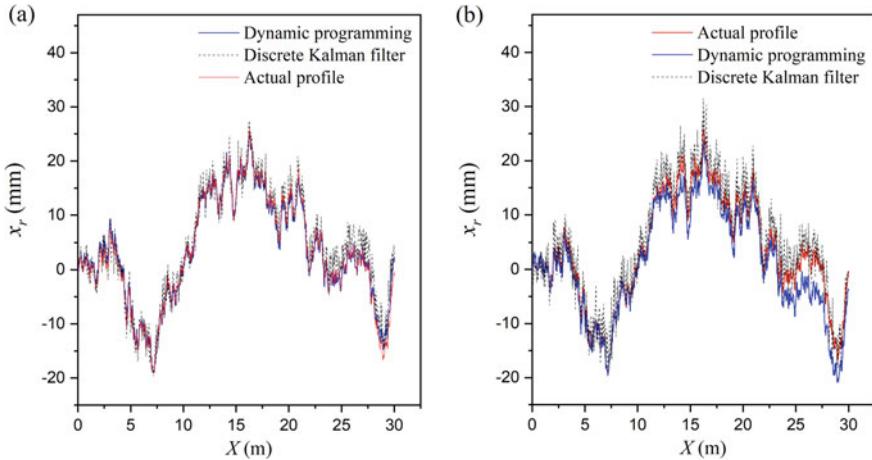


Fig. 2 Estimated and actual roughness profiles for: **a** 10% measurement noise in the responses; **b** 15% measurement noise in the responses

3.3 Results for Identification Based on Kalman Filter-Unknown Input

Noise-contaminated acceleration measurements used in the former case is used for identification. In this case, the initial estimates, the system noise covariance matrix (Q), measurement noise covariance matrix (R) [4] are assumed to be arbitrary values ($\hat{x}_{0/0} = [0, 0, 0, 0]^T$, $u_0^* = 0$, $Q = [0.1^2, 0.1^2, 0.1^2, 0.1^2]$, $R = [1, 1]$). Error covariance matrix (for state), $P_{0/0} = \text{diag}(5, 5, 5, 5)$, and for unknown input $S_0 = 50$ is adopted. The results are shown in Fig. 2. The least square error is obtained as 5×10^{-3} m and 8.1×10^{-3} m for 10% and 15% measurement noise, respectively.

3.4 Performance Comparison

To obtain an unbiased inference regarding the performance, the same measurements, $y = [\ddot{x}_b \ddot{x}_w]$ have been used in both the methods. Based on the results presented in Fig. 2, the following comparisons are made:

- For both 10% as well as 15% measurement noise, the estimated roughness profile based on dynamic programming technique is more accurate.
- Dynamic programming involves only a single parameter to be tuned (λ_2) leading to better performance. The parameter λ_2 was tuned based on L -curve method [16] using different sets of noise-contaminated acceleration responses.
- The discrete Kalman filter involves multiple parameters to be tuned. In the case of the discrete Kalman filter, the parameters are mostly assumed to be arbitrary

values [14]. Multi-parameter tuning has been a major issue with Kalman filter and requires a tedious and time-consuming procedure [17].

- In addition, the constraint $D^* \neq 0$ and the observability condition pose a limitation to the use of the discrete Kalman filter-unknown input (DKF-UI) for identification. For example, if the vehicle displacements (x_w, x_b) are measured in place of accelerations, then Eq. (4) gives $D^* = 0$, thereby violating the constraint. The discrete Kalman filter cannot be used in such cases.

4 Conclusions

Based on the study, the following conclusions are made:

- Dynamic programming technique yields more accurate results in comparison with discrete Kalman filter when:
 - Limited measurement data is available (due to the constraint condition on matrix D^*).
 - Arbitrary values are assigned to the noise covariance matrices.
- The constraint on D^* and the observability condition further imposes a restriction on the applicability of the filter.
- It is feasible to employ dynamic programming technique for roughness identification due to the relative ease in parameter tuning (*L*-curve method) and is free of constraints.

References

1. T.D. Gillespie, *Measuring Road Roughness and Its Effects on User Cost and Comfort*: A symposium, vol. 884. (ASTM International, 1985)
2. T.D. Gillespie, Effect of heavy-vehicle characteristics on pavement response and performance. Transp. Res. board, **353** (1993)
3. M.W. Sayers, Guidelines for conducting and calibrating road roughness measurements. (1986)
4. ISO 8608, Mechanical vibration-Road surface profiles-Reporting of measured data, International Standards organization (1995)
5. K. Bogsjö, I. Rychlik, Vehicle fatigue damage caused by road irregularities. Fatigue Fract. Eng. Mater. Struct. **32**(5), 391–402 (2009)
6. C.M. Kuo, C.R. Fu, C.Y. Chen, Effect of pavement roughness on rigid pavement stress. J. Mech. **27**(1), 1–8 (2011)
7. M.W. Sayers, The little book of profiling: basic information about measuring and interpreting road profiles. (1998)
8. N.K. Harris, A. González, E.J. O'Brien, P. McGetrick, Characterisation of pavement profile heights using accelerometer readings and a combinatorial optimization technique. J. Sound Vib. **329**(5), 497–508 (2010)
9. P. McGetrick, C.W. Kim, A. González et al., Dynamic axle force and road profile identification using a moving vehicle. Int. J. Archit. Eng. Constr. **2**(1), 1–16 (2013)

10. W. Fauriat, N. Matstrand, A. Gayton, A. Beakou, T. Cembrzynski, Estimation of road profile variability from measured vehicle responses. *Veh. Syst. Dyn.* **54**(5), 585–605 (2016)
11. R. Bellman, *Dynamic Programming* (Princeton University Press, Princeton, NJ, 1957)
12. D.M. Trujillo, H.R. Busby, *Practical Inverse Analysis in Engineering* (CRC Press LLC, Boca Raton, FL, 1997)
13. S. Gillijns, B. De Moor, Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough. *Automatica* **43**(5), 934–937 (2007)
14. S.W. Kang, J.S. Kim, G.W. Kim, Road roughness estimation based on discrete Kalman filter with unknown input. *Veh. Syst. Dyn.* **57**(10), 1530–1544 (2019)
15. T. Zhu, S.N. Xiao, G.W. Yang, Force identification in time domain based on dynamic programming. *Appl. Math. Comput.* **235**, 226–234 (2014)
16. P.C. Hansen, The *L*-curve and its use in numerical treatment of inverse problems. *Comput. Inverse Prob. Electrocardiol.* 119–142 (2000)
17. Y. Oshman, I. Shaviv, Optimal tuning of a Kalman filter using genetic algorithms, in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, vol. 4558 (2000)

Multi-disciplinary Optimization of Sensor Bracket



Rajapandian Ramamoorthy and Shabareesh Kumar

Abstract Sensors are used in vehicles for sensing traffic which can be used for cruise control and collision warning. Sensor mounting bracket needs to often satisfy contradictory performance requirements such as adequate dynamic stiffness so as to exhibit adequate frequency response function (FRF) characteristics and adequate compliance to meet lower leg injury metric when subjected to pedestrian protection loadcase. The current manual design process utilizing incremental design iterations requires significant effort from the engineers to get the design for meeting the requirements, which also may not be optimized. Purpose of this piece of work is to demonstrate application of multi-disciplinary optimization (MDO) process to arrive at optimized bracket design while satisfying the contradictory loadcases and also to minimize mass. Parametric design optimization with shape and gauge variables is considered. Choosing the right shape variables for MDO which are sensitive to the performance can often be a challenge. Here, the typical sensitive shape parameters are judiciously decided after examining the results of free-size optimization. Suitable meta-modeling technique and optimization scheme are used to arrive at the optimal design. It is found that MDO process is able to perform well in two common design scenarios: (1) mass savings and (2) meeting performance target when the baseline design is not meeting the requirements.

Keywords MDO · FRF · Pedestrian protection

R. Ramamoorthy (✉) · S. Kumar
General Motors Technical Center India (GMTCI), Bangalore, India
e-mail: rajapandian.ramamoorthy@gm.com

S. Kumar
e-mail: shabareesh.kumar@gm.com

1 Introduction

Trends in automotive safety are pushing sensor systems to higher levels of accuracy and reliable target identification for blind spot detection and collision prevention assistance. Consequentially, engineers need to better understand how to design the sensor mounting bracket for body exterior loadcase and also pedestrian protection. Designing the sensor module mounting bracket involves two contradictory CAE domains such as pedestrian protection and body exterior loadcases. To satisfy pedestrian protection, the bracket is expected to deform and absorb most of the kinetic energy from lower leg impact loadcase resulting in lower injury metrics. In addition to that, bracket is required to have high stiffness under dynamic vibration in body exterior loadcase. It is important to develop the common process of optimization (balanced optimum design) for both the simulation domain areas. Traditionally, each of the individual CAE domain carries out simulations and design performance analysis to meet certain pre-defined product performance targets. However, the design changes happen in parallel within each domain with parallel communication with the design group. This iterative loop continues until each of the CAE group meets the specified performance targets which mostly involves the penalty of mass addition. Thus, under the traditional approach, this iteration loop happens individually in each of the CAE domains as shown in Fig. 1. After going through several design iterations, this method may arrive at a feasible solution which meets all product performance constraints, but this feasible design probably may not be an optimal solution in terms of mass saving.

This paper describes how the free-size optimization for concept generation and subsequent process with MDO process for detailed optimization was developed for sensor bracket so as to meet target performance requirements for both contradictory loadcases.

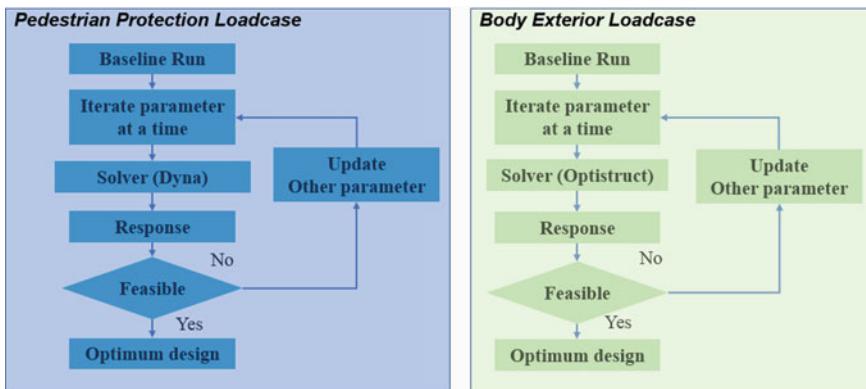


Fig. 1 Traditional optimization involving contradictory loadcases

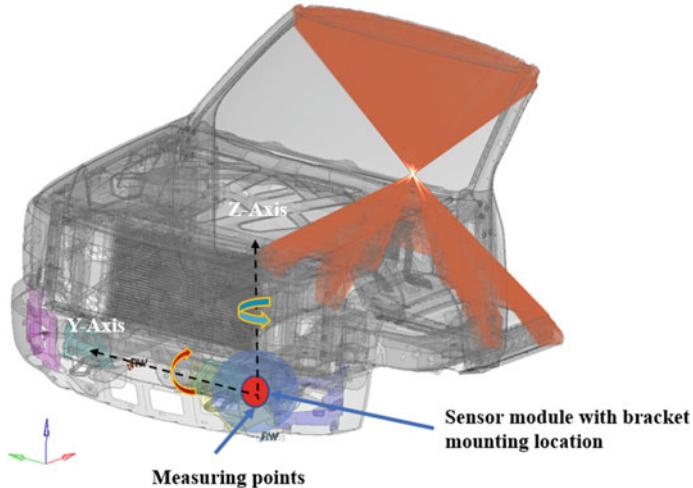


Fig. 2 Overview of body exterior loadcase. Source of the model from—<https://www.nhtsa.gov/crash-simulation-vehicle-models>

2 Loadcases for Sensor Bracket and Design Requirements

2.1 Body Exterior Loadcase

Frequency response function (FRF) loadcase is used to evaluate the dynamic response of the sensor bracket to calculate horizontal and vertical angular movement of the module. The application of gravity load in three directions through body across frequency range (0–280 Hz) of interest is shown in Fig. 2. The associated performance design requirements are vertical angle < GM targets (X, Y, Z excitations) and horizontal angle < GM targets (X, Y, Z excitations).

2.2 Pedestrian Protection Loadcase

FLEXible Pedestrian Legform Impactor [1] (FLEX PLI) is impacted at 40 km/h into front exterior assembly. It is noted that impactor engages the front fascia assembly and sensor bracket assembly as shown in Fig. 3. Legform impactor located from center of vehicle at 450 mm. The associated global technical regulation (GTR) [2] performance design requirements are Tibia Bending Moment < 272 Nm, MCL Elongation < 17 mm and LCL/ACL/PCL Elongations < 10 mm.

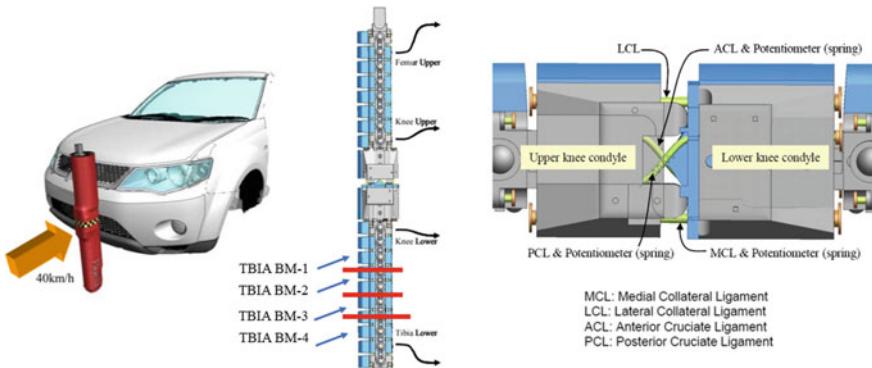


Fig. 3 Overview of pedestrian protection loadcase

3 Development of Methodology

Proposed methodology has two areas of the design process: Free-size optimization for identification of right shape variables that are sensitive to performance and MDO process with shape and gauge variables to carry out detailed optimization process. The main steps of the proposed methodology are shown in Fig. 4. In the first step, FRF loadcase response was used for free-size optimization. In detailed optimization step, the multi-disciplinary approach was used, in which responses from pedestrian protection and body exterior loadcase were used for optimization.

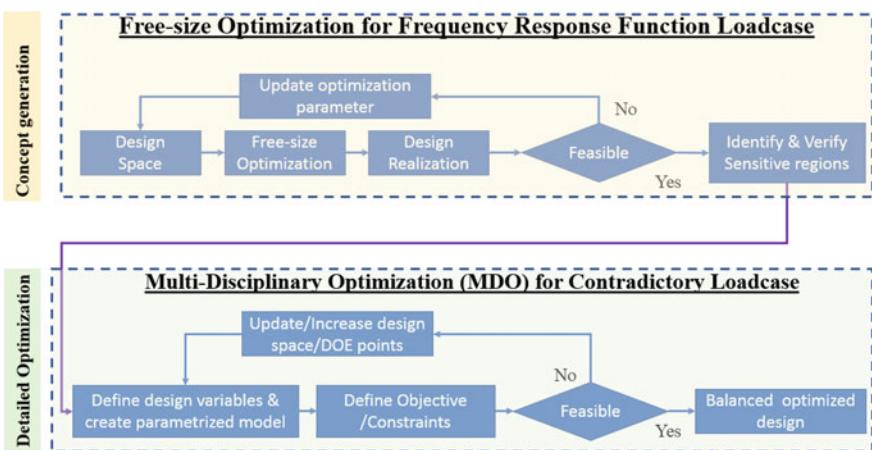
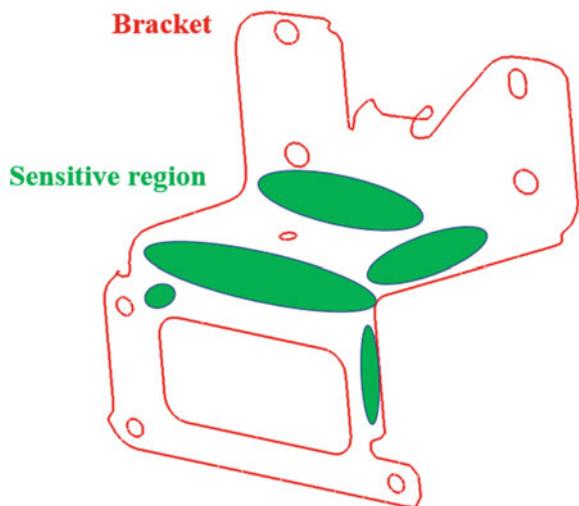


Fig. 4 Basic steps of proposed methodology in a flowchart

Fig. 5 Optimum thickness distribution (sensitive region with high thickness) from free-size optimization



3.1 Concept Generation (Identification of Sensitive Shape Variables)

Free-size optimization method [3] provides direction to achieve solution faster, significantly reducing product development times. For optimization purposes [4] using “Sizing Variables”, i.e., cross section and thickness of finite elements, many mathematical programming approaches have been tested and implemented into finite element programs. Free-size optimization in Altair-Optistruct [5] is very useful method to predict sensitive shape parameters in early stage of concept generation. This method optimizes the thickness of every element in the design space to generate an optimized thickness distribution, for the given objective under given constraints. Free-size optimization considers the FRF loadcase and with multiple constraints. The optimized high thickness distributions are very sensitive, and it is considered for detailed optimization (MDO) as shown in Fig. 5.

3.2 Detailed Optimization

Multi-disciplinary design optimization (MDO) uses optimization methods to solve design problems by incorporating all relevant disciplines simultaneously. In vehicle engineering, the objective typically is to optimize the weight of the body exterior while maintaining attribute performance involving various loadcases from crash safety, NVH, durability and strength, forming and vehicle dynamics [6–11]. Mathematical representation of optimization problem definition is shown below. Now that we have defined the objective function, design variables and constraints.

Minimize $f(x)$ —Mass of bracket
 by varying $x \in R^n$ —bracket shape & thickness
 Subject to Constraints $c_j(x) < 0; j = 1, 2, \dots m$
 $c_k(x) < 0; k = 1, 2, \dots n$

f objective function, output (Minimizing the mass of bracket)
 x vector of design variables, inputs (e.g., bracket shape and thickness); bounds can be set on these variables.
 c_j & c_k Pedpro loadcase constraints and body exterior loadcase.

where f is the objective function, x is the vector of n design variables, c_j and c_k are constraints.

The objective was to develop a methodology using multi-disciplinary approach for sensor bracket involving body exterior and pedestrian protection loadcases together. MDO process flowchart is shown in Fig. 6.

Number of geometric shape variables (sensitive regions) were 15 and 1-gauge variable, totally 16 variables considered. Geometric feature with shape morphing was implemented using DEP-Morphing software [12] in a parameterized model to

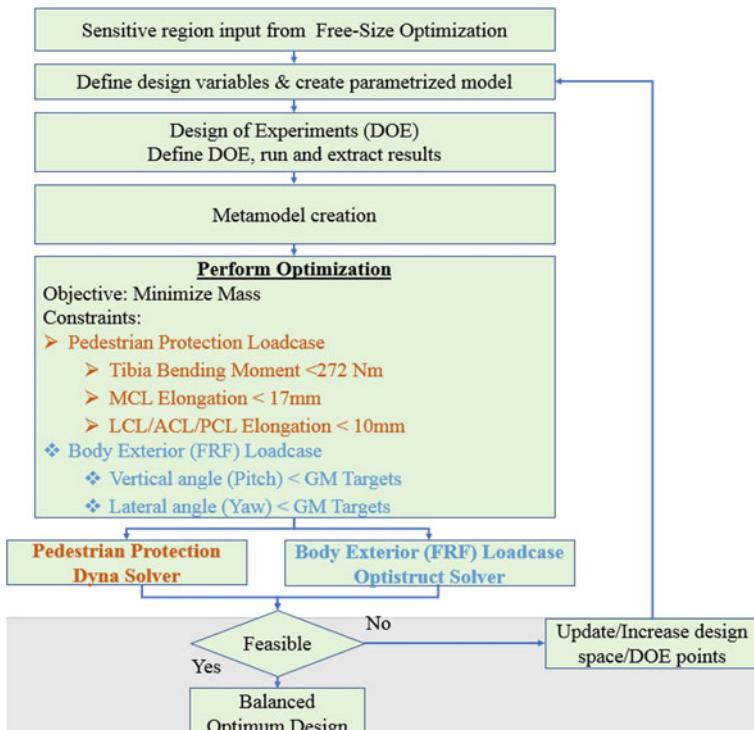


Fig. 6 MDO process flowchart

minimize the mass while determining the balanced optimum bracket design with satisfying the contradictory loadcases. The design of experiment (DOE), Kriging meta-model surface creation and optimization are done by GM internal tool. Totally, 60 DOE points were considered. Postprocessing scripts were developed and used to extract the results obtained from optistruct and dyna output files for each response.

3.3 Results and Discussion

Table 1 shows the comparison between baseline and optimized design summary. The proposed MDO process identified the balanced optimum design, which is meeting both loadcase performance requirements with 10% mass saving. Based on the results, it is clear that the proposed MDO process presented in the report is highly effective in identifying balanced optimized design for contradictory loadcase performance targets. The value of optimum design variable summary is shown in Fig. 7.

Table 1 Results summary of sensor bracket design using MDO methodology

Loadcase	Pedestrian protection					Body exterior		Mass saving	
Performance requirements	Tibia BM (Nm)	LCL (mm)	MCL (mm)	ACL (mm)	PCL (mm)	Horizontal angle (Degree)	Vertical angle (Degree)		
Target	<272	<10	<17	<10	<10	GM targets			
Baseline	285	12	13.2	16.4	2.3	32%	56%		
Balanced optimum design	192	4.60	4.03	2.63	1.13	66%	37%	-(10%)	

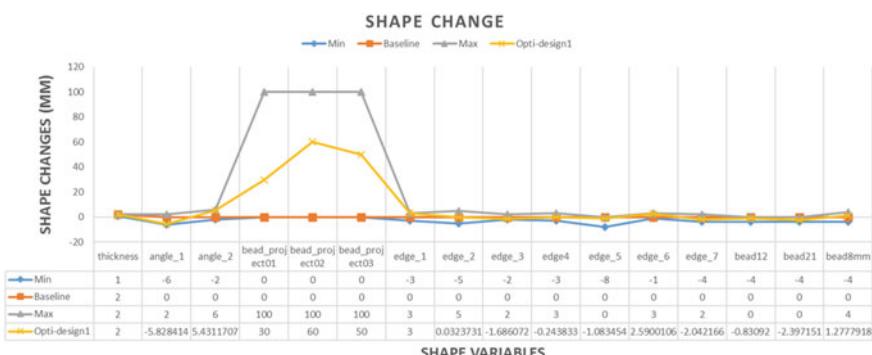


Fig. 7 Value of optimum design variable summary

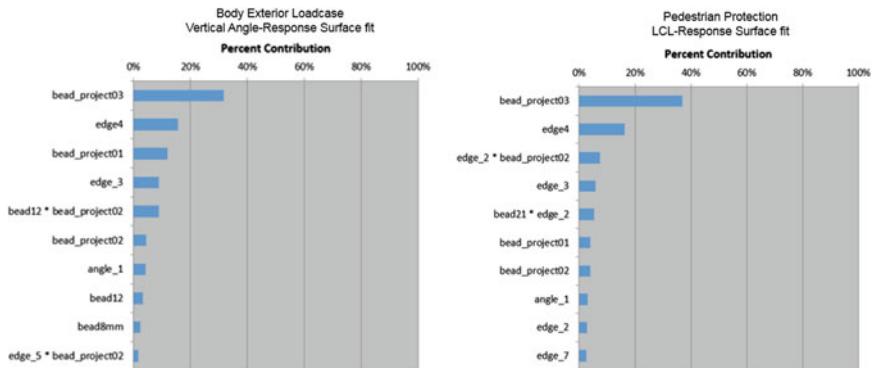


Fig. 8 Analysis of variance (ANOVA) plot

The analysis of variance (ANOVA) plot is shown in Fig. 8. Based on ANOVA, bead_project03 and edge4 variables are top two contributors for both body exterior and pedestrian protection loadcases. The next contributor for body exterior is Bead_project01 variable, whereas for pedestrian protection edge_2*bead_project02 variable.

4 Summary

Application of MDO process to arrive at optimized bracket design while satisfying the contradictory loadcases and also to minimize mass is demonstrated. Parametric design optimization with shape and gauge variables is considered. Free-size optimization is performed first to decide sensitive shape parameters which are then taken forward for MDO along with gauge variables. Suitable meta-modeling technique and optimization scheme are used to arrive at the optimal design. It is found that MDO process is able to perform well in two common design scenarios: (1) mass savings and (2) meeting performance target when the baseline design is not meeting the requirements. The optimization methodology can quickly explore large number of designs while balancing the conflicts/trade-offs between mass budget and performance targets. Process automation can further reduce the time of parameterized model creation. MDO approach can be observed as an imperative process for a development cycle as significant mass and time saving can be obtained which leads to reduced costs and better performance.

5 Acknowledgements

Inputs on the work during the initiation of the project from Simon Xu are acknowledged. The authors would like to thank Mr. Biswajit Tripathy for giving us the opportunity to work on this challenging project and to present this paper in the national conference on multi-disciplinary design, analysis and optimization.

References

1. NHTSA, On June 4, 2014, posted a report to Docket No. NHTSA-2008-0145 entitled, Technical Evaluation of the Flexible Pedestrian Leg Impactor (FLEX-PLI)
2. https://www.unece.org/trans/main/wp29/wp29wgs/wp29gen/wp29glob_registry.html
3. E.D. Morales, L.V. da Silva Macedo, S.F. Alexandre, *Optimization methods applied to development of vehicle structures*. SAE Technical pater, 2013-36-0455
4. B. Nima, *A new approach for sizing, shape and topology optimization*. SAE Technical pater, 960814
5. Altair Hyperworks user manual for free-size optimization
6. S.I. Yi, K. Shin, G.J. Park, Comparison of MDO method with mathematical examples. *Struct. Multidiscip. Optim.* **35**, 391–402 (2008)
7. S. Kodiyalam, *Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Phase I*. NASA Contractor Report, NASA CR-1998-208716, National Aeronautics and Space Administration (1998)
8. G.M.C. Adam, E.-G. Moustafa, *Front Underride Protection Devices (FUPDs): multi—objective optimization*. SAE Technical paper, 2015-01-1488
9. E. Ramesh, *Structural optimization simulation studies in new product development*. SAE Technical paper, 2007-01-4263
10. A. Ryberg, R. Domeij Backryd, L. Nilsson, Metamodel-based multidisciplinary design optimization for automotive applications. LiU-Tryck, Linköping, Sweden, LIU-IEI-R-12/003 (2012)
11. D.E. Malen, Fundamentals of automobile body structure design. In *SAE International*, Warrendale, PA, ISBN 978-0-7680-2169-1 (2011)
12. DEP Meshwork 7.22 User Manual. Detroit Engineered Product Inc (2014)

Kriging Driven Optimization of Vehicle Components Using Equivalent Static Load (ESL) Methodology



Kshitiz Swaroop and Varun Agarwal

Abstract Passenger safety requirements along with requirements for strength and rigidity lead to conflicting performance targets on automotive parts. Vehicle interior brackets are a prime example with a requirement of being soft to minimize head injury during impact, adequate stiffness during operation, and non-failure during abusive loadings. The problem is further compounded with the safety loadcase being a non-linear dynamic loadcase, whereas stiffness and abusive loading belonging to linear static domain. Multi-disciplinary optimization methods typically used to address such problems are likely to fail in cases of severe performance violation of the baseline configuration, since the solution may actually require a completely new load path or topology for the component to meet performance requirements. Non-parametric topology optimization methods can generate a new design with fresh load paths but are typically restricted to linear static loadcases. The problem of component design is addressed in this report by approximating the safety dynamic loadcase using the equivalent static load (ESL) methodology which generates equivalent static loading at multiple time instances of the impact. Since the performance of injury in the impact loadcase is measured using a complex head injury criteria (HIC) metric, the same is brought into the optimization framework by generating a Kriging response surface correlating the component strain energy at the different time steps of impact and the HIC criteria. A functional ANOVA is performed on the response surface to determine the weightage of HIC at different time steps. These weightages are used to construct a surrogate objective function in the ESL-based optimization to minimize HIC while meeting the abusive loading and stiffness performance criteria. This methodology is found to generate designs which meet the conflicting performance criteria serving as an excellent starting point for further iterations.

Keywords Kriging · Supervised learning · Equivalent static loading · Topology optimization · HIC safety

K. Swaroop · V. Agarwal (✉)
General Motors Technical Centre India, Bengaluru, India
e-mail: varun.agarwal@gm.com

K. Swaroop
e-mail: kshitiz.swaroop@gm.com

1 Introduction

Vehicle safety rating is increasingly an important consideration driving customer's vehicle buying decisions. This along with strength and rigidity requirements lead to increased challenges for vehicle designers to account for complex and contradicting performance requirements. This paper focusses on the challenges in meeting the performance requirements on vehicle interior components such as coat hook brackets. The brackets need to be resistant to abusive loading for durability, provide adequate stiffness during normal operations for customer satisfaction as well as be meet FMVSS201U USA federal safety regulations [1] for upper interior head impact protection. Parametric optimization methods such as multidisciplinary optimization have been used extensively to meet the required performance requirements by modifying the existing features of the design. Parametric methods though fail in cases of severe performance constraint violation where the solution typically lies in creation of new load paths possible only through non-parametric optimization methods which are infeasible for application to dynamic loadcases such as FMVSS201U head impact loadcases. Equivalent static loading, proposed by Prof Parks [2], provides a method for approximating explicit dynamic loadcases but allows for only linear static response as constraints like displacements and strain energy. The FMVSS201U has a defined requirement for head injury criterion (HIC) which is included in the optimization setup using the strain energy of the various ESL loadsteps and constructing a weighted response from the same using a surrogate constructed using Kriging. The proposed method is then demonstrated on a design of coat hook bracket for abusive loading and FMVSS201U requirement as compared to design just for abusive loading or using a simple average of ESL load step strain energies.

2 Overview of Loadcases and Requirements

Non-parametric optimization methods such as topology and topometry available within commercial solvers require gradients to drive the optimization restricting their applications to linear static analysis. Equivalent static loading (ESL) method proposed by Prof Parks allows the application of non-parametric optimization methods to non-linear dynamic loadcases by approximating the loadcase as a series of equivalent static loadcases derived from displacements at various time steps. The process involves extracting displacements at various time steps. The equivalent forces for the linear static model are then calculated for each time step to apply the displacements observed in the explicit analysis. A separate load step is calculated for each time-step, and the linear static model is taken for further optimization. On convergence, the derived topology is fed back to the explicit dynamic model and the simulation is run again. The displacements are again used to create new loadcases and loop continues till convergence. Interested readers may refer to reference [2] for further details. A schematic of the method is given under Fig. 1.

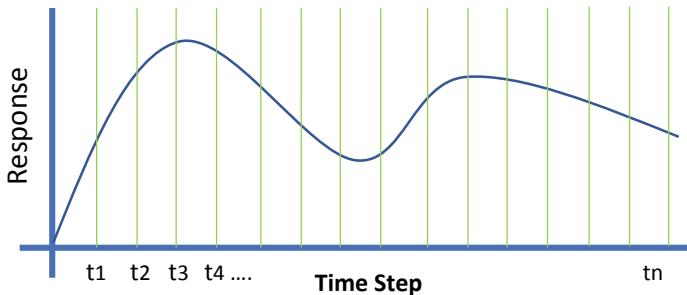
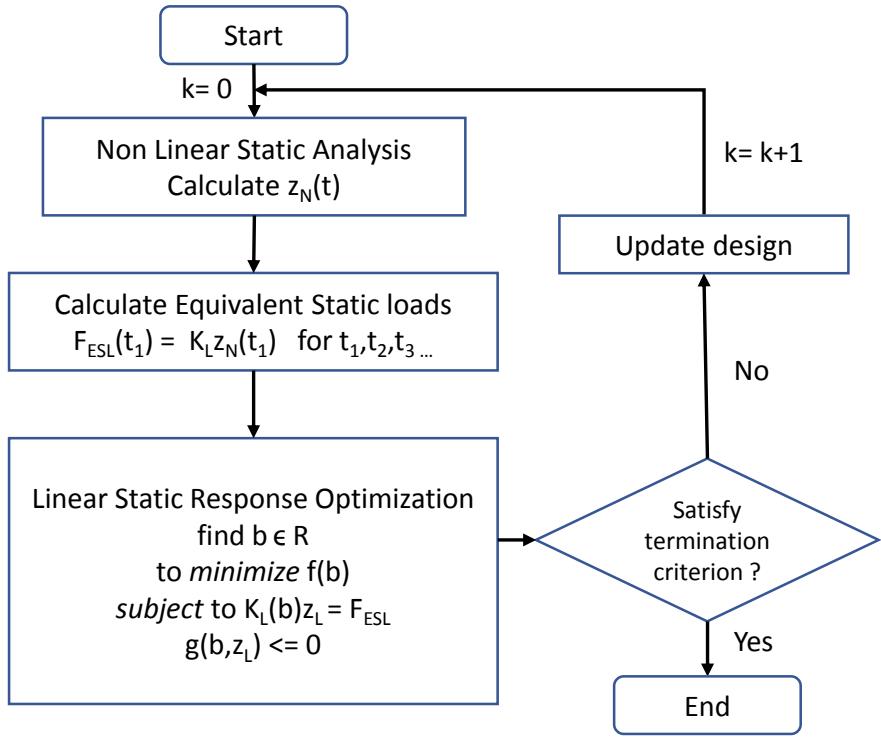


Fig. 1 Equivalent static loading schematic

Since the method approximates dynamic responses as independent load steps, the method can easily tackle performance constraints including intrusions (displacements) and even velocity-based performance constraints using finite difference between displacements at various time-steps and are available as constraints in commercial application of the methods.

The method poses some limitations in placing constraints on some injury performances and metrics such as head injury criterion (HIC) defined [3] as

$$\text{HIC} = \left\{ \left[\frac{1}{(t_2 - t_1)} \int_{t_1}^{t_2} a(t) dt \right]^{2.5} (t_2 - t_1) \right\}$$

where t_1 and t_2 are the initial and final times (in seconds) chosen to maximize HIC, and acceleration $a(t)$ is measured in standard gravity acceleration. The time duration, $t_2 - t_1$, is limited to a maximum value of 36 ms.

The HIC performance constraint cannot be inputted directly as a constraint or objective in ESL optimization.

- The first reason is being that only displacement and other linear static responses can be used as constraints in ESL-based optimization. HIC being a complex response derived from acceleration of the head form impacting the vehicle, the response cannot be computed directly from the linear static loadcases generated in ESL optimization.
- The second reason is due to the nature of the loadcase. Since, in the dynamic analysis, the head impactor is thrown at the vehicle, the vehicle and the impactor are not connected via structural members. Hence, in the linear static analysis, any update to the structure will not have any impact or change in the acceleration on the head impactor. The head impactor is typically grounded to zero displacement and does not show any displacement during the linear static iterations of ESL. All the updates or displacements are observed only on the vehicle structure. Even if equivalent displacements are calculated on the head impact, since there is no structural connection between the vehicle and head impactor, the two bodies will be independent in the analysis.

A surrogate or indirect response needs to be constructed from available responses from linear static analysis from the vehicle structure. A common approach is to maximize the sum of strain energy of various time steps [5, 6], and the reasoning being to generate the softest possible structure to minimize injuries. A novel approach for constructing a Kriging [7] surrogate from the strain energies of the loadsteps is proposed. A functional ANOVA [8] is performed on the generated surrogate to develop a weighted average of the strain energies to improve correlation between the surrogate objective and HIC criteria.

2.1 Problem Statement

The proposed approach is utilized to design a coat hook bracket for the FMVSS201U and abusive loading; a brief description of the two loadcases is given below.

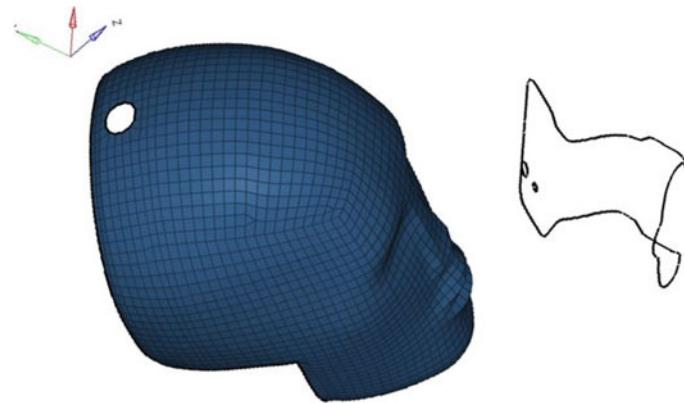


Fig. 2 FMVSS201U loadcase

2.1.1 FMVSS201U

The loadcase involves impacting a head form to the upper rail structure at the speed of 15 m/s as seen in Fig. 2. The performance requirement on the loadcase is to be lower than a given target value HIC_t (GM internal CAE target). Interested readers may refer to [4] for greater details about the loadcase setup in CAE.

2.1.2 Abuse Loading

A linear static unit load is applied on the bracket in the vertical direction as seen in Fig. 3. The performance requirement for the loadcase is to achieve a stress value lower than the stress target STR_T (GM Internal CAE target).

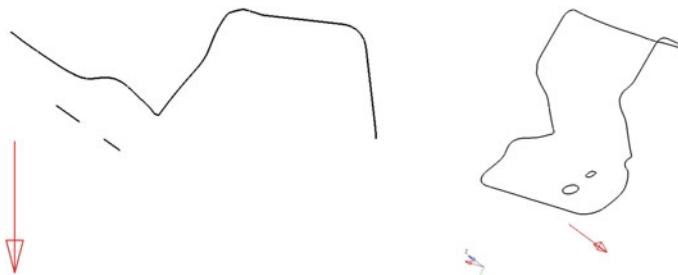


Fig. 3 Abuse pull loadcase

Table 1 Baseline performance and requirement

Loadcase	Baseline	Target
FMVSS201U	$1.23 \times \text{HIC}_t$	<HIC _t
Abuse	$0.86 \times \text{STR}_T$	<STR _T

2.1.3 Baseline Performance and Requirements

See Table 1.

3 Solution Framework

3.1 Equal Weightage to ESL Loads

The optimization problem is set-up as follows:

Objective: Maximize the sum of normalized strain energies of ESL Loads

Constraint: Stress < STR_T for pull load.

The base coat hook design is taken for topology optimization. Figure 4 compares the evolution of objective (sum of normalized strain energies of ESL loads) and HIC values corresponding to different iterations of topology optimization. The final output of topology optimization is shown in Fig. 5 which results in a design with the HIC (d) = $1.07 \times \text{HIC}_t$.

The above method reduces the HIC (from 1.23 HIC_t to 1.07 HIC_t) but there is poor correlation between the surrogate objective constructed from strain energies and the HIC of the simulation as seen in Fig. 4. The correlation is not directionally similar with a decrease in the objective not corresponding to a change in the HIC value. Hence, using equal weights is counterproductive as the optimization algorithm will take the solution in directions where objective is minimized but the HIC value may not be. A possible reason for the same is that the energy absorbed by the structure at different time steps has a different correlation and impact on final HIC

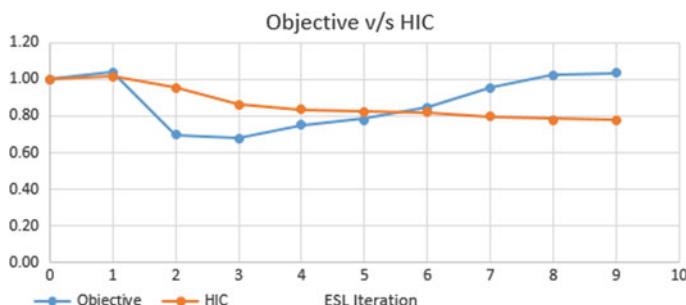
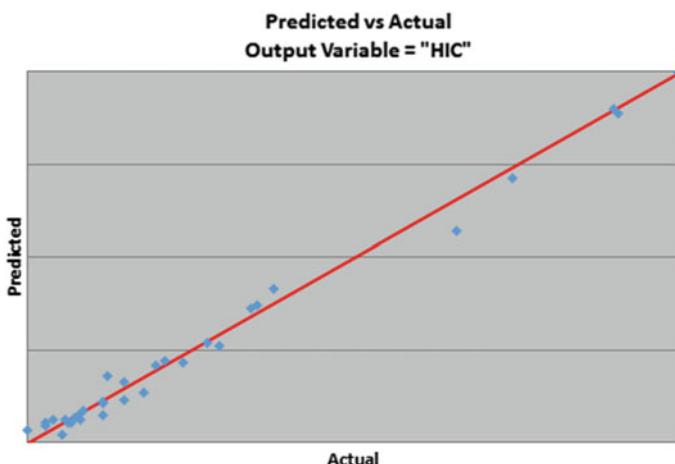


Fig. 4 HIC versus surrogate objective

Fig. 5 Realized design

value. As a possible improvement to this problem, a Kriging response surface is constructed on the ESL loadcase strain energies. This is to derive a weighted sum of the strain energies and see if the solution improves. A one-off cross validation is performed on the Kriging response which showed excellent correlation as seen in Fig. 6. A functional ANOVA is performed on Kriging response to calculate the relative importance of various ESL loadcases. The original ESL setup had more than 120 load steps derived at various time instances. Five of the load steps at various time instances show the maximum contribution to the final HIC value. It can reasonably concluded that only a few selected time instances are important in the final HIC value observed and using equal weightages for every time step is detrimental (Fig. 7).

The information is used to construct a weighted average strain energy response to show better correlation with HIC injury metric.

**Fig. 6** Cross validation

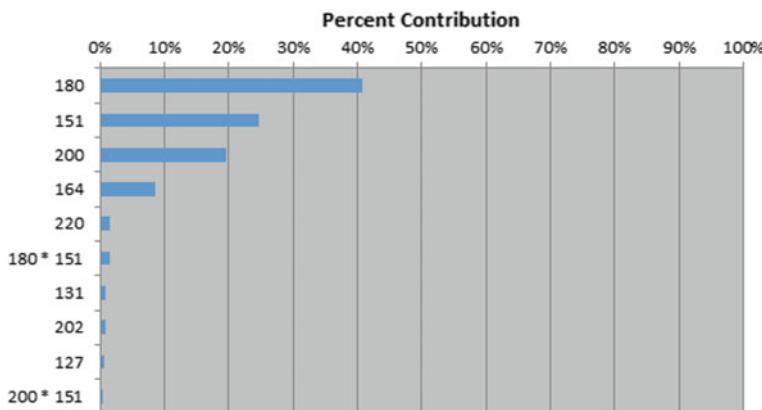


Fig. 7 ANOVA

3.2 Unequal Weightage to ESL Loads

The optimization problem is setup as follows:

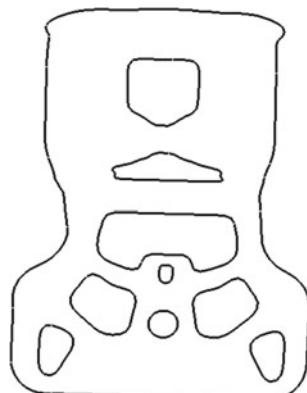
Objective: Maximize the sum of weighted normalized strain energies of ESL loads based on percentage contribution from ANOVA.

Constraint: Stress < STR_f for pull load

The base coat hook design is taken for topology optimization. The final output of topology optimization is shown in Fig. 8 which results in a design with the HIC (d) = $0.95 \times \text{HIC}_t$.

The optimization is rerun using the weighted average surrogate with the topology output showing an improved HIC (d) of $0.95 \times \text{HIC}_t$ compared to $1.07 \times \text{HIC}_t$, using simple average of strain energy. The realized topology shown in Fig. 5 is also significantly different from the one observed using simple-weighted averages (Fig. 8).

Fig. 8 Realized topology using weighted averages



4 Conclusions

A method to generate a surrogate for HIC injury metric from strain energies using Kriging is proposed. The method shows significantly improved results as compared to using simple average of strain energies and novel topology load paths proving the impact of the methodology. Most work on utilizing non-parametric method for crashworthiness include complex injury metrics using approximations and hence the approximations can only set as objectives since the actual injury metrics are not available. The Kriging response surface can compute the actual injury metric from the linear static approximation and be included in the optimization framework directly as a constraint or objective. This is especially beneficial since injury metrics typically have a requirement and any further improvement to the same is not beneficial, so a method to apply the same as constraint is beneficial. A similar structure can also be used for other complex injury metrics such in pedestrian protection such as Medial Collateral Ligament, Posterior Cruciate Ligament, etc.

References

1. Nhtsa.gov. Laboratory test procedure for FMVSS No. 201U (2019), <https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/tp-201u-02.pdf>
2. W.S. Choi, G.J. Park, Structural optimization using equivalent static loads at all time intervals. *Comput. Methods Appl. Mech. Eng.* **191**(19), 2105–2122 (2002)
3. J. Hutchinson, M.J. Kaiser, H.M. Lankarani, The head injury criterion (hic) functional. *Appl. Math. Comput.* **96**(1), 1–16 (1998)
4. B. Dennis, R. Sturt, Cae methods for simulating fmvss 201 interior head impact tests (No. 2001-06-0023). SAE Technical Paper (2001)
5. H. Müllerschön, et al., *Application of the equivalent static load method for impact problems with GENESIS and LS-DYNA. Automotive CAE Grand Challenge*. Hanau (2013)
6. S.I. Yi, J.Y. Lee, G.J. Park, Crashworthiness design optimization using equivalent static loads. *Proc. Inst. Mech. Eng. Part D J. Automobile Eng.* **226**(1), 23–38 (2012)
7. N. Cressie, The origins of kriging. *Math. Geol.* **22**(3), 239–252 (1990)
8. J. Kaufmann, A.G. Schering, *Analysis of variance ANOVA*. Wiley StatsRef: Statistics Reference Online (2014)

Reliability-Based Design Optimization of a Door Beam for FMVSS214 Requirements



Kshitiz Swaroop, Nihal Kakodkar, and Biswajit Tripathy

Abstract Door beams are the principal components used in vehicle doors to satisfy FMVSS214 side impact load case and meet associated performance criteria. The beams are typically designed with multiple tailor rolled blanks (TRB) to minimize mass. This poses challenge to the designer to optimize the section sizes as well as thicknesses of different regions while meeting various performance requirements. One such requirement for the door beam is not to undergo buckling at places other than ram engagement location (called end buckling) in addition to other requirements such as adequate load resistance. Machine learning-based techniques can be used to effectively predict subjective performances like end buckling. Physical properties of the door beam such as thickness and material property can vary around the mean resulting in variations in performance. This calls for reliability-based design optimization (RBDO)—designs which would meet the required reliability in performance when the input variables undergo random variations. This piece of work combines the machine learning methods used to predict performance along with reliability analysis tools to develop a RBDO framework which can predict reliability of a design and can also come up with a design to meet required reliability figures. The framework developed is applied to a simplified door beam, and the results are presented. Different supervised machine learning models are investigated for predicting peak load, average load, mass and end buckling. These techniques are integrated with particle swarm optimization (PSO) technique to perform design optimization. A reliability analysis is done using a Monte Carlo Simulation within the PSO. The RBDO framework can develop a Pareto front of door beam mass as a function of reliability, which can help the designer to select a door beam to meet required performance with least mass.

K. Swaroop · B. Tripathy (✉)
General Motors Technical Centre India, Bengaluru, India
e-mail: biswajit.tripathy@gm.com

K. Swaroop
e-mail: kshitiz.swaroop@gm.com

N. Kakodkar
Indian Institute of Technology, Kharagpur, India

Keywords RBDO · Supervised learning · Vehicle safety · Meta-heuristic optimization

1 Introduction

Advancements in manufacturing technologies have enabled tailor rolled blank (TRB) with continuous thickness transition through the structure. Although these TRB structures exhibit excellent crashworthiness with least mass, it is not easy to obtain the optimal thickness profile. A door beam is fitted to the inside of car door and is typically designed to provide adequate strength and stiffness against side impact. The door beam design needs to meet all the performance criteria for the FMVSS214 static load case [1]. In this load case, the door beam is typically loaded by a ram laterally, and it is required to meet certain performance targets like (i) peak load resistance and (ii) average load resistance over fixed time of loading and (iii) no buckling at places other than the center. In this work, various supervised machine learning models are trained as surrogates for the various performance requirements. Previously, a deterministic optimization using machine learning methods was undertaken to determine optimal design for door beam [2]. A deterministic design optimization does not account for the random variations that exist in modeling, simulation and manufacturing processes [3]. Examples are variations in material properties, sheet thickness and section sizes. A RBDO is preferred to a deterministic optimization since it leads to a more robust system after considering statistical distributions of design variables. Currently, a deterministic optimization requires an additional safety factor to ensure adequate margin in the presence of such variations, which leads to an over or under designed system. The target is to obtain a design to meet desired reliability figure rather than a design with an arbitrary safety margin. The reliability of responses, predicted by machine learning models in combination with Monte Carlo method, is posed as constraints with a particle swarm optimization scheme to obtain an optimal design with required reliability. The RBDO framework can develop a Pareto front of door beam mass as a function of reliability, which can help the designer to select a door beam to meet required performance.

2 Objective Function, Design Responses and Variables

The objective function is the mass of the TRB door beam which needs to be minimized while meeting desired reliability on performances such as (i) average load resistance, (ii) peak load resistance and (iii) buckling position (end or center) at 6 inches of deformation. The work by Sinha et al. [4] has shown that simultaneous shape and gauge optimization leads to better design exploration with much lesser mass figure as

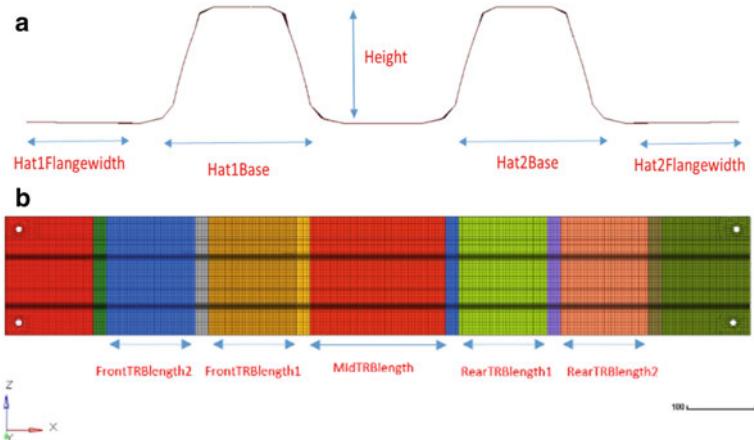


Fig. 1 **a** Shape variables. **b** Shape variables

compared to just gauge optimization. These two kinds of design variables: (a) gauge (thicknesses) variables for constant thickness zones (CTZs) and (b) shape variables, as shown in Fig. 1a, b are considered.

Material properties are assumed to vary with (i) Young's modulus between 178.5 and 241.5 GPa and (ii) the yield strength between -15% and $+15\%$. The two material properties can be simultaneously varied by varying the y scale factor of stress-strain graph between 0.85 and 1.15. While the values of gauge variables are absolute in mm, the values of shape variables are the deviations from baseline design after being scaled down by a factor of 5, units being mm. The ranges of the variables are defined according to engineering experience and the TRB manufacturing capacity. Table 1 lists all the variables and their typical bounds.

There are 21 independent design and material variables.

2.1 Design of Experiments (DOE)

We have 21 design variables, which have been considered having uniform distributions with an appropriate lower and upper bound for each variable. Latin Hypercube Sampling (LHS) [5] is used to generate a DOE of 1200 points in HyperStudy®. A summary of the DOE is provided in Table 2.

Table 1 Parameters of door beam

		Variable	Description	Lower bound	Upper bound	Baseline design	Variation
Design variables	Gauge variables	x_1	T_CenterBeam	1.5	3	1.6	0.1125
		x_2	T_FrontBeam1	1	3	1.6	0.15
		x_3	T_FrontBeam2	1	3	1.6	0.15
		x_4	T_FrontBeam3	1	3	1.6	0.15
		x_5	T_RearBeam1	1	3	1.6	0.15
		x_6	T_RearBeam2	1	3	1.6	0.15
		x_7	T_RearBeam3	1	3	1.6	0.15
Shape variables		x_8	Front Height	0	1	0	0.3
		x_9	Rear Height	0	1	0	0.3
		x_{10}	HAT1 Angle	0	10	0	0.3
		x_{11}	HAT2 Angle	0	10	0	0.3
		x_{12}	HAT Gap	-10	10	0	0.3
		x_{13}	HAT1 Width	-5	5	0	0.075
		x_{14}	HAT2 Width	-5	5	0	0.075
		x_{15}	HAT Flange Width	-2	2	0	0.75
		x_{16}	FrontTRBlength1	-2	2	0	0.75
		x_{17}	FrontTRBlength2	-2	2	0	1.5
		x_{18}	RearTRBlength1	-2	2	0	0.75
		x_{19}	RearTRBlength2	-2	2	0	0.75
		x_{20}	MidTRBlength	-2	2	0	0.3
Material variables		x_{21}	Y scale factor	0.85	1.15	1	0.15

Table 2 DOE Summary

Total no. of DOE points	1200
Mass range	3.1–6.5 kg
Peak load range	18,824–80,019 N
Average load range	12,566–48,659 N
No. of points with no end buckling	423
No. of points with feasible peak load	1172
No. of points with feasible average load	1000
Feasible DOE point with least mass (design 1)	3.41 kg
Peak load for design 1	26,849.6 N
Average load for design 1	19,234 N

2.2 Methods to Predict Responses

For predicting the responses, different machine learning techniques, suitable for supervised learning [7], were tried to fit data for each response:

1. Linear models such as linear/ridge/Bayesian ridge/least angle regression
2. Neural network
3. Support vector machines
4. Stochastic gradient descent
5. Ensemble methods such as Ada Boost, random forest, gradient boosting.

Different ensemble methods are used to predict nonlinear parameters—peak load, average load, end buckling since these methods provide better accuracy than linear regression, ridge regression, etc. In [6], they use an ensemble of four metamodels (artificial neural network, radial basis function, etc.) using weighted average method to improve R-squared accuracy. Accuracies for each model were assessed using root mean square error (RMSE), R-squared and mean absolute error (MAE). Good accuracy is reflected by remarkably lower RMSE and MAE and higher R-squared values. However, we do not use this technique since other methods provide a better accuracy as will be shown in the subsequent section.

Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees for training and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for habit of overfitting of decision trees [7].

To test their performance, the number of DOE points is split randomly into 80–20 ratio, out of which 80% DOE points are used to train these models and the rest 20% are used as test samples to check their accuracy of prediction. Then, this process is iterated 500 times, and mean accuracy is calculated to get a clear picture of prediction accuracy in accordance with the law of large numbers (LLN). The accuracy (R square) results are shown in Fig. 2, which is defined as follows and varies between 0 and 100%.

$$R - \text{squared} = \frac{\text{explained variation}}{\text{total variation}} = 1 - \frac{u}{v} = 1 - \frac{\sum(y_{\text{true}} - y_{\text{predicted}})^2}{\sum(y_{\text{true}} - y_{\text{true mean}})^2}$$

For the classifiers, the score parameter is used which returns the mean accuracy for the given test data and labels. Since there are only two labels, this is an appropriate metric.

Thus, the most appropriate methods for each response are found to be:

1. Random forest regressor [7]—for peak load
2. Gradient boosting regressor [8]—for average load
3. Linear regression [9]—for mass
4. Gradient boosting classifier [8]—for end buckling prediction.

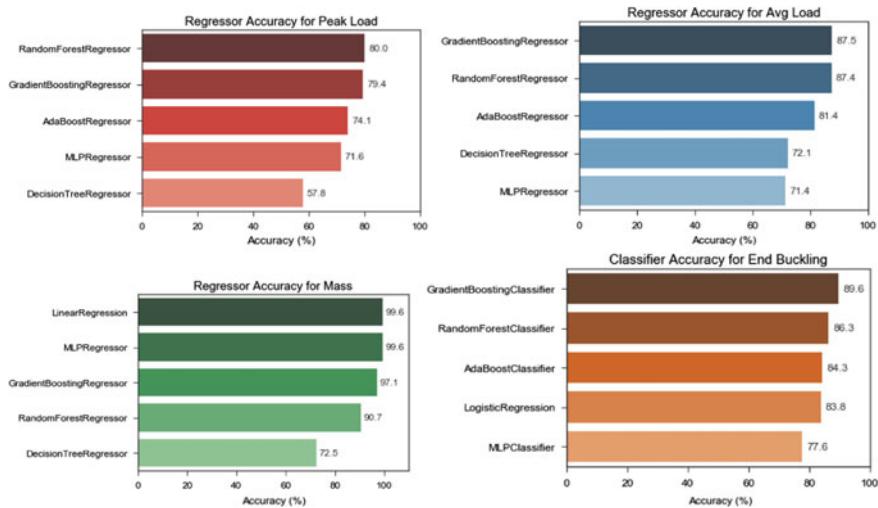


Fig. 2 Summary of accuracies

2.3 Proposed Framework for Reliability-Based Design Optimization (RBDO)

Particle swarm optimization (PSO) has been successfully applied in many different application areas due to its robustness and simplicity [3]. In comparison with other stochastic optimization techniques, PSO has fewer complicated operations and fewer defining parameters and can be coded in just a few lines. In Yang et al. [10], a decoupled method of integration is used in which solutions, found from the PSO, are fed back into the machine learning model (support vector machine) to be used as new training samples. The model is thus revised, and new solutions are obtained from the PSO, and thus, the method works like a feedback loop. PSO is used in this piece of work for optimization where the constraints include reliabilities for different performances (Table 3).

A comparison of different reliability analysis methods is made in [8], shown in Table 4. Monte Carlo Simulation (MCS) is used in this piece of work to estimate reliability.

Table 3 Accuracies for different responses

Output	1200 DOE (after optimizing parameters)
Peak load	Random Forest Regressor: 80.5%
Average load	Gradient Boosting Regressor: 90.1%
Mass	Linear Regression: 99.6%
End buckling	Gradient Boosting Classifier: 90.4%

Table 4 Reliability analysis methods

Reliability analysis method	Advantages	Disadvantages
Monte Carlo Simulation (MCS)	<ul style="list-style-type: none"> Most flexible as it does not rely on assumptions of the reliability constraint 	<ul style="list-style-type: none"> Requires a prohibitively large number of samples for evaluating small failure probabilities
Approximation methods (FOSM, FORM and SORM)	<ul style="list-style-type: none"> Much more time efficient 	<ul style="list-style-type: none"> Accuracy is doubtful when limit state functions are highly nonlinear and non-differentiable

The framework implemented (Fig. 3) used a double loop reliability analysis with the reliability constraint applied, using a Monte Carlo Simulation of (i) 100 points and (ii) 500 points, within the PSO framework. Once the optimal point was obtained, a MCS for 4000 points was done was a more accurate reliability value. The different values of probability were used to create a Pareto front.

Particle swarm optimization is used to minimize mass while applying six constraints:

1. Peak load > 22,241.1
2. Average load > 17,792.8
3. Probability of no end buckling > p_1 [$p_1 = \{0.6, 0.7, 0.8, 0.9\}$]
4. Reliability for peak load > p_2
5. Reliability for average load > p_2
6. Reliability for no end buckling > p_2 [$p_2 = \{0.5, 0.6, 0.7, 0.8, 0.9\}$].

Reliability > 0.9 means that if we vary each design variable and material property in a range of $\pm 15\%$ of its total range about the optimal point, then at least 90% of considered points must meet the constraint.

3 Optimization Results

A reliability-based optimization was done for the door beam model. Different reliability constraints were used (90–50%) to create a Pareto front to compare reliability of optimal points. The number of MCS points was taken to be 100 and then 500, and then, the results were compared to check which one gave better results and the difference in time taken for both per iteration. The results are shown in Table 5.

Figure 4 shows how the mass changes with reliability constraint for both MCS of 100 and 500 points. For the 100-point MCS at 70% reliability constraint, the average mass does not follow the trend since the PSO gives a different optima at each iteration, due to which the line may shift upwards or downwards.

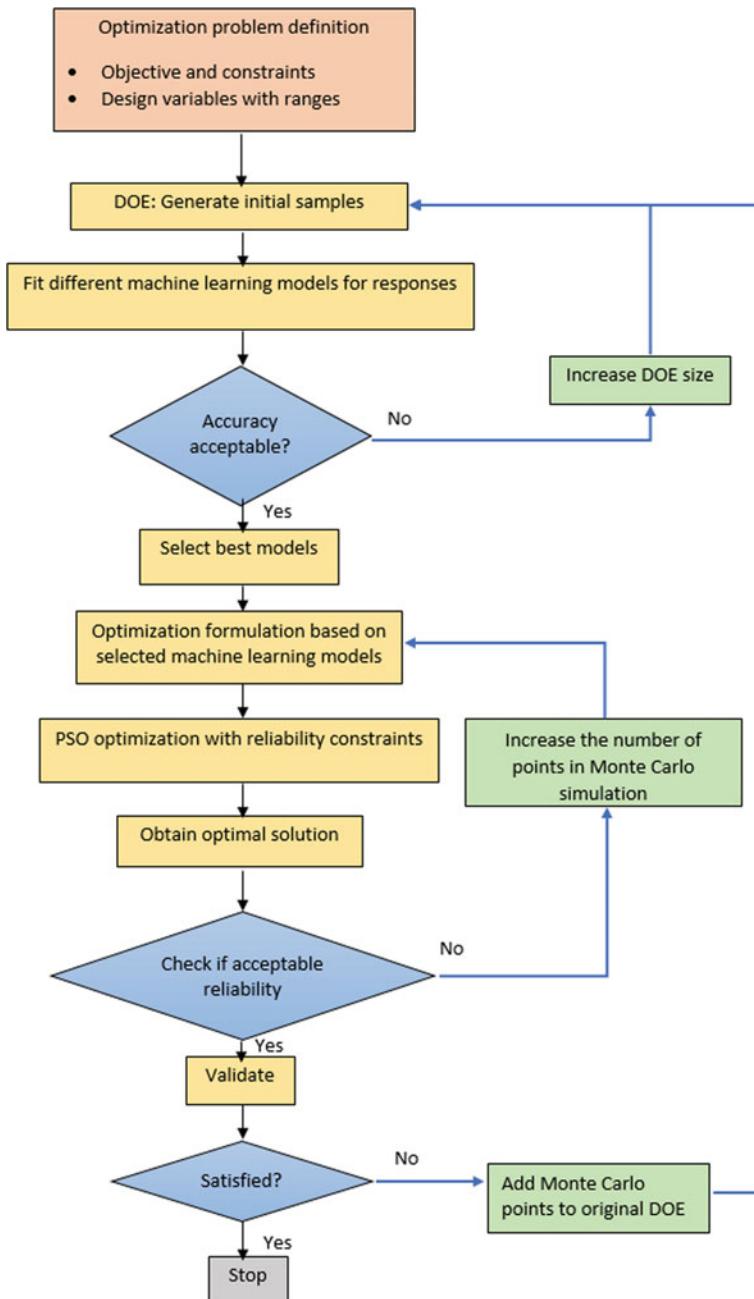
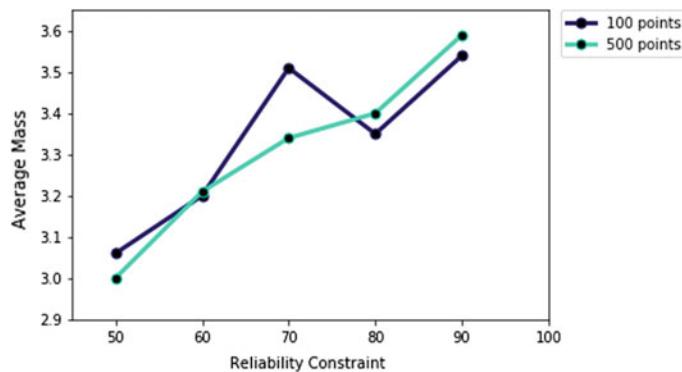


Fig. 3 RBDO framework

Table 5 Reliability for Pareto front

Response	Average reliability (%)										
	Reliability check	90%		80%		70%		60%		50%	
No of MCS points	100	500	100	500	100	500	100	500	100	500	
Peak load		100	100	100	100	99.5	100	99.9	99.5	98.2	99.99
Average load		92.5	91.4	78.9	86.9	67.3	68.8	61	59.5	55.3	58
End buckling		88.7	89.5	80	78.6	72.7	71.3	58.8	60.7	48.3	47.5
Mass (kg)		3.54	3.59	3.35	3.4	3.51	3.34	3.2	3.21	3.06	3

**Fig. 4** Pareto front for different reliabilities

4 Conclusion and Future Work

A RBDO framework is proposed which combines machine learning-based metamodels for performance prediction and PSO algorithm for optimization, along with reliability constraints estimated using Monte Carlo method. The framework when applied to reliability-based design optimization of a simplified door beam appears to work well. For a closer prediction of reliability values, the Monte Carlo points used for validating results can be added into the original DOE. Then, the models can be selected, and optimization process runs again. This helps to make the process adaptive which can result in giving a better accuracy to models as there may be places in the response surface where the original model is inaccurate due to lack of points in that vicinity in the DOE.

References

1. U.S. Department of Transportation National Highway Traffic Safety Administration Laboratory Test Procedure for FMVSS 214S (STATIC), Side Impact Protection, report TP-214S-05, 15 April 1992. [Online]. Available: <http://www.safercar.gov/DOT/NHTSA/VehicleSafety/TestProcedures/AssociatedFiles/TP-214-s05.pdf>
2. P. Chaurasia, K. Swaroop, B. Tripathy. *Machine Learning Based Method For Door Beam Optimization f2018/F2018-STN-011* (n.d.). Retrieved February 11, 2019 [Online]. Available <https://www.fisita.com/resources/papers/F2018/F2018-STN-011>
3. B. Dizangian, M. Ghasemi. Reliability-based design optimization of complex functions using self-adaptive particle swarm optimization method. *Int. J. Optimization Civil Eng.* (2015)
4. V.K. Sinha, S. Vanimisetti, B. Tripathy. Demonstration of a robust method for section and thickness optimization of automotive body structures using Optistruct, in *Altair Technology Conference India* (2015)
5. Latin Hypercube Sampling. [Online]. Available: https://en.wikipedia.org/wiki/Latin_hypcube_sampling
6. P. Zhu, Y. Zhang, G. Chen, Metamodeling development for reliability-based design optimization of automotive body structure. *Comput. Industry* **62**, 729–741 (2011)
7. L. Breiman. Random Forests. *Mach. Learn.* **45**(1), 5–32 (October 2001). <https://doi.org/10.1023/A:1010933404324>
8. J.H. Friedman. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **38**(4), 367–378 (February 2002). [http://dx.doi.org/10.1016/S0167-9473\(01\)00065-2](http://dx.doi.org/10.1016/S0167-9473(01)00065-2)
9. X. Yan, X.G. Su. *Linear Regression Analysis: Theory and Computing*. (World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2009)
10. I.-T. Yang, Y.-H. Hsieh. Reliability-based design optimization with cooperation between support vector machine and particle swarm optimization. *Eng. Comput.*, 151–163 (2013)

Design Exploration Using Unique Multidisciplinary Design Optimization Framework



Garima Singh and N. C. Praveen

Abstract Multidisciplinary design optimization (MDO) approach can be effectively used in the product design and development phase as it allows simultaneous exploration of design requirements of different disciplines. The MDO methodology can be adopted for design development in aerospace, automobile, and electronics domains. Altair Multidisciplinary Design Optimization Director™ (MDOD) is the unique framework developed to aid rapid set-up, execution, postprocessing, and design exploration of MDO problems. Altair's MDOD has three key differentiators, i.e., a user-friendly interface, automatic different discipline model synchronization, and reduction of computational effort using *Sub-Space* technology. This study intends to compare the advantages of utilizing this unique MDO framework in the design process over traditional MDO approach. In this study, Altair's MDOD suitability is assessed on an automotive chassis where thickness of the selected parts forms the design variables. It is a classic MDO problem where mass is minimized while maintaining performance requirements derived from different disciplines such as durability, stiffness, NVH, and safety. Altair MDOD utilizes Altair HyperStudy™ in the background to perform design exploration using design of experiments and meta-model-based design optimization (MBDO). All in all, Altair MDOD is best fit for performing MDO studies as it improves handling of large, complex data and aids rapid product development cycle by effectively integrating into the existing product development environment.

Keywords Multidisciplinary design optimization (MDO) · Meta-model-based design optimization (MBDO) · Multidisciplinary design optimization director (MDOD)

Garima Singh · N. C. Praveen
Altair Engineering India Pvt. Ltd., Bengaluru, India
e-mail: garima.singh2@altair.com

N. C. Praveen
e-mail: praveen.chandrashekariah@altair.com

1 Introduction

In a traditional CAE cycle of product development, there are repeated loops of parallel development phases and requires in-between synchronization within the different design groups. These individual group development cycles are driven by optimization activities and the feasible design obtained from the individual cycles may not be optimal which satisfies all the other relevant discipline design requirements [1]. Automotive companies perform MDO to identify an optimal design considering multidisciplines simultaneously, but these MDO methods need to effectively integrate into the product development cycle and that can be a challenging task [2]. Generally, first-level MDO is carried out at the concept level where there is a large scope of design change and as the design matures, the scope of design change narrows down and then, second-level MDO is performed at the final stages of the product development to fine-tune the design further. The MDO studies are generally carried out at a vehicle level or at a sub-structure level and are executed either using MBDO or single-level optimization methods [1, 2]. The MDO study is performed to achieve one of these goals, i.e., weight reduction or performance improvement or both. The goal of this study is to introduce the new and unique MDO framework to identify an optimal design considering two or more disciplines in conjunction and create a process which can be integrated effectively into the existing product development environment.

2 MultiDisciplinary Optimization Framework

The paper discusses the Altair's Multidisciplinary Design Optimization Director™ (MDOD) which is the unique framework developed to aid rapid set-up, execution, post-processing, and design exploration of MDO problems. Altair MDOD utilizes Altair HyperStudy™ to perform design exploration using Design of Experiments (DOE) and MBDO. The framework utilizes *Sub-Space* approach to solve the MDO problem which uses approximations built in sub-spaces of the design variables. Here, each approximation is built in the sub-space significant to the corresponding discipline while the optimization problem is solved in the full design variable space [3–5]. The *Sub-Space* approach helps in reducing the computational requirement as the approximations are built on sub-spaces that have reduced dimensionality while retaining the overall quality of the solution (as seen in Fig. 1). The success of the *Sub-Space* approach is dependent on the selection of the design variables considered for each of the different disciplines and thus, sensitivity studies for each of the discipline play a key role in selecting the relevant design variables. There is a challenge associated with the quality of the approximations as the number of training points needed to get a good quality approximation increases super linearly as the number of design variables goes up except in a case where the responses are linear [3].

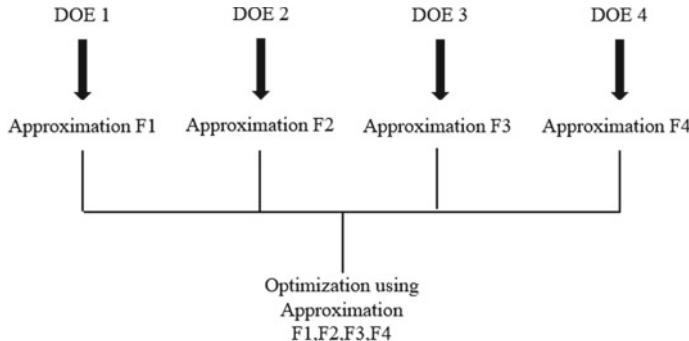


Fig. 1 Ideology behind *Sub-Space* approach where F1, F2, F3, and F4 approximations are built on sub-spaces that have reduced dimensionality while the optimization problem is solved in the full design space

2.1 Multidisciplinary Optimization Problem Formulation

Single discipline optimization problem considering single-objective is defined below using Eq. (1). The aim is to find optimal value of the design variable \mathbf{x} that minimizes the objective function f while meeting the constraints \mathbf{g} .

$$\begin{aligned} \text{Min } & f(\mathbf{x}) \\ \text{subject to } & \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \end{aligned} \quad (1)$$

The optimization problem will be termed as multidisciplinary if the design variables, objective function, and constraints influence different design attributes [1]. In this study, the optimization problem is single-objective, i.e., minimizes mass and is multidisciplinary as it considers gauges of 15 components from durability, stiffness, NVH, and safety disciplines as the design variables.

2.2 Traditional MDO Setup Versus MDO Problem Setup in Altair's MDOD

The most important step before solving MDO problem is the problem setup and it is the most complex and time-consuming step. The MDO setup comprises of the following steps:

- Defining the design variables and its linkage with the design variables from other disciplines
- Setting up design responses
- Defining the scope of the optimization problem.

In a typical vehicle-level MDO, 50–100 design variables form part of the problem formulation and these design variables need to be synchronized with the same design variables from other disciplines. The manual way of model synchronization is time consuming because finite element models from different disciplines are generally different in terms of mesh configuration and component naming/numbering convention. Model synchronization also requires defining symmetry conditions on design variables within one discipline model and then, applying the same condition in all the disciplines included. In a traditional MDO setup, the synchronization step is done without a visual interface, so, there are always chances of human errors while defining the linkages within and across the models. The next time-consuming step is to define design responses for all the disciplines and this need to be done in an effective way. Using the traditional way of setting up MDO, the optimization setup must be in full-space comprising of all the design variables for all the disciplines and this generally leads to increased computational run times. Also, all the included disciplines in the setup need to be part of the optimization problem formulation and cannot be excluded. Thus, there is no flexibility to perform an optimization on a selected set of disciplines and verify results on the disciplines which are not part of the optimization formulation. This also means that assessing optimization combinations such as durability with NVH, NVH with crash is not possible.

It has been observed that automotive companies perform MDO on NVH and crashworthiness disciplines [6–10] and then performs a manual overcheck on other disciplines. All the steps mentioned above are entirely dependent on an experienced engineer who sets up the MDO problem and once the manual MDO setup is done, it can neither be modified easily nor be re-used with a different optimization scope.

Now, the challenges faced during the manual MDO setup and its execution can easily be met using Altair's MDOD. Here, to demonstrate its usability, MDO study has been carried on an automotive chassis where 15 components were included as design variables. The MDOD framework comprises of five important stages as seen in Fig. 2 and the same has been discussed in detail in this section. Here, four disciplines (durability, stiffness, NVH, and safety) were included and in total, five models were setup (one for durability, stiffness and safety, and two models for NVH discipline). Here, NVH model was split into two models to aid the setup of normal modes and point mobility load cases.

Stage 1: Model import: In this stage, models are required to be imported in Altair HyperView™ and then, the models are scanned for designable properties such as mass, center of gravity, surface area, volume, thickness and the physical bounding box dimensions. In this study, 4 OptiStruct & 1 RADIOSS models were imported (as seen in Fig. 3) and were scanned for designable properties and this information



Fig. 2 Process flow depicting the five different stages inside the Altair's MDOD

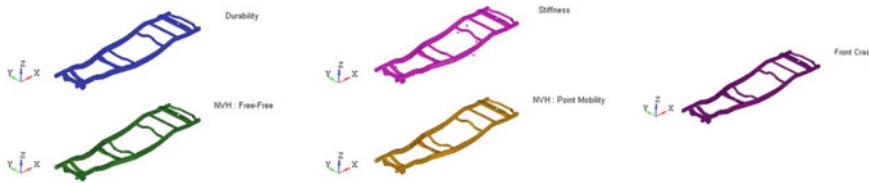


Fig. 3 Different discipline chassis models (total 5) imported in the MDOD in stage 1

served crucial for the model synchronization phase (stage 2).

Stage 2: Model synchronization: In this stage, 15 components (as seen in Fig. 3) were automatically linked across different disciplines unlike the traditional manual linking process. It is one of the key differentiators of the MDOD as the linking process is automatic, independent of numbering or naming conventions of the finite element models (relies on the scanning process done in stage 1). Additionally, symmetrical linking can be created within each model and the same can be applied to all other linked models automatically. The linking process has a visualization layer to review or change the design variable linking, unlike the traditional MDO method.

Stage 3: Optimization setup: In the MDOD, there are two ways of performing MDO i.e. i-MDO (iterative MDO) and a-MDO (automatic MDO). In i-MDO, single discipline optimization is performed, and the optimal values are over-checked in other disciplines included in the MDO setup. Whereas, in a-MDO, all the disciplines are included and a MBDO is carried out. The MDOD has a unique feature where i-MDO and a-MDO setup can be inter-changed and the same setup can be used in multiple ways depending on the project requirement.

Here, in this study, a-MDO was performed on all the four disciplines. The gauge design variables on the 15 components were setup in one go on the first four models (durability, stiffness, and NVH), whereas for the front offset safety model, design variables were defined separately on nine components which were the relevant ones in terms of frontal crash scenario. The two-step definition was carried out to setup the *Sub-Space* optimization approach inside the MDOD for reducing the computational expense. The design variables bounds were defined in terms of percentage, i.e., $\pm 15\%$ of the nominal values. Unique feature of the MDOD is that the design variable information can be exported in a text file format and can be imported later for setting up another MDO study on the same models. Later, design responses and their respective targets for each of the disciplines were defined using ‘import option’ (as seen in Fig. 4) which aids response definition using a text file.

Stage 3: Optimization execution: In this stage, MDOD creates parametrized template files for each discipline and a HyperStudy™ setup is automatically defined where the DOE runs needs to be manually executed. After DOE execution, integrity of the runs is assessed which checks for outliers, invalid, and bad values in the design space. In the next step, approximation model (fit) is created and its quality is assessed before running the combined response-surface-based optimization run (as seen in Fig. 5). Here, modified extensible lattice sequence (MELS) DOE method was chosen as it is an excellent space filling DOE scheme and has the property

Model label	Durability	Import...	Export...
	1:Modal Optimization		
TorsionD:TorsionD	>25.95		
BendingD:BendingD	>26.46		
Mass:Mass	min		

Model label	NVH_Free	Import...	Export...
	1:FreeFree Optimization		
VBending:VBending	>21.40		
FTorsion:FTtorsion	>25.70		
LBending:LBending	>38.54		

Fig. 4 Response definition and their respective targets for durability and NVH discipline in the MDOD

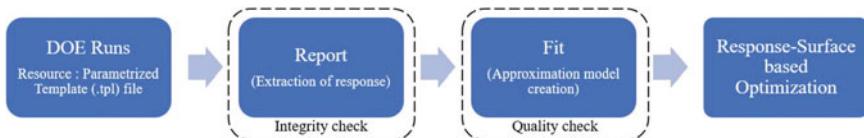


Fig. 5 Optimization execution process flow in the MDOD

of extensibility and adds more data points to provide equal converge. FAST Fit Automatically Selected by Training (FAST) fit method was used as it selects the best available fit (from available methods like least square regression, moving least squares method, HyperKriging, etc.) and automatically adjusts hyperparameters to avoid overfitting. By using *Sub-Space* approach over full-space approach, only 11 DOE runs were required for the safety discipline instead of 17 runs and this resulted into approximately 35% reduction in the computational run time (as seen in Fig. 6). In this study, fit quality was found to be moderate (~ 85%) and no DOE extension was required. In the MDOD, DOE extension is a one-click process and it creates a modified HyperStudy™ session with the previous DOE runs included automatically.

Under the optimization execution tab, different combination of optimization runs can be created such as durability with NVH, durability with stiffness, or NVH with crash (as seen in Fig. 7). Thus, giving a flexibility to evaluate individual effect of disciplines on the weight and performance. In this study, a combined optimization run (for all the four disciplines) was carried out. The validation panel in the MDOD automatically applies the optimal design variable values to the other discipline models which were not part of the optimization setup.

Disciplines/Models	Full-Space Approach		Sub-Space Approach	
	Dimension	Evaluations/Iteration	Dimension	Evaluations/Iteration
Durability	15	17	15	17
Stiffness	15	17	15	17
NVH_Free	15	17	15	17
NVH_MOB	15	17	15	17
Front Crash	15	17	9	11
Total		85		79

Fig. 6 Comparison of number of DOE runs between a Full-Space and Sub-Space approach setup

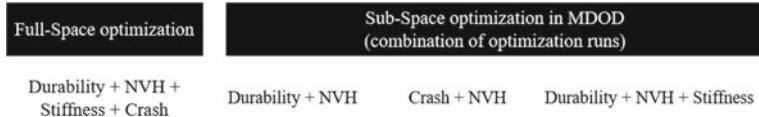


Fig. 7 Ability to perform combination of optimization runs using Sub-Space optimization approach in MDOD

Stage 4: Post-Processing: In this stage, results from the DOE, optimization, validation, and overcheck runs can be postprocessed in the MDOD without having to use any scripts. The results obtained from this study are discussed in the next section.

3 Results in MDOD and Summary

This section discusses the MDO results which were post-processed inside the MDOD and helped in acquiring further design direction. Figure 8 depicts the *optimization summary* which comprises of contour plot of property thickness, component masses, proximity of design variables to their bounds, and an overview of parallel coordinate plot (PCP).

Detailed MDO results are highlighted in this section where Fig. 9 depicts the final-optimized gauge obtained using *gauge contour plot* option and Fig. 10 depicts the mass summary for durability, NVH, and crash discipline obtained using *mass contour plot* option. Later, *DV bound* and *linear effects* options were used to find out if the optimization problem was over or under constrained and to find out the effects of design variables on output responses, respectively. Figure 11 clearly shows that the rails 3a–3b and cross_member 6–7 were hitting the upper gauge bound whereas the rail 1a, 1b, and 2b were hitting the lower gauge bound and cross_member 4–5 were found to be in the acceptable range. Figure 12 depicts the multivariate effects on design variables for stress response arising from Twist 2 g and Rear 3 g load case

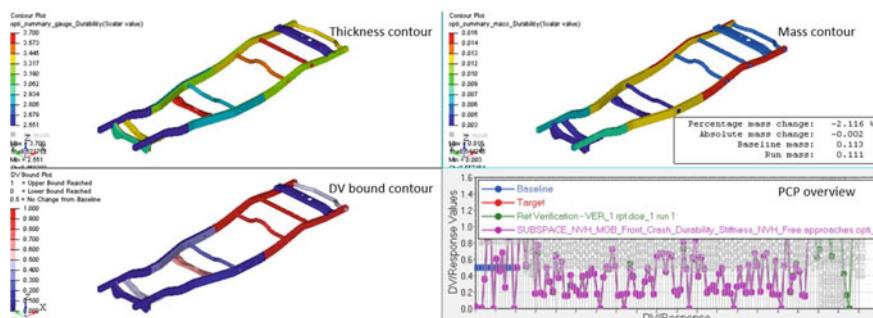


Fig. 8 Optimization summary in MDOD depicting contour plot of thickness, masses, DV bounds, and PCP

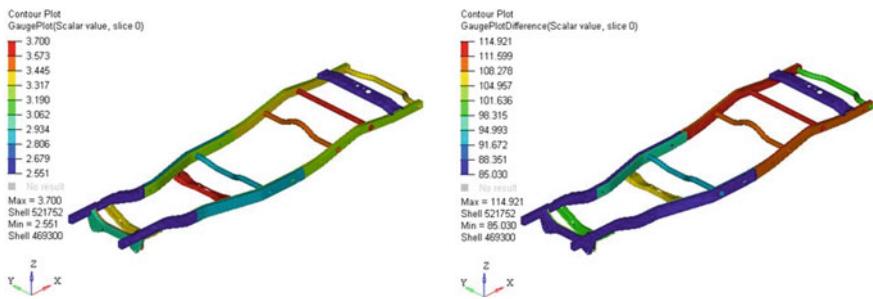


Fig. 9 Top left image depicts the gauge contour and top right image depicts the gauge in terms of % difference to baseline in MDOD. Here, value > 100 indicates increase in thickness and value < 100 indicates decrease in thickness

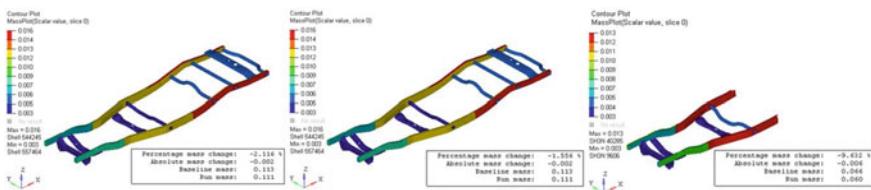


Fig. 10 Mass summary for the durability, NVH, and crash discipline (left to right) in MDOD

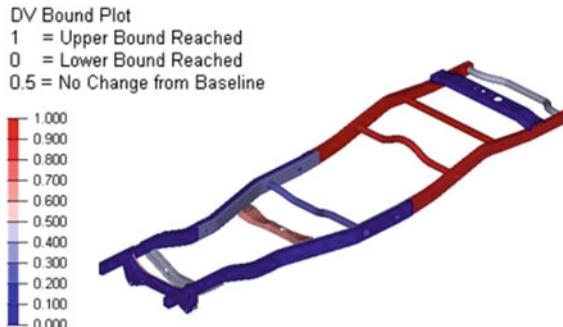


Fig. 11 DV bound post-processing feature indicating the gauges which are hitting the upper (red) and lower bound (blue)

indicating that increasing the highlighted design variable value improves the response and vice versa. Table 1 depicts the new optimal thickness values with respect to the baseline and the same has been visualized using the parallel coordinate plot feature as seen in Fig. 13.

From this MDO study, it was clear that trade-off would be required between mass and performance. Two scenarios were concluded, and they are as follows (see Table 2).

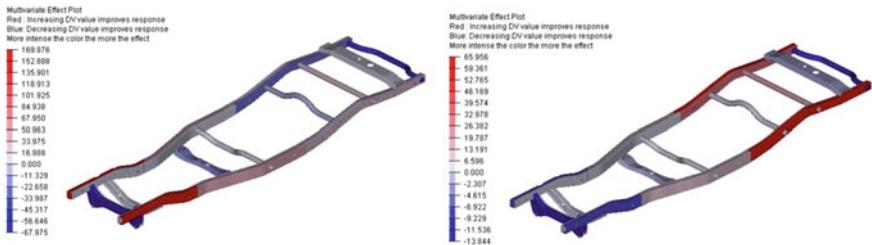


Fig. 12 Multivariate DV effects for two stress responses arising from Twist 2 g (left) and Rear 3 g (right) load case

Table 1 Table depicting the optimal design variable values versus the baseline values

Component	Thickness (Original)	Thickness (Optimal)
Railla	3.00	2.6
Raillb	3.00	2.6
Rail2a	3.40	2.9
Raillb	3.40	3.3
Rail3a	3.00	3.3
Rail3b	3.00	3.4
Crosss mem1	3.45	2.9
Crosss metn2	3.45	3.6
Crosss mem3	3.40	3.6
Crosss mem4	3.12	2.9
Crosss mem5	3.12	3.5
Crosss mem6	3.22	3.7
Crosss mem7	3.00	2.6
Crosss tnem8	3.40	3.4
Crosss mem9	3.40	3.4

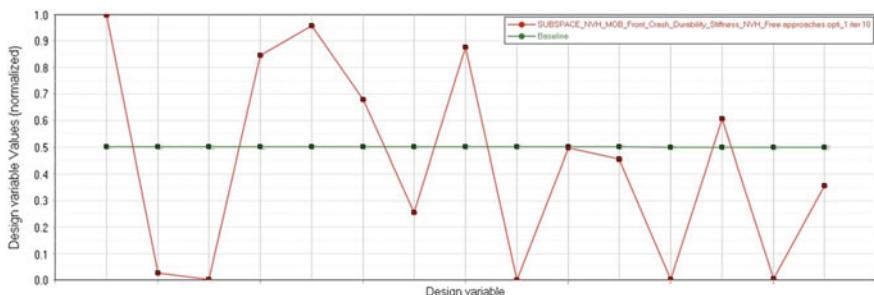


Fig. 13 Optimal design variable values versus normalized baseline values in a graphical representation using PCP in MDOD

Table 2 Scenario 1 and 2 concluded from the MDO study

Loadcase/scenario	Mass	Durability torsion mode	Durability bending mode	Stiffness stress front 2g	Stiffness stress front 3g	NVH vertical bending	NVH first torsion bending	NVH lateral bending	Front crash DISP 1	Front crash DISP 2Y	Front crash DISP 2Z	Front crash DISP 3Y	Front crash DISP 3Z	Front crash DISP 4Y	Front crash DISP 4Z	
Scenario 2	0.1795	26.6	27.2	412.1	415.6	228.1	222.2	26.2	39.0	356.1	190.8	236.1	176.1	78.3	159.5	134.4
Scenario 1	0.1728	26.1	26.5	376.9	388.1	264.4	21.4	25.9	38.9	296.2	237.5	199.9	231.6	67.7	215.9	121.3
Target		>259	>26.4	<465	<430	<430	>21.4	>25.7	>38.5	>311	<200	>198	<93	<182	<100	

- Scenario 1 where mass was slightly less than the baseline with improved durability, stiffness, NVH performance, and few violations arising from safety discipline.
- Scenario 2: mass was higher than the baseline with marginal improvement in durability, stiffness, NVH performance, and few violations arising from safety discipline.

4 Conclusion

The idea behind this study was to introduce the unique framework, for setting up and executing MDO problems, which can take out the extensive manual synchronization steps, reduces computational expense, and aids rapid problem setup. Using the **Sub-Space** approximation approach, 34% reduction in computational expense was achieved. The complete MDO problem setup from model synchronization, design variable creation, and target definition were done in 2 h compared to 12–15 h when done manually using traditional method. The demonstrated chassis results served just an example to highlight the usability and reach of the MDOD and do not represent the actual performance expected from the involved disciplines.

References

1. R. Domeij Backryd, A.B. Ryberg, L. Nilsson, Multidisciplinary design optimisation methods for automotive structures. *Int. J. Automotive Mech. Eng.* **14**(1), 4050–4067 (2017)
2. R. Domeij Backryd, A.B. Ryberg, L. Nilsson, A metamodel-based multidisciplinary design optimization process for automotive structures. *Eng. Comput.* **31**(4), 711–728 (2015). <https://doi.org/10.1007/s00366-014-0381-y>
3. J. Ollar, V. Toropov, R. Jones. Mid-range approximations in sub-spaces for MDO problems with disparate discipline attributes, in *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 16–20, 2014. <https://doi.org/10.2514/6.2014-2437>
4. J. Ollar, V. Toropov, R. Jones. Adaptive sub-space approximations in trust-regions for large scale MDO problems, in *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 1585 (2015). <https://doi.org/10.2514/6.2015-1585>
5. J. Ollar, V. Toropov, R. Jones, Sub-space approximations for MDO problems with disparate disciplinary variable dependence. *Struct. Multidisc. Optimization* **55**(1), 279–288 (2017). <https://doi.org/10.1007/s00158-016-1496-0>
6. K. Craig, N. Stander, D. Dooge, S. Varadappa. MDO of automotive vehicle for crashworthiness and NVH using response surface methods, in *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 5607 (2002). <https://doi.org/10.2514/6.2002-5607>
7. R. Yang, L. Gu, C. Tho, J. Sobieszcanski-Sobieski. Multidisciplinary design optimization of a full vehicle with high performance computing, in *19th AIAA Applied Aerodynamics Conference*, 1273 (2001). https://doi.org/10.1007/978-1-4615-0288-3_6
8. R. Yang, L. Gu, C. Tho, J. Sobieszcanski-Sobieski, Optimization of car body under constraints of noise, vibration, and harshness (NVH), and crash. *Struct. Multidisc. Optimization* **22**(4), 295–306 (2001). <https://doi.org/10.1007/s00158-001-0150-6>

9. A. Hoppe, M. Kaufmann, B. Lauber. Multidisciplinary optimization considering crash and NVH loadcases, in *ATZ/MTZ Virtual Product Creation* (2005)
10. F. Duddeck, Multidisciplinary optimization of car bodies. *Struct. Multidisc. Optimization* **35**(4), 375–389 (2008). <https://doi.org/10.1007/s00158-007-0130-6>

Model-Based Design and Multi-objective Robust Optimization of Electric Vehicle for Performance, Range and Top Speed Objectives



Aniket Karandikar , Ashok Kumar Ravi , Elavarasan Dharumaseelan , and Elanchezhian Tamilselvam

Abstract This paper deals with model-based design and robust design optimization of a compact electric vehicle (EV) for achieving better vehicle range for the given performance, top speed and load requirements. An Indian compact diesel hatchback was taken as reference vehicle to start the design process for the EV. Detailed battery, motor and simple longitudinal vehicle models were built using Modelica language platform, and pre-sizing of the components were done based on the reference vehicle characteristics and required EV objectives. A hybrid optimization algorithm was used to find optimal sizing parameters of EV components that can provide maximum vehicle range and minimum mass satisfying all the constraints.

Keywords Electric vehicles · Model-based design · Modelica · Robust design optimization · Response surface models · Uncertainty quantification · Polynomial chaos expansion

1 Introduction

The world is moving away from fossil fuels and toward electrification in mobility sector for promoting low carbon emissions and cleaner environment. Like the state-of-the art internal combustion engine vehicles, EVs also employ multi-disciplinary

A. Karandikar ()
ESTECO Software India Private Limited, Pune, India
e-mail: karandikar@esteco.com

A. K. Ravi · E. Dharumaseelan · E. Tamilselvam
Modelon Engineering Private Limited, Trichy, India
e-mail: ashokkumar.ravi@modelon.com

E. Dharumaseelan
e-mail: elavarasan.dharumaseelan@modelon.com

E. Tamilselvam
e-mail: elanchezhian.tamilselvam@modelon.com

components to deal with growing customer and legal objectives like higher performance, higher range, greater top speed, lower running cost, etc. Some of these objectives contradict with each other like higher performance and lower running cost. This suggests the use of model-based systems engineering approach for design of electric vehicles from concept to design verification stage so that multi-disciplinary and multi-objective problems of electric vehicle development are systematically tackled. In this work, model-based design (MBD) and multi-objective robust design optimization (MORDO) of an electric vehicle for performance, range and top speed were done by satisfying constraints such as vehicle mass, battery voltage, battery temperature, motor max. speed. An Indian compact diesel hatchback was taken as reference vehicle, and the EV components were pre-sized accordingly like the approach mentioned in the reference paper [1]. Vehicle system model based on Modelica language and Modelon libraries [2] were created and used to validate the initial sizing of the EV components. The optimal sizing of the EV components is found using a commercial optimization tool, modeFRONTIER [3]. The Modelica model was converted into a functional mockup unit (FMU) and run together within the optimization tool using packages like PyFMI [4].

2 Model-Based Design of the Electric Vehicle

As stated above, initial design of the electric vehicle was done based on an Indian 1.5 L diesel hatchback as reference vehicle. Models were involved during pre-sizing as well as validation of the pre-sizing parameters.

2.1 Component Selection and Pre-sizing of EV

The reference vehicle was modeled, and the chassis load and tractive force at wheel in each gear were noted. DC electric machine and single gear transmission were chosen and sized to approximately match the reference vehicle maximum tractive force and speed characteristics. Series/parallel battery pack configuration was chosen along with suitable cooling system that was sized according to the given load, mass and temperature requirements.

The initial sizing was arrived based on reference vehicle characteristics, authors' experience in the field, literature [1, 5, 6] and market study on similar class of vehicles. The initial sizing is mentioned in Table 1. These initial sizing were verified using a longitudinal vehicle model against the following vehicle objectives which is explained in Sect. 2.2. Objectives are mentioned in Table 2.

Table 1 Initial sizing of EV components

Parameters	Values
Motor max. power (P_{\max})	70 kW
Motor max. torque (τ_{\max})	± 240 Nm
Transmission gear ratio (ratio)	8.75
Single cell capacity (Q_{cell})	50 Ah
Single cell max. OCV (V_{cell})	4.2 V
No. of series cells in the pack (n_s)	100
Number of parallel cells in the pack (n_p)	2
Cooling system pump speed (N_{pump})	2000 rpm
Cooling radiator width (y)	0.35 m
Vehicle wheel radius (r)	0.3 m
Vehicle coefficient of drag (C_d)	0.32

Table 2 EV objectives

Parameters	Values
Drive cycle	Modified Indian
Performance	0–80 kmph in 10 s
Top speed	120 kmph
Range	As max. as possible

2.2 Vehicle Model

Simple longitudinal vehicle model was created to evaluate the given EV objectives for the selected components and sizing parameters. Right level of component details was employed for the given problem, and Modelica language was chosen as modeling platform. Modelica is an open standard and suitable for multi-physics modeling. The component models were adapted from Modelica-based Modelon libraries. Both the system and component models are scalable, configurable and reusable, and the model complexities can be extended based on the objectives under study. The vehicle model consists of simple driver, DC machine, single gear transmission, chassis, battery pack, cooling system, DCDC converter and fixed power load. The individual component models are explained in the subsequent sections. A schematic of vehicle model is shown in Fig. 1.

Simple Driver. The driver model is made up of simple PID controller determining torque required to follow the given drive cycle that has built in performance and top speed curves. Modified Indian drive cycle is used in this study for range calculation, and the target performance and top speed values were embedded along this drive cycle at the initial section. For range calculation, this initial section is ignored and only drive cycle period is used.

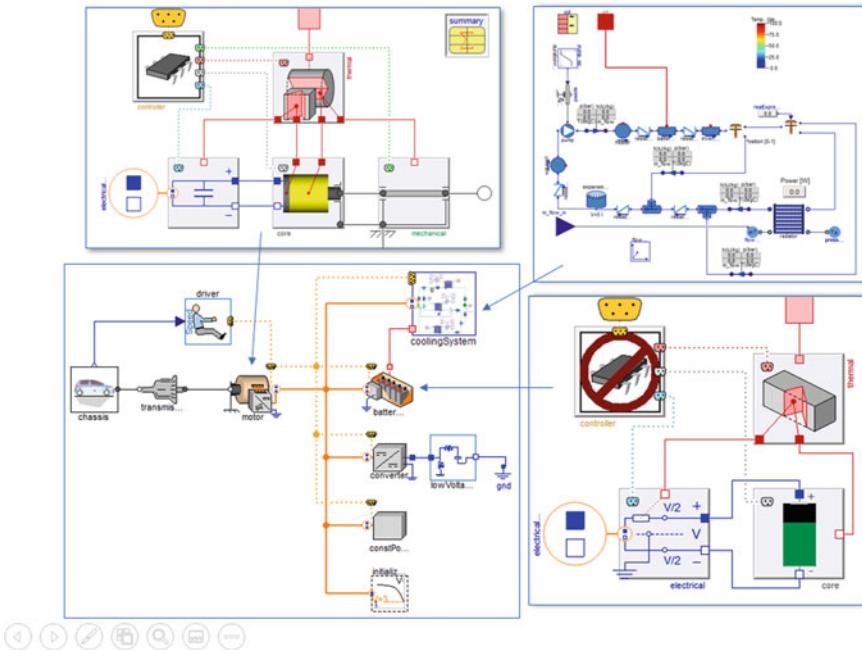


Fig. 1 Vehicle, motor, battery and cooling system schematics

DC Machine and Single Gear Transmission. DC machine is modeled with ideal limits taking peak torque and peak power as inputs. The machine is controlled by a torque controller. The reference torque is limited by the available torque and maximum rated power. The machine can regenerate during deceleration maneuvers. The machine losses are modeled as torque, speed dependent and variation based on motor power also captured. Change in mass due to change in motor size is captured in Chassis component. Simple lumped thermal model is employed and connected to ambient for cooling. Transmission is modeled as ideal gear with single transmission ratio. Inertia losses are ignored for simplification.

Chassis. Chassis is modeled as simple resistance considering tire rolling resistance, aerodynamic and gradient resistance. Vehicle mass capable of longitudinal motion is modeled as an inertia. Chassis mass is inclusive of individual component masses like battery, DC machine and cooling system. Whenever the size of these components change, corresponding mass change is reflected on Chassis to enable detail study on relationship between vehicle mass and vehicle range.

Battery Pack. Battery pack model is generic consisting of series and parallel cell configurations. Cell capacity was modeled as ideal and fixed. Cell open circuit voltage was modeled as temperature and state of charge (SOC) dependent. Battery impedance

is modeled as first-order RC loss component based on battery temperature and SOC. Lumped thermal model is incorporated to find dynamic battery temperature. Change in mass due to change in battery capacity is captured in Chassis component. Battery management system is not used in this study for simplification.

Battery Cooling System. Dynamic model of a liquid cooling circuit was included as cooling system. The battery coolant flow (ethylene glycol and water mix) is driven by an electric pump connected to the main battery. A radiator with thermostatic bypass valve cools the liquid coolant. The thermostatic valve is controlled by hysteretic temperature control. Airflow to the radiator is supplied through the vehicle front grill, and it is vehicle speed dependent. The radiator is modeled as heat exchanger with liquid in horizontal microchannel flat tubes. This model is geometry based, and overall radiator width (liquid side) is used in the optimization study as cooling parameter. Effect of change in radiator width in mass is captured in Chassis.

2.3 Model-Based Validation of the Initial EV Sizing

As stated in Sect. 2.1, the initial sizing of the components was validated against the given objectives by simulating the vehicle model over the given drive cycle repeatedly. We started off the simulation with 100% battery SOC and terminated the simulation when battery reached critical SOC of 20%. The vehicle range attained is noted and passed onto the optimization. The following are the results of this validation study.

From the above simulation using initial electrical vehicle sizing parameters, the obtained vehicle range was 139.40 km and the overall vehicle mass was kept as 1427 kg. In the following section, same vehicle models are used in optimization tool in the form of Functional Mock-up Interface (FMU) to find robust optimal designs for the sizing parameters that can maximize vehicle range while satisfying the constraints given such as vehicle mass. The results obtained for initial sizing are shown in Fig. 2.

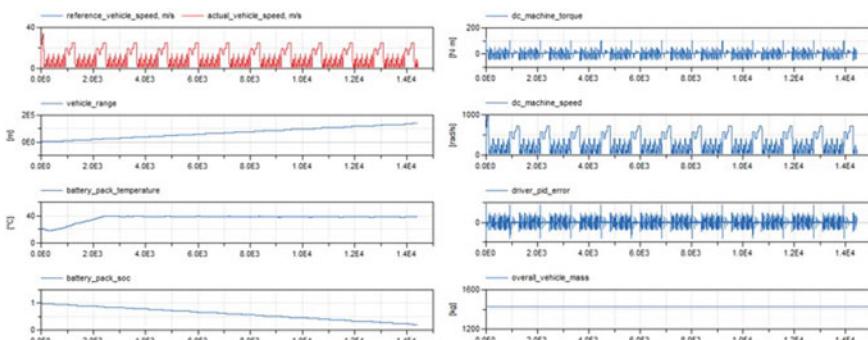


Fig. 2 EV results for initial sizing

3 Design Optimization

In the current computational model of EV, the total number of cells in the battery and radiator width determine the mass and hence the cost of the driveline. The vehicle range is directly correlated to the total number of cells and the configuration of battery. More batteries must be used to increase the vehicle range which increases the vehicle mass and hence the cost. Hence, electric vehicle driveline configuration must be designed by solving a multi-objective optimization problem of maximizing the range and minimizing the mass at the same time. These two objectives contradict against each other, and the solution of this problem results in form of a Pareto front showing this trade-off. The aim of the current study is to understand the trade-off between vehicle mass and vehicle range for different component configurations including uncertainties in input variables. A constrained multi-objective optimization problem is formulated with component configuration parameters as inputs varying in their respective ranges as shown in Table 3. The problem consisted of both discrete and continuous variables. The problem was solved using two approaches, namely deterministic optimization and robust optimization. The details of the approaches are discussed in the following sections.

In the optimization tool, FMUs are integrated and run in automation. For every iteration, FMU is run for the entire drive cycle and responses are recorded as time series vectors. For every iteration, the vector responses recorded are vehicle range, vehicle mass, battery voltage, battery temperature and controller error. Maximum and minimum values of battery voltage and controller error are calculated. Maximum values of vehicle range and battery temperature are calculated using inbuilt calculator facility of optimization tool.

3.1 Design of Experiments (DOE)

A design of experiments (DOE) was evaluated to explore the design space. The purpose of evaluating a DOE was to evaluate sufficient design points which can be used to train response surface models.

Table 3 Variable with their upper bounds, lower bounds, type and standard deviation

Variable	Lower bound	Upper bound	Type	Standard deviation
Motor power (W)	30,000	70,000	Continuous	700
Gear ratio	4	11	Continuous	0
Number of cells in parallel	1	5	Discrete	0
Number of cells in series	50	200	Discrete	0
Radiator width (m)	0.35	0.5	Continuous	0.002
Motor torque (Nm)	75	240	Continuous	1

Uniform Latin Hypercube (ULH). The initial DOE was generated using uniform Latin hypercube algorithm [7]. Uniform Latin hypercube uses a constrained Monte Carlo sampling to randomly populate the design space and solves an optimization problem to minimize the correlations between input variable as well as to maximize the Euclidian distance between generated designs to achieve uniformity in design space exploration. DOE of 60 designs generated using ULH was used for the first iteration of RSM training and validation. The uniformity of generated DOE was measured using Pearson correlation coefficients, and the results are shown in Fig. 3 [8]. In Fig. 3, a scatter matrix chart is shown where variables are represented in a matrix form. Each element in the lower triangle of matrix represents Pearson correlation coefficient (PCC). For a perfectly distributed DOE, PCC should be zero. However, values approximately equal to zero can be considered acceptable for a space filler algorithm. The upper triangle of the matrix represents 2D scatter plots for two corresponding variables, and diagonal elements represent probability density function for each variable between its defined range. It was observed that ULH algorithm was able to generate a uniform distribution even in the presence of both discrete and continuous variables.

Incremental Space Filler (ISF). Incremental space filler algorithm present in the optimization tool is developed to add additional points to the existing DOE [3]. New design points are sequentially added to the database by maximizing the minimum distance between the existing and new design point. Genetic algorithm is used to solve this optimization problem within ISF. Additional design points generated using ISF are added to the datasets used for RSM training and validation which improves the accuracy of prediction for RSMs.

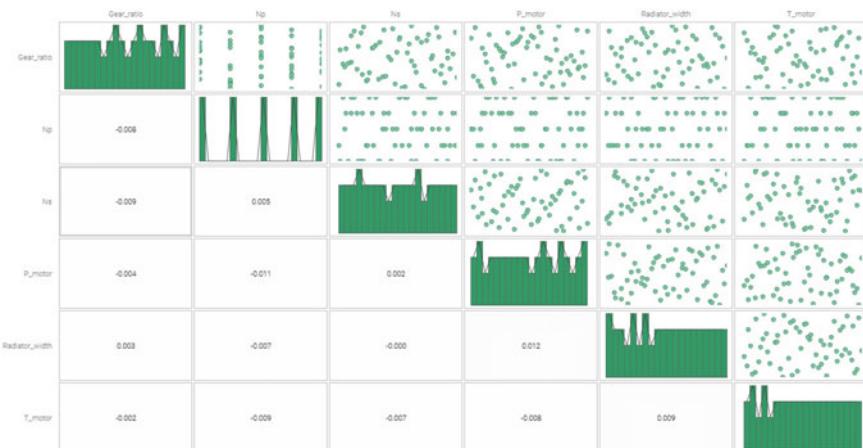


Fig. 3 Scatter matrix for initial DOE

3.2 Response Surface Model (RSM) Development

Heuristic optimization algorithms often require significant number of iterations to solve a multi-objective optimization problem. It may be practically unfeasible to run the numerical model for hundreds or thousands of iterations. Response surface methodology can be used to develop data driven functions between input variables and responses. Response surface models (RSMs) with sufficient accuracy of prediction can be used to replace the numerical model.

A flowchart for RSM training is shown in Fig. 4. A DOE of 60 designs generated using ULH was used as initial dataset. Forty-eight designs (80% of the dataset) were used for initial RSM training. Remaining 20% of the designs were used for RSM validation. The selection of 20% designs was done randomly. RSMs were trained for all outputs using multiple algorithms simultaneously, and the algorithm giving the least prediction error was selected. Mean normalized error was considered as the selection criterion.

Forty designs were added to the initial DOE using ISF to improve the prediction accuracy of RSM. RSM with mean normalized error less than 2% was selected after the iterative process. The details of selected RSMs for each output are given in Table 4.

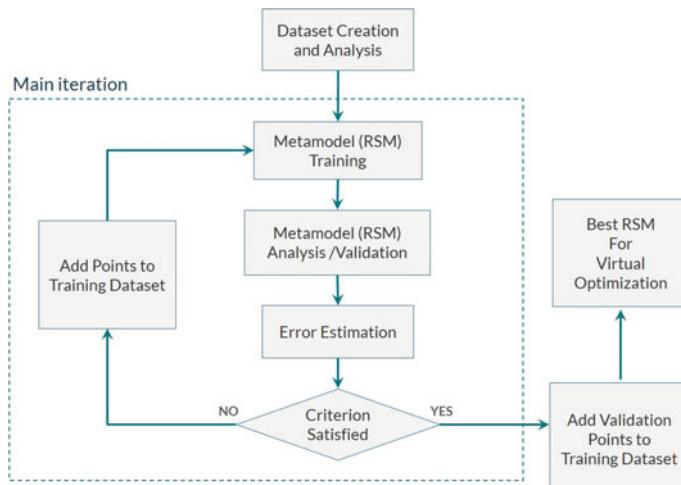


Fig. 4 Flowchart of iterative RSM training process

Table 4 Response surface models

Variable	RSM algorithm	Mean normalized error (%)	R-squared
Vehicle range	Gaussian processes	0.846	0.995
Battery voltage	Neural networks	0.5	1
Vehicle mass	Polynomial SVD	~ 0	1
Controller error	Neural networks	2.06	0.989
Battery temperature	Anisotropic Kriging	1.27	0.98

Table 5 Objectives and constraints

Method	Deterministic optimization		Robust optimization	
Variable	Objective	Constraint	Objective	Constraints
Controller error (m/s)	–	0.3 < nominal value < –0.3	–	0.3 < ($\mu + 3\sigma$) < –0.3
Battery voltage (V)	–	245 < nominal Value < 410	–	245 < ($\mu + 3\sigma$) < 410
Battery temperature (K)	–	nominal value < 318	–	($\mu + 3\sigma$) < 318
Vehicle mass (kg)	Minimize nominal value	nominal value < 1350	Minimize $\mu + 3\sigma$	($\mu + 3\sigma$) < 1350
Vehicle range (km)	Maximize nominal value	nominal value > 100	Maximize $\mu - 3\sigma$	$\mu > 100$

3.3 Deterministic Optimization

A deterministic approach was considered to solve the optimization problem explained in Sect. 3. For deterministic optimization, standard deviation was assumed to be zero for all variables; i.e., no random variation is present in any variable. The variables are taken at their exact deterministic values during the iterative optimization process. The objectives were to maximize the range and minimize the mass. The constraints applied in the optimization process are explained in Table 5.

3.4 Robust Design Optimization

In practical conditions, all continuous variables randomly deviate away from their designed values due to randomness involved in the manufacturing and assembly processes. This random variation cannot be avoided; however, it can be quantified in terms of a probability density functions. Probability density functions can be obtained by measuring the manufactured samples. However due to lack of availability of such data in literature, certain assumptions are made to define the probability density functions in the present study. In this study, random variations in the continuous

variables, namely motor power, motor torque and radiator width, were considered. For each variable, the probability density function was assumed to be equal to that of a normal distribution. Standard deviations of these normal distributions were assumed, and their values are mentioned in Table 3.

Due to random variations present in input variables, responses also exhibit a random variation. Such random variations in responses affect the performance. In such cases, even an optimized design may exhibit poor off-design performance. Uncertainties present in the responses needs to be quantified and formulated in objective functions and constraint functions in terms of their respective mean values (μ) and standard deviation (σ) values to avoid designs with poor off-design performance [9]. Formulation of objectives and constraints for robust optimization is given in Table 5. In this study, polynomial chaos expansion is used as the uncertainty quantification method.

3.5 Polynomial Chaos Expansion (PCE)

Polynomial chaos expansion is a technique to estimate the mean and standard deviation of a sample set [10]. The samples were generated using Latin hypercube sampling method. In polynomial chaos expansion, the uncertain variable are represented as a series expansion of a suitable polynomial. The polynomial chosen must be orthogonal to the probability density function of the input variables. The statistical mean and standard deviations are calculated analytically by truncating the series expansion up to a suitable degree of polynomial. The degree of polynomial is called the order of polynomial chaos expansion. In this study, it was found that order of 3 was sufficient for accurate estimation of mean and standard deviation. The number of minimum samples necessary to calculate mean and standard deviation depends on number of uncertain input variables and the order of PCE [11]. The required sample size (s) is given by Eq. (1).

$$s = (n + k)! / n!k! \quad (1)$$

where n is the number of stochastic input variables and k is the order of PCE. In the present study, 25 samples were evaluated for each design to quantify the uncertainties in response.

3.6 Optimization Methodology

The multi-objective problem defined in the previous sections was solved using an algorithm called HYBRID which is a combination of a gradient-based method and a genetic algorithm [12]. The genetic algorithm was magnifying front genetic algorithm (MFGA) which is not a generation-based algorithm but a steady-state algorithm

with controlled elitism [13]. The gradient-based algorithm is based on sequential quadratic programming (SQP) method. The SQP algorithm was used for local refinement internally by HYBRID algorithm. The ratio of genetic algorithm iterations to SQP-based algorithm iterations was 3:1 to achieve global search as well as local refinement. Since computational model was replaced with a response surface model which could be solved within a fraction of seconds, the number of iterations was not a limitation. The algorithm was configured to run for 10,000 iterations. The algorithm was capable of automatically generating a starting DOE with ULH algorithm to start optimization. In robust design optimization, 25 samples were generated using Latin hypercube sampling to evaluate uncertainties present in the responses. Optimization was done on RSM instead of actual computational model, and the optimized model was validated using the computational model.

4 Results

The optimization study resulted in a Pareto front showing a trade-off between vehicle mass and vehicle range for both deterministic optimization and robust design optimization. Figure 5 shows the samples evaluated in deterministic optimization. It was evident that the designs lying on or near Pareto front were surrounded by unfeasible designs indicating the possibility of poor off-design characteristics. Figure 5 shows the designs on the Pareto front showing a trade-off between vehicle mass and vehicle range. The Pareto front appeared to be discrete which could be due to the presence of discrete variables.

Figure 6 shows the mean values of samples evaluated in robust design optimization. Some unfeasible designs were observed near feasible designs, but since the uncertainties in responses were considered in objective and constraint functions, the robust designs did not exhibit poor off-design characteristics. Figure 6 shows the Pareto front obtained through robust design optimization. The Pareto front appeared to be discrete similar to that of the deterministic optimization.

A comparison of Pareto front is shown in Fig. 7, clearly indicating that the mean values of the Pareto designs found by robust design optimization are lower than

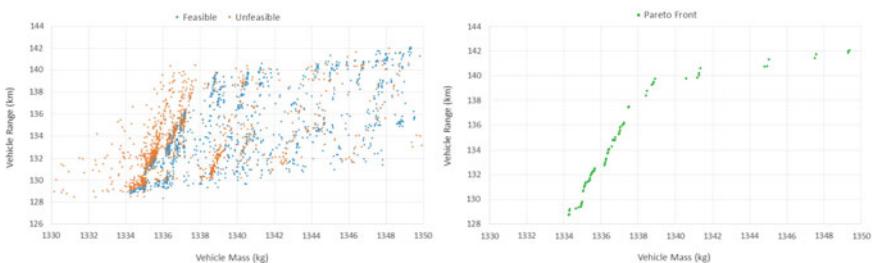


Fig. 5 Vehicle range versus vehicle mass scatter plot and Pareto front for deterministic optimization

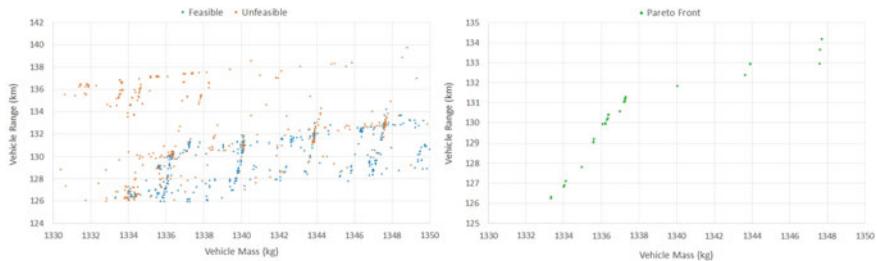


Fig. 6 Vehicle range vs. vehicle mass scatter plot and Pareto front for robust design optimization

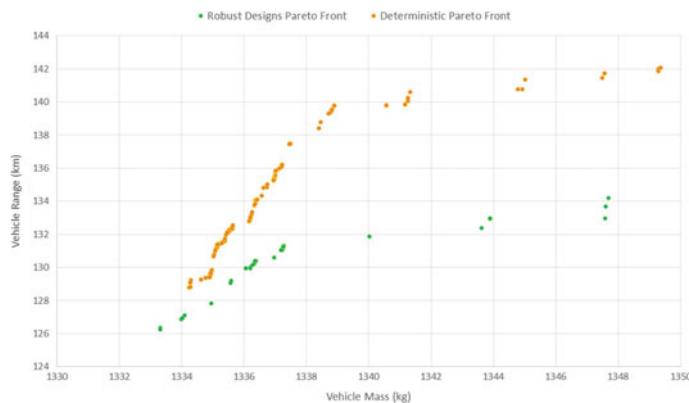


Fig. 7 Comparison of Pareto fronts

the deterministic values of the Pareto designs found by deterministic optimization. Pareto designs found by deterministic optimization are not robust enough to perform as per expectations under stochastic input variables.

4.1 Design Selection

A feasible design having highest vehicle range was selected as the final solution. The details of this design are given in Table 6. Three sigma limits are considered as minimum and maximum values of input variables.

Table 6 Selected design

Design variable	Nominal value (mean, μ)	Maximum value ($\mu + 3\sigma$)	Minimum value ($\mu - 3\sigma$)	Standard deviation (σ)
Motor torque (Nm)	133.59	136.59	130.59	1
Motor power (W)	46,113	48,213	44,013	700
Transmission ratio	8.19	8.19	8.19	0
Number of cells in series	85	85	85	0
Number of cells in parallel	2	2	2	0
Radiator width (m)	0.423	0.429	0.417	0.002

4.2 Validation

The optimization was done on a response surface model and not on the actual computational model. The selected design was validated using the actual computational model. The nominal values given in Table 6 were taken as mean values, and 25 samples were generated using Latin hypercube sampling method. The samples were evaluated using actual computational model, and the output values were recorded. Uncertainty quantification was done using polynomial chaos expansion, and results were compared with the values obtained from RSM-based optimized design. The comparison is given in Table 7.

Figure 8 shows the distribution of variation of different responses. Mass variation occurs due to variation in radiator width. Vehicle range is significantly varying about the mean which could be due to variations in vehicle mass, motor torque and motor speed. The 3σ bound of vehicle range spreads over 5 km on either side of the mean value. This could result in total of more than 10 km of variation in actual working conditions.

Table 7 Validation of optimized design

Variable	RSM prediction		Simulation results	
	μ	σ	μ	σ
Mass (kg)	1347.71	0.7101	1347.71	0.7114
Vehicle range (km)	134.51	1.7142	134.19	1.7591
Battery temperature (K)	312.84	0.0019	313.02	0.0021
Minimum battery voltage (V)	313.24	0.1912	312.94	0.1649
Maximum battery voltage (V)	347.81	0.01874	348.12	0.018
Controller error (m/s)	0.1377	0.0541	0.1341	0.052

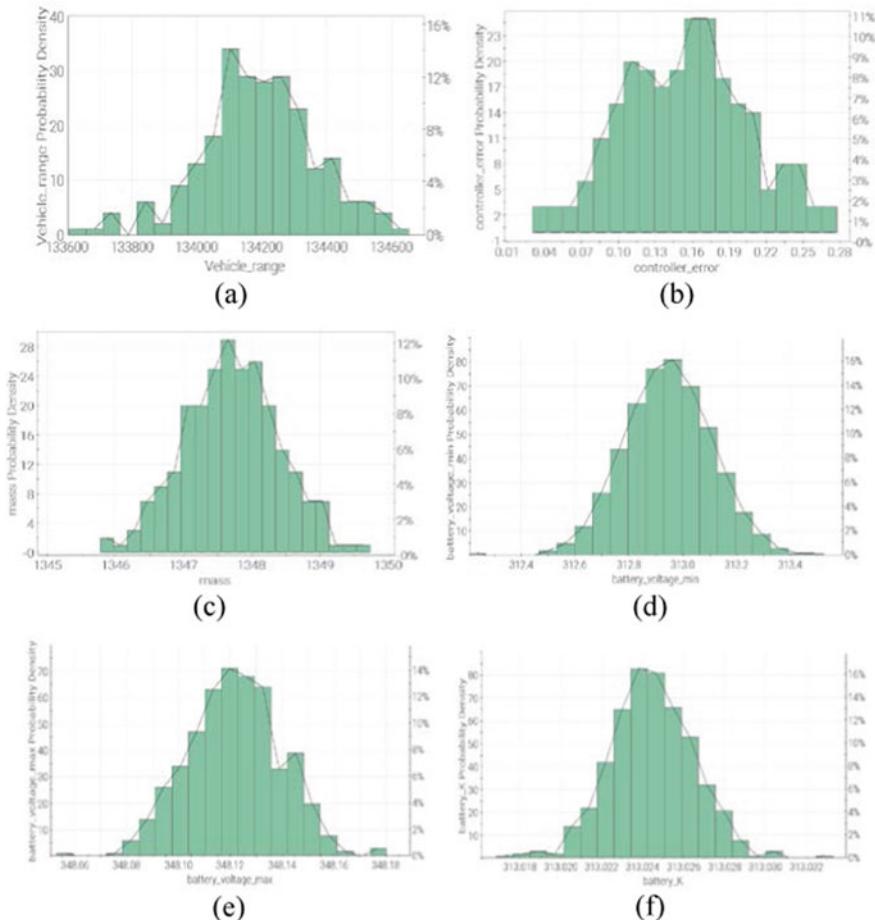


Fig. 8 Probability distribution function of **a** vehicle range (m), **b** controller error, **c** mass **d** minimum battery voltage **e** maximum battery voltage **f** battery temperature for selected design

Controller error also significantly varies on either side of the mean value. The standard deviation is 0.052 m/s, and 3σ variation is 0.156 m/s. The mean value of controller error for selected robust design is 0.1341 m/s which is driven significantly away from the constraint limit of 0.3 m/s by the optimization algorithm to accommodate this random variation.

Battery temperature does not vary much since it is driven by the variation in radiator width which is insignificant. The overall spread of battery temperature is well within the constrained limits. The 3σ limits of minimum and maximum battery voltages are also within the limiting values for the selected design.

5 Conclusion

In the above paper, an electric vehicle was designed and optimized for mass and range using multi-physics, model-based approach combined with multi-objective robust design optimization technique. A set of component sizing parameters were considered, and optimal design choices were obtained against them using optimization routine. The difficulties of multi-disciplinary and multi-objective design of the electric vehicle were tackled efficiently using the above discussed method. A detailed comparison between deterministic optimization and robust optimization was done. Further, consideration of additional electric vehicle objectives like battery life, cost, drivability, handling, safety, etc., are possible within the same framework without much efforts because of the flexible and scalable nature of the models and tools used.

Acknowledgements Authors would like to thank Mr. Anand Pitchaikani for his valuable inputs for this paper. Authors would also like to thank ESTECO S.p.A. for providing computational infrastructure to perform the study.

References

1. I.J.M. Basselink, P.F. van Oorschot, E. Meinders and H. Nijmeijer, Design of an efficient, low weight battery electric vehicle based on a VW Lupo 3L, *The 25th World Battery, Hybrid and Fuel Cell Electric Vehicle Symposium and Exhibition*, EVS-25 Shenzhen, China, Nov. 5–9, 2010
2. MODELON, <https://www.modelon.com/library/electrification-library/>
3. ModeFRONTIER User Guide, ESTECO S.p.A, <https://www.esteco.com/>
4. <https://pypi.org/project/PyFMI/>
5. Y. Wang, C. Du, X. Li, F. Yan, Transmission ratio optimization of electric vehicle powertrain. *Appl. Mech. Mater.* **397–400**, 987–992 (2013). Trans Tech Publications, Switzerland
6. Q. Yin, Z. Wu, X. Rui, Parameter Design and Optimization of Electric Vehicle, in *ITEC Asia-Pacific 2014 1569949781*
7. M.D. McKay, W.J. Conover, R.J. Beckman, Latin hypercube sampling: a Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **21**(2), 239–245 (1979)
8. J. Lee Rodgers, W.A. Nicewander, Thirteen ways to look at the correlation coefficient. *Am. Stat.* **42**(1), 59–66 (1988). <https://doi.org/10.1080/00031305.1988.10475524>
9. S. Sundaresan, K. Ishii, D.R. Houser, A robust optimization procedure with variations on design variables and constraints. *Eng. Optim.* **24**(2), 101–117 (1995). <https://doi.org/10.1080/03052159508941185>
10. R. Askey, J. Wilson, Some basic hypergeometric polynomials that generalize Jacobi polynomials *Memoirs Amer. Math. Soc. AMS Providence RI* 319 (1985)
11. D. Xiu, G.E. Karniadakis, A new stochastic approach to transient heat conduction modeling with uncertainty. *Int. J. Heat Mass Transf.* **46**, 4681–4693 (2003)
12. A. Turco, HYBRID—Description, ESTECO Technical Report 2011–004
13. A. Turco, C. Kavka, MFGA: a GA for complex real-world optimization problems. *Int. J. Innovative Comput. Appl.* **3**(1), 31–41 (2011)

General Applications of Optimization

Multi-disciplinary Optimization Using ISIGHT to Reduce Thermal Gradients



Krishna Nelanti and Venugopal Lingala

Abstract Gas turbine component's design optimization studies are most commonly carried to improve its life and to reduce cost and weight. Thermal gradients are the key contributors for thermal stresses induced in any component which in turn affects its life. Hence, design optimization studies are performed to reduce thermal gradients, especially for the components operating at high temperatures. Multi-disciplinary optimization (MDO) is performed using ISIGHT by integrating Unigraphics, ANSYS, flow network tool, thermal boundary condition tool and post-processing tool. A flange model is considered as a pilot project with thermal barrier coating (length, thickness) and scallop dimensions as design variables. To minimize thermal gradients, a design of experiment (DoE) is performed using "Latin hypercube" sampling technique and also an optimization study is carried out using "genetic algorithm." Transient thermal analysis of the flange is performed for these studies. These studies are carried out by involving activities from design, flow and heat transfer disciplines. Flange thermal gradients are reduced by 20% with these studies.

Keywords Thermal gradients · Thermal stresses · ISIGHT · DoE · Optimization

1 Introduction

Bolted flange joints are very important in joining casings of a gas turbine engine. They have two important functions, namely:

- (a) To maintain the structural integrity of the flange joint
- (b) To prevent the leakage through the flange joint.

K. Nelanti · V. Lingala
Cyient Limited, Hyderabad, India
e-mail: Krishna.Nelanti@cyient.com

V. Lingala
e-mail: Lingala.Venugopal@cyient.com

Thermal analysis of the bolted flange joints is critical as they experience huge thermal gradients during the engine operation. Thermal stresses induced because of thermal gradients are also played an important role in deciding the LCF life of a bolted flange joint.

Bouzid and Nechache [1] outlined the theoretical analysis used for the determination of the steady-state operating temperature and deflections in bolted flange joints. They detailed the theoretical equations necessary to predict the temperature profiles and thermal expansion difference between the joint components necessary for the evaluation of the load redistribution for the two cases of a flange pair and a flange with a cover plate. Fukuoka [2] proposed a simple equation for evaluating the amount of heat flow through a small gap by defining apparent thermal contact coefficient. Accordingly, a numerical approach, which could accurately analyze the thermal and mechanical behaviors of a bolted joint was established. His study showed that only a slight difference in coefficients of linear expansion among the joint members significantly affects the variations of bolt preloads. Ando et al. [3] investigated the leakage behavior of bolted flange connections using the latest computational fluid dynamics (CFD) techniques and the techniques of structural thermal analysis with nonlinear material properties in flange and gasket. Yousuf and Nelanti [4] performed parametric studies on bolted flange model using flange height, flange thickness and gap between two flanges as parameters. Impact of each parameter on flange thermal gradients is discussed. Bandari and Nelanti [5] performed bolted flange model optimization studies and proposed an optimized design subjected defined constraints.

As many researchers are performed bolted flange model optimization studies in standalone mode, i.e., without integrating subactivities of the analysis. Hence, this paper attempted to perform optimization studies by integrating all the subactivities using ISIGHT tool.

2 Pilot Project

Three-dimensional five bolted flange model with thermal barrier coating (TBC) on inner sides of casings is considered as pilot project to reduce flange thermal gradients by performing MDO using ISIGHT. TBC thickness, TBC length and scallop dimensions are considered as design variables as shown in Figs. 1 and 2. Maximum allowable limits of design variables are described in Table 1. These variables are taken from [5].

A simple flight cycle mission illustrated in Fig. 3 is considered for this study [5]. The sample mission starts with stabilized low-power condition (Idle) and accelerated to high-power condition (take-off) in 15 s is known as snap accel duration. To stabilize the temperatures at take-off, a dwell of 25 min is introduced, then the engine is decelerated to idle condition in 25 s is known as snap decel duration and then engine is allowed to run for a period of 40 min at idle condition.

Fig. 1 3D parametric Flange model side view

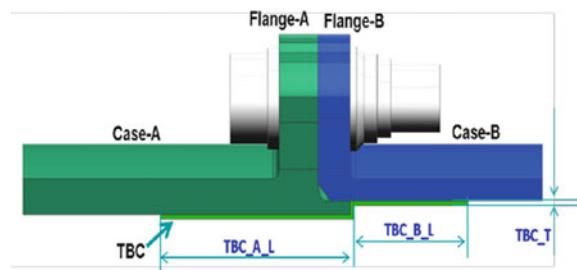


Fig. 2 3D parametric Flange model front view

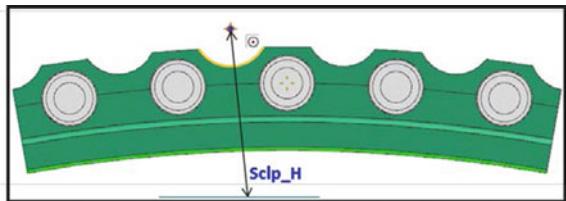
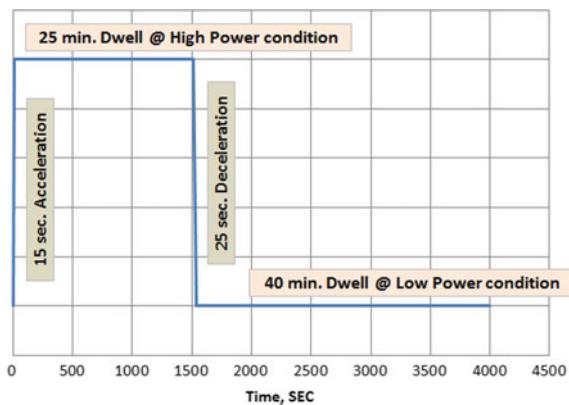


Table 1 Design variables allowable limits

Parameter description	Parameters	Allowable limits	
		Minimum	Maximum
Scallop height from case center, inch	Sclp_H	12.6	12.65
TBC coat length under Case-A, inch	TBC_A_L	0.62	1.62
TBC coat length under Case-B, inch	TBC_B_L	0.04	1.04
TBC coat thickness, mils(10e-3 inch)	TBC_T	10	30

Fig. 3 Flight cycle mission



3 Project Setup

MDO project is set up in ISIGHT as shown in Fig. 4. Two inbuilt components of ISIGHT, namely DATA EXCHANGER and SIMCODE, are used to update Unigraphics (UG V9.0) part file using its expression file and generate geometric model. Two customized components are used for flow and thermal boundary conditions estimation. DATA EXCHANGER component is used to update boundary condition output file. SIMCODE component is used to sort analysis files and perform thermal analysis using ANSYS V18.1. DATA EXCHANGER and SIMCODE components are used to post process the analysis results. At the end, SIMCODE component is used to perform the entire process for each sample.

Detailed work flow of MDO process is explained below.

Step.1: DOE generates design variables matrix using specified DoE technique for required number of loops

Step.2: Updates UG expressions file with sample data from design variable matrix

Step.3: Loads expressions file into parametric UG model using a perl script batch file to update the geometry

Step.4: Flow network tool executes flow model and generates flow values and supplies to BCs tool

Step.5: BCs tool generates thermal boundary conditions

Step.6: Updates thermal boundary condition file and supplies to ANSYS through data exchanger component

Step.7: Finite element modeling tool generates FE model in ANSYS by importing the attributed UG model

Step.8: Performs transient thermal analysis using ANSYS

Step.9: Generates transient plots using a customized tool and inserts into a presentation

Step.10: Extracts maximum transient thermal gradients for each flange and provides to DoE to continue the loop.

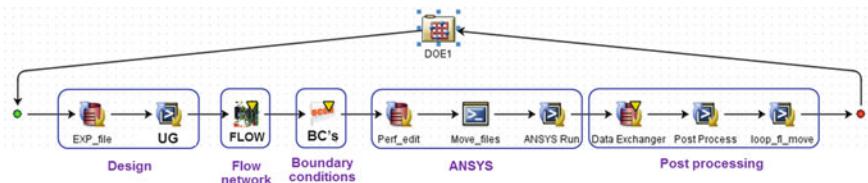


Fig. 4 Project setup

4 Flow System

Flow model is simulated based on the restrictions at flange ID interference fit (flange snap joint) and at radial gaps between flange and bolt head/nut are modeled using isentropic nozzle relations based on open literature survey [5]. Leakage along the radial gap and at snap joint between two flanges are modeled as friction loss, which is calculated based on Moody's chart for turbulent flow, as the gap is too small and the roughness of the both flanges would not allow the flow to be laminar. The radial gap and the snap gap are assumed to be uniform circumferentially. Leakage flows are estimated for both low-power (idle) and high-power (take-off) conditions. It is noticed that snap joint is metering the leakage flow through the flanges, and it is also observed that most of the air is leaking out through the gap between the flange and bolt heads/nuts.

5 Thermal Boundary Conditions

Flange inner side is considered as a high-temperature and high-pressure zone and outer side is assumed as a cold environment with low pressure, i.e., flange is leaking from ID to OD cavity. The bulk air temperatures with transient responses are applied for both inner and outer sides of flange to capture the actual lag in air temperature response compared to rotor speed response.

Convective heat transfer coefficient on the inner side of the flanges is estimated using the correlation for flow over a flat plate, which is taken from [6] and given in Eq. (1)

$$\text{Nu} = 0.0296 \text{ Re}^{0.8} \text{Pr}^{0.333} \quad (1)$$

On the outer side of the flange, conditions for free convection exist, so the correlation given in Eq. (2) is used to model free convection, which is taken from [6].

$$\text{Nu} = 0.15 \text{ Ra}^{0.33} \quad (2)$$

For modeling convection due to leakage between the flanges and between bolts and nuts, the well-known Dittus–Boelter correlation given in Eq. (3) is used. This is taken from [6].

$$\text{Nu} = 0.023 \text{ Re}_D^{0.8} \text{Pr}^{0.3} \quad (3)$$

Radiation mode of heat transfer is considered in the form of equivalent convective heat transfer and is simulated along with convection.

6 Finite Element Model

A customized modeling tool is used for finite element creation, which does background APDL scripting for meshing and surface effective elements creation after importing UG model into ANSYS. SOLID70 elements are used for meshing. Thermal contact at the flange interface with bolt/nut and between the flanges is modeled with TARGE170 and CONTA174 elements.

FLUID116 elements are used for fluid network to simulate the leakage through flange, bolts and nuts. Convective boundary conditions are applied using SURF152 elements. A user-specified APDL script is developed to model fluid elements, contacts and fluid node attachments in ANSYS FE model. Materials made up of nickel alloy, which can withstand temperatures up to 1200 °F, are chosen for the flanges and fasteners. FE model with fluid network and attachments is shown in Fig. 5.

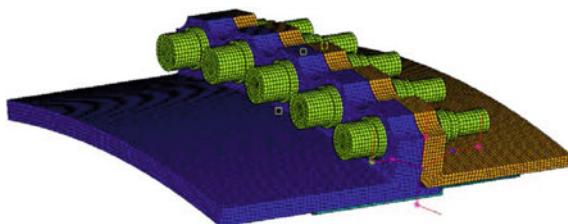


Fig. 5 ANSYS 3D FE model

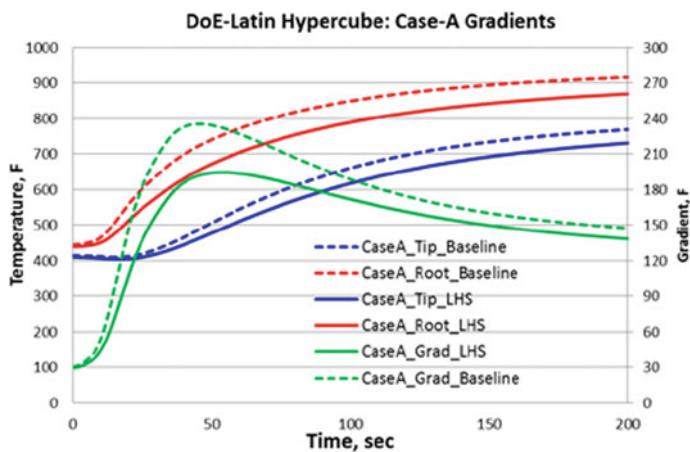


Fig. 6 Transient Case-A flange gradients

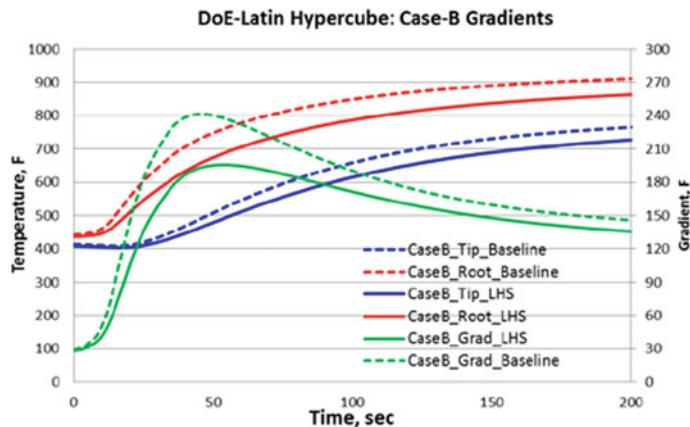


Fig. 7 Transient Case-B flange gradients

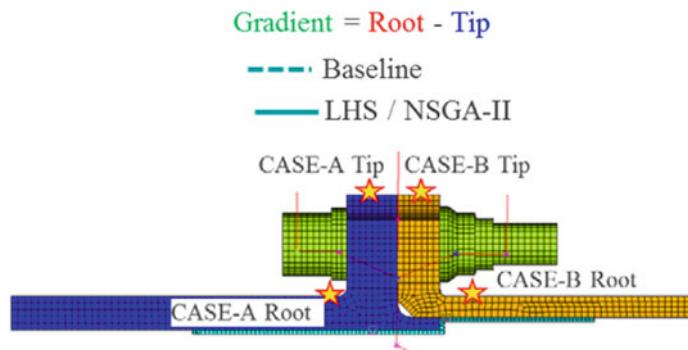


Fig. 8 Flange Gradient nodal location representation

7 Results and Discussion

Initially, three-dimensional flange model thermal analysis is performed using baseline geometry, i.e., without TBC, without scallops and predicted maximum transient thermal gradients as 235 and 241 °F for the flanges A and B, respectively [5]. Two types of multi-disciplinary optimization studies are performed using DoE and optimization techniques.

8 DoE—Latin Hypercube Sampling Technique

Multi-disciplinary optimization studies are performed using DoE with Latin hypercube sampling technique. Iteration 16 predicted that maximum flange gradients can be reduced to 195 and 196 °F for the flanges A and B, respectively. Table 2 summarizes the results of 20 iterations. All design variables are influential on maximum flange gradients, but TBC thickness has more influence on both the flange gradients. The impact of TBC thickness is 40% and 30% on the flanges A and B, respectively, as shown in Fig. 9. Transient gradients are plotted between case root and flange tip for both Case-A and Case-B, shown in Fig. 8. Transient gradients are shown in Figs. 6 and 7; maximum transient gradients are occurring during accel and less than 50 s time for both Case-A and Case-B flanges.

Table 2 Latin hypercube DoE results

Run Path							Parameters for all iterations (Done)			
	Run #	Run #	Sdp_H_o	TBC_A_L	TBC_B_L	TBC_t_o	Max_grd_Flng_A	Max_grd_Flng_B		
✓	1	1	12.46	0.42	0.724	0.0305	204.664	205.3764		
✓	1	2	12.6026	1.567	0.093	0.0184	198.2506	223.4169		
✓	1	3	12.6053	1.62	0.882	0.0142	203.3574	212.9849		
✓	1	4	12.6079	1.515	0.777	0.0153	201.8807	212.2603		
✓	1	5	12.6105	1.51	0.769	0.0121	200.8193	210.8585		
✓	1	6	12.6132	1.094	0.672	0.0174	201.3827	210.2401		
✓	1	7	12.6158	0.725	0.04	0.0216	203.2103	223.2821		
✓	1	8	12.6184	1.409	0.566	0.0237	191.7976	204.4048		
✓	1	9	12.621	0.935	0.713	0.0158	194.9495	202.564		
✓	1	10	12.6237	0.88	0.303	0.0208	192.6189	206.0698		
✓	1	11	12.6263	1.146	0.198	0.0163	203.5534	220.7015		
✓	1	12	12.6289	1.304	0.987	0.0132	206.6377	215.0554		
✓	1	13	12.6316	1.462	0.461	0.0111	209.9204	222.265		
✓	1	14	12.6342	1.25	0.251	0.01	212.9396	226.9951		
✓	1	15	12.6368	1.199	0.176	0.0126	195.2179	210.9433		
✓	1	16	12.6395	0.831	0.935	0.0229	194.5473	195.7052		
✓	1	17	12.6421	0.673	0.514	0.0247	201.8704	205.142		
✓	1	18	12.6447	1.357	0.145	0.03	186.6471	213.1295		
✓	1	19	12.6474	0.988	1.04	0.0195	201.0696	206.8279		
✓	1	20	12.65	0.776	0.637	0.0268	197.2713	201.5972		

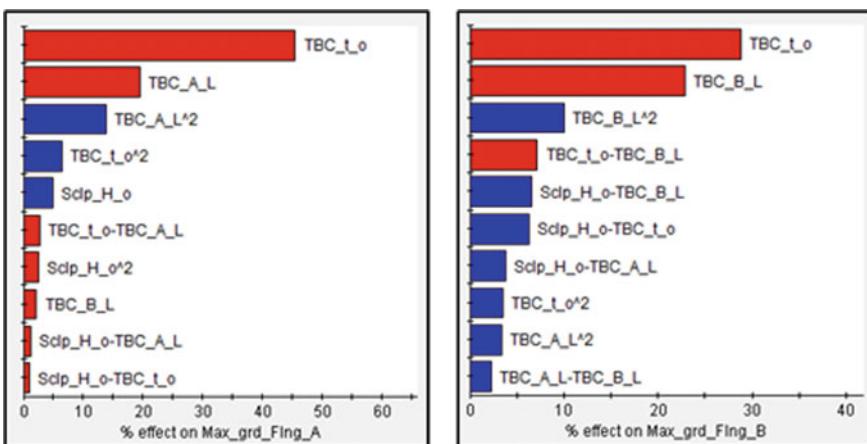


Fig. 9 Design variables impact on flange gradient

9 Genetic Algorithm (NSGA-II) Optimization

Multi-disciplinary optimization studies are performed using genetic algorithm NSGA-II technique. Iteration 11 predicted that maximum flange gradients are reduced to 190°F and 199°F for the flanges A and B, respectively. Table 3 summarizes the results of 17 iterations. Transient gradients are shown in Figs. 10 and 11; maximum transient gradients are occurring during accel and less than 50 s time for both Case-A and Case-B flanges.

For this NSGA-II optimization, transient gradients are plotted between case root and flange tip for both Case-A and Case-B shown in Fig. 6. Transient gradients are shown in Graph 3 and Graph 4 below; maximum transient gradients are occurring during accel and less than 50 s time for both Case-A and Case-B flanges.

Table 3 NSGA-II optimization study results

Run Path	Parameters for all Iterations (Done)										
	Run #	Scd_H_0	Trg_L_0	Trg_R_0	Max_gnd_Flg_A	Max_gnd_Flg_B	Max_gnd_Flg_B	Design Feasibility	Objective and Penalty	Objective Function	Penalty Functions
✓ 1 1 1 12.41229515 1.77261648 1.07294796 0.01894942812 197.1166 200.3296 9 402.6462 402.6462 0.0											
✓ 1 2 2 12.82974052 1.06466251 0.637984762 0.02480563975 193.5887 202.454 9 396.0127 396.0127 0.0											
✓ 1 3 3 12.60090330 1.38950104 0.219883459 0.01447492401 203.5669 221.4506 1 131541.3479 425.1075 131111.2404											
✓ 1 4 4 12.84082325 0.748174364 0.028054423194 0.02030112906 201.1462 206.3644 7 407.5106 407.5106 0.0											
✓ 1 5 5 12.41229515 1.38950104 0.549742384 0.02480563975 193.5887 200.3296 7 396.0127 396.0127 0.0											
✓ 1 6 6 12.82974052 1.802951297 0.04529514669 0.01176909043 213.3854 218.4564 1 81982.07992 431.6218 81530.45912											
✓ 1 7 7 12.81547020 1.1802951297 0.04529514669 0.01176909043 203.0861 227.1379 1 294137.8405 430.224 293707.6184											
✓ 1 8 8 12.41229515 0.7566036561 0.04427299043 0.02970854008 194.1799 200.8001 9 304.979 304.979 0.0											
✓ 1 9 9 12.82974052 1.38950104 0.219883459 0.01447492401 203.5669 221.4506 7 396.0127 396.0127 0.0											
✓ 1 10 10 12.41229515 0.7566036561 0.04427299043 0.02970854008 194.1799 200.8001 7 396.0127 396.0127 0.0											
✓ 1 11 11 12.84082325 0.998149609 0.998149609 0.02547234745 189.7569 198.5725 9 388.3294 388.3294 0.0											
✓ 1 12 12 12.43177116 1.38950104 0.01894942812 0.02547234745 197.5201 213.3658 1 11737.47585 410.8856 11326.5925											
✓ 1 13 13 12.84082325 0.980149609 0.790803779 0.02547234745 197.5201 200.8001 7 404.9386 404.9386 0.0											
✓ 1 14 14 12.43177116 0.821234453 0.02547234745 0.02547234745 197.5201 213.3658 1 409.4512 409.4512 40653.7729											
✓ 1 15 15 12.84082325 0.74026321 0.02547234745 0.02547234745 194.9586 201.0579 7 395.6537 395.6537 0.0											
✓ 1 16 16 12.43177116 0.922080093 0.861979302 0.02601512906 200.9976 206.3336 7 407.3338 407.3338 0.0											
✓ 1 17 17 12.64372991 1.531914909 0.998149609 0.02547234745 189.7569 198.5725 8 388.3294 388.3294 0.0											

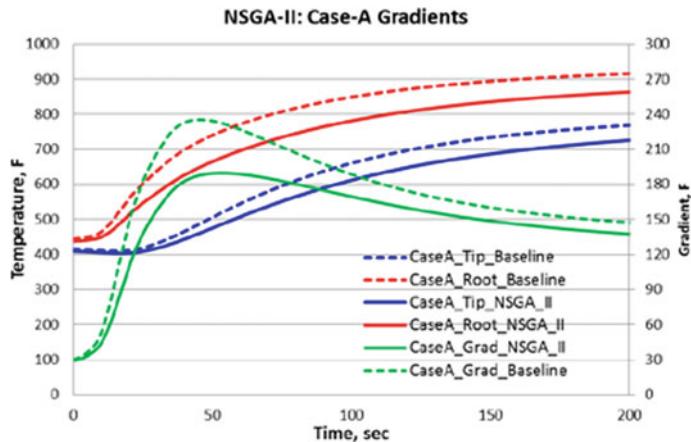


Fig. 10 Transient Case-A flange gradients

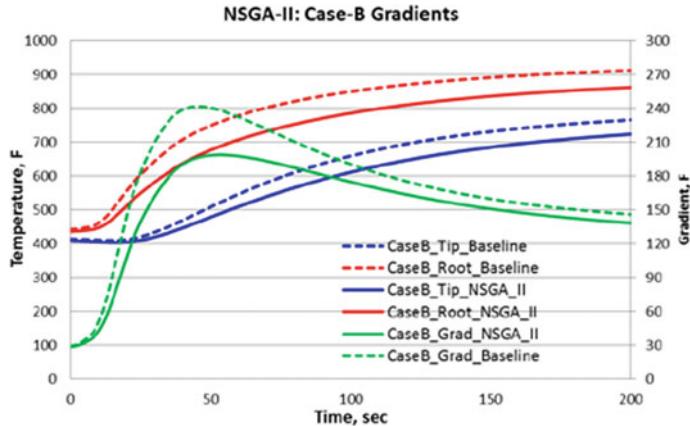


Fig. 11 Transient Case-B flange gradients

10 Conclusions

Multi-disciplinary optimization setup is made in ISIGHT using 3D flange model to reduce its transient thermal gradients. Latin hypercube sampling technique is used for DoE studies and predicted that flange thermal gradients are lowered by 17% and 19% on flange A and flange B, respectively. NSGA-II genetic algorithm technique is used for optimization studies and predicted that flange thermal gradients are lowered by 19% and 17% on flange A and flange B, respectively. Table 4 summarizes the comparison of flange max thermal gradients against baseline model.

Table 4 Maximum gradients conclusion

Analysis	Iteration	Sclp_H_o (inch)	TBC_A_L (inch)	TBC_B_L (inch)	TBC_t (inch)	Max_grd_flnq_A °F	Max_grd_flnq_B °F	Case-A Gradients reduction %	Case-A Gradients reduction %
Baseline (No TBC)	1	NA	NA	NA	NA	235	241		
DOE Latin Hypercube	20	12.63	0.831	0.935	0.0279	194	195	17	19
Optimization NSGA-II	17	12.6	1.53	0.99	0.025	190	199	19	17

References

1. A.-H. Bouzid, A. Nechache, Thermally induced deflections in bolted flanged connections. *J. Pressure Vessel Technol.* **127**(4), 394–401 (2005)
2. T. Fukuoka, Finite Element Analysis of the thermal and mechanical behaviors of a bolted joint. *J. Pressure Vessel Technol.* **127**(4), 402–407 (2005)
3. F. Ando, Y. Ido, T. Oginuma. Simulation of leakage behavior on bolted flanged connections in elevated temperature service, in *ASME 2008 Pressure Vessels and Piping Conference*, vol. 7, pp. 331–340 (2008)
4. ASME GTINIDA, M. Yousuf, K. Nelanti. *Parametric Studies on Bolted Flange Thermal Analysis to Improve Lcf Life*, December 5–6, 2013, Bangalore, Karnataka, India (2013)
5. NCMDAO-2018, V. Bandari, K. Nelanti. *Optimization of Bolted Flange Design through Parametric Studies using Multi-Disciplinary Analyses*, March 2018, IISc, Bangalore
6. F.P. Incropera, D.P. DeWitt. *Fundamentals of Heat and Mass Transfer*, 5th edn. (John Wiley and Sons, Singapore)

Optimization of Kalman Filter for Target Tracking Applications



T. Sunil Kumar and Punithavathi Duraiswamy

Abstract The computational complexity, area overhead and power consumption are the major drawbacks of Kalman filter that limits its use for various real-time applications including target tracking. This paper aims to optimize the Kalman algorithm to meet the growing demands of today's tracking system. A reduced recursive scalar Kalman filter is proposed by simplifying the calculations of the posterior error co-variance estimate. This reduces the hardware cost and computational time. For hardware implementation, the architecture is divided into two parts. The first part calculates Kalman gain and posterior error co-variance estimate, and the second part calculates the output estimate. Both pipelining and parallel processing techniques are employed to increase the speed of computation. The area and power consumption are reduced by time sharing the architecture as the algorithm is recursive. The proposed technique is implemented on FPGA, and the performance of the Kalman filter is verified. The implementation requires nine adders, three subtractors, six multipliers and two dividers. A maximum operating frequency of 145 MHz is obtained with a worst case computational delay of seven clock cycles (6.881 ns). A parabolic path is used to verify the tracking performance of the architecture, and the simulation results show an accuracy loss of 1.98% between the actual and the estimated path. The architecture consumes a power of 0.128 W at 100 MHz.

Keywords Kalman filter · FPGA · Pipelining · Parallel processing · Time-shared architecture

1 Introduction

Tracking targets using radars may be very challenging due to the randomness of the data and presence of background noise, clutter and other interference [1–3]. Due to the randomness in the radar data, tracking algorithms play a vital role in target detection

T. Sunil Kumar · P. Duraiswamy
Department of Electronics and Communication Engineering,
M S Ramaiah University of Applied Sciences, Bangalore, India
e-mail: punithavathi.ec.et@msruas.ac.in

and clutter/noise removal. Kalman filter is a powerful and precise linear estimator that minimizes the mean square error in target detection and tracking. Ever since it was proposed by R. E. Kalman in 1960 [4], it has been widely used for applications in the field of control systems, signal processing and wireless sensor networks [5–11]. It is a complex algorithm that requires a high degree of parallelism to suit the growing demands of today’s tracking systems involving multi-target tracking [12]. In order to reduce the complexity of the algorithm, the original Kalman filter state transition equations are converted to scalar arrays. A scalar form of Kalman filter and a parallel architecture implementation is proposed in [13] for a track while scan (TWS) radar. As the algorithm is recursive, it resulted in idle time for few hardware blocks, wasting the area and power. To efficiently use the hardware resources, and further increase the processing speed, authors in [14] developed a pipelined parallel architecture using the scalar form proposed in [13]. The authors showed an accuracy loss of 4.9% for tracking a straight line path and a semicircle path. In this paper, we investigate the scalar Kalman filter equations proposed in [13] and further simplify it to optimize architecture for area and speed. A reduced recursive scalar algorithm is proposed and implemented efficiently using a time-shared pipelined parallel architecture. The performance of the Kalman filter is verified, and the results are reported.

2 Kalman Filter Algorithm

2.1 Algorithm

Kalman filter works in two steps: prediction state and correction state. In the prediction state, the Kalman filter gives an approximation for the current state along with uncertainties. This state is responsible for calculating the a priori error co-variance required for the correction stage. In the correction or update state, the measurements are observed and corrected using a weighted average. This state is responsible for obtaining an improved posteriori error co-variance. Finally, the Kalman gain is calculated to update the output and error. These calculations are recursive with a certain initial estimate. In real time, it requires only present input, previous state and uncertainties in previous measurements eliminating the need to store large amount of previous data. To understand the working of Kalman filter, a TWS described in [12], also used in [13, 14], will also be used in our work whose model equations are given below:

$$X(k+1) = AX(k) + W(k) \quad (1)$$

$$Y(k) = CX(k) + V(k) \quad (2)$$

where $A = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$ is the state transition matrix;

$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ is the observation matrix;

$X(k) = \begin{bmatrix} X_1(k) \\ X_2(k) \\ X_3(k) \\ X_4(k) \end{bmatrix}$ is the state variable matrix;

$W(k) = \begin{bmatrix} 0 \\ U_1(k) \\ 0 \\ U_2(k) \end{bmatrix}$ is the process noise variable matrix;

$Y(k) = \begin{bmatrix} Y_1(k) \\ Y_2(k) \end{bmatrix}$ is the measurement variable matrix;

$V(k) = \begin{bmatrix} V_1(k) \\ V_2(k) \end{bmatrix}$ is the measurement noise variable matrix.

T is sampling time (inverse of radar scan rate), state variables $X_1(k)$ and $X_3(k)$ are the range of target in X (aircraft range) and Y (radial velocity) direction, respectively. $X_2(k)$ and $X_4(k)$ are the rate of change of position or velocity of target in X (aircraft bearing) and Y (angular velocity) direction, respectively. The process noise variable $U_1(k)$ represents the rate of change in radial velocity and $U_2(k)$ represents the rate of change bearing over the interval T . $V_1(k)$ and $V_2(k)$ are the measured noise variables. They are all assumed to be Gaussian. For the above descriptions, Kalman filter equations are written in the following form.

$$\begin{aligned} PI(k/(k-1)) &= AP((k-1)/(k-1))A^T + Q(k-1) \\ \hat{X}I(k/(k-1)) &= A\hat{X}((k-1)/(k-1)) \\ \hat{Y}I(k) &= C\hat{X}(k/(k-1)) \\ G(k) &= PI(k/(k-1))C^{-1}[CPI(k/(k-1))C^{-1} + R(k)]^{-1} \\ \hat{X}I(k/k) &= \hat{X}I(k/(k-1)) + G(k)[Y(k) - \hat{Y}(k)] \\ P(k/k) &= PI(k/(k-1)) - G(k)CPI(k/(k-1)) \end{aligned} \quad (3)$$

where

$$P(k/k) = \begin{bmatrix} P_{11} & P_{12} & 0 & 0 \\ P_{21} & P_{22} & 0 & 0 \\ 0 & 0 & P_{33} & P_{34} \\ 0 & 0 & P_{43} & P_{44} \end{bmatrix}; \quad PI(k/(k-1)) = \begin{bmatrix} PI_{11} & PI_{12} & 0 & 0 \\ PI_{21} & PI_{22} & 0 & 0 \\ 0 & 0 & PI_{33} & PI_{34} \\ 0 & 0 & PI_{43} & PI_{44} \end{bmatrix}$$

$$Q(k)) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \sigma_1^2(k) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_2^2(k) \end{bmatrix}; \quad G(k)) = \begin{bmatrix} G_{11} & 0 \\ G_{21} & 0 \\ 0 & G_{32} \\ 0 & G_{42} \end{bmatrix}$$

$$\hat{X}I(k/(k-1)) = \begin{bmatrix} XI_1 \\ XI_2 \\ XI_3 \\ XI_4 \end{bmatrix}; \quad R(k) = \begin{bmatrix} \sigma_p^2(k) & 0 \\ 0 & \sigma_\theta^2(k) \end{bmatrix}$$

$$\hat{X}I(k/k)) = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}; \quad Y(k) = \begin{bmatrix} \hat{X}_1 \\ \hat{X}_2 \end{bmatrix}$$

In Eq. (3), $PI(k/(k-1))$ and $P(k/k)$ are the a priori and posterior error co-variance estimate, respectively. $Q(k)$ is the system noise co-variance estimate, $\hat{X}I(k/(k-1))$ and $\hat{X}(k/k)$ are the a priori and posterior state estimate, respectively. $R(k)$ is the measurement noise co-variance matrix, $\hat{Y}I(k)$ is the output estimate, and $G(k)$ is Kalman gain matrix. $\sigma_1^2(k)$ and $\sigma_2^2(k)$ are the variances of T multiplied by the radial and angular acceleration, respectively. $\sigma_p^2(k)$ and $\sigma_\theta^2(k)$ are the variances of T multiplied by radial and angular measurement noise, respectively. The values used here are $\sigma_p^2(k) = E[v_1^2(k)] = 1$, $\sigma_\theta^2(k) = E[v_2^2(k)] = 1$, $\sigma_1^2(k) = E[u_1^2(k)] = 330$, $\sigma_2^2(k) = E[u_2^2(k)] = 1.3 \times 10^{-8}$. At time $k=1$ and $k=2$, initialization of $\hat{X}(2/2)$ and calculation of $P(2/2)$ of Kalman filter is carried out. Thus, $\hat{X}I(3/2)$ and $Y(3)$ are calculated using $X(\hat{2}/2)$, and the first data set derived is $[P(2/2), (\hat{X}I(3/2), \hat{Y}(3))]$. Using the first data set, the next data sets such as $(P(3/3), \hat{X}I(34/32), \hat{Y}(4))$ and so on are calculated.

2.2 Scalar Kalman Filter

Representation of Kalman filter equations in Eq. (3) are complex to implement in hardware. A scalar Kalman filter implementable in hardware, proposed in [13], reduces the computational complexity and the same is represented through Eqs. (4)–(9). Here, we further simplify scalar Kalman filter equations and propose a reduced recursive scalar algorithm to optimize the speed, area and power making it suitable for real-time implementations.

$$\begin{aligned} PI_{11} &= P_{11} + TP_{21} + TPI_{12} & PI_{33} &= P_{33} + TP_{43} + TPI_{34} \\ PI_{12} &= P_{12} + TP_{22} & PI_{34} &= P_{34} + TP_{44} \\ PI_{21} &= P_{21} + TP_{22} & PI_{43} &= P_{43} + TP_{44} \\ PI_{22} &= P_{22} + \sigma_1^2 & PI_{44} &= P_{44} + \sigma_2^2 \end{aligned} \quad (4)$$

$$\begin{aligned} XI_1 &= X_1 + TX_2 \\ XI_2 &= X_2 \\ XI_3 &= X_3 + TX_4 \\ XI_4 &= X_4 \end{aligned} \tag{5}$$

$$\begin{aligned} \hat{Y}_1 &= XI_1 \\ \hat{Y}_2 &= XI_3 \end{aligned} \tag{6}$$

$$\begin{aligned} G_{11} &= PI_{11}/(PI_{11} + \sigma_p^2) \\ G_{21} &= PI_{21}/(PI_{11} + \sigma_p^2) \\ G_{32} &= PI_{33}/(PI_{33} + \sigma_\theta^2) \\ G_{42} &= PI_{43}/(PI_{33} + \sigma_\theta^2) \end{aligned} \tag{7}$$

$$\begin{aligned} X_1 &= XI_1 + G_{11}[Y_1 - \hat{Y}_1] \\ X_2 &= XI_1 + G_{21}[Y_1 - \hat{Y}_1] \\ X_3 &= XI_1 + G_{32}[Y_2 - \hat{Y}_2] \\ X_4 &= XI_4 + G_{42}[Y_2 - \hat{Y}_2] \end{aligned} \tag{8}$$

$$\begin{aligned} P_{11} &= PI_{11} - G_{11}PI_{11} & P_{33} &= PI_{33} - G_{32}PI_{33} \\ P_{12} &= PI_{12} - G_{11}PI_{12} & P_{34} &= PI_{34} - G_{32}PI_{34} \\ P_{21} &= PI_{21} - G_{21}PI_{11} & P_{43} &= PI_{43} - G_{42}PI_{33} \\ P_{22} &= PI_{22} - G_{12}PI_{12} & P_{44} &= PI_{44} - G_{42}PI_{34} \end{aligned} \tag{9}$$

3 Proposed Reduced Scalar Kalman Filter

Upon investigating Eqs. (7) and (9), used for Kalman gain calculation and posterior error co-variance estimate, it provides provision for further simplification. Equation (9) can be further simplified to reduce the number of arithmetic operations reported in [13]. Consider the static equation of Kalman gain calculation in Eq. (9), to further simplify, we multiply and divide G_{21} in Eq. (9) by PI_{11}/PI_{11} and represent it in terms of G_{11} . Similarly, multiplying and dividing G_{42} by PI_{33}/PI_{33} , we represent it in terms of G_{32} . The new representations of G_{21} and G_{42} are shown in Eqs. (10) and (11). Accordingly, P_{21} and P_{43} calculations are modified as given in Eqs. (12) and (13). It is seen that G_{21} is eliminated from P_{21} calculation and G_{42} is eliminated from P_{43} calculation. Thus, posterior error co-variance estimate scalar equation reduces to simplest form as summarized in Eq. (14).

$$G_{21} = G_{11} \times PI_{21}/PI_{11} \tag{10}$$

$$G_{42} = G_{32} \times PI_{43}/PI_{33} \quad (11)$$

$$P_{21} = (1 - G_{11})PI_{21} \quad (12)$$

$$P_{43} = (1 - G_{32})PI_{43} \quad (13)$$

$$\begin{aligned} P_{11} &= (1 - G_{11})PI_{11} & P_{33} &= (1 - G_{32})PI_{33} \\ P_{12} &= (1 - G_{11})PI_{12} & P_{34} &= (1 - G_{32})PI_{34} \\ P_{21} &= (1 - G_{11})PI_{21} & P_{43} &= (1 - G_{32})PI_{43} \\ P_{22} &= PI_{22} - G_{21} \times PI_{12} & P_{44} &= PI_{11} - G_{42} \times PI_{34} \end{aligned} \quad (14)$$

Hardware implementation of Eq. (14) requires four subtractors less compared to the implementation of Eq. (9), reported in [13]. The arrived Eq. (14) is reduced form of scalar Kalman filter. This reduces the time required for calculating posterior error co-variance estimate making the computation faster.

4 Time-Shared Pipelined Parallel Architecture

The fully parallel architecture [13] divides the architecture into three parts. However, due to the recursive nature of the algorithm, this resulted in idle time for few hardware blocks wasting the area and power. A pipelined parallel architecture that efficiently uses the hardware resources by time sharing concept [14] improves the speed and reduces the area requirement. Our reduced scalar algorithm being less complex can take advantage of pipelining, parallel processing and time sharing. Since the Kalman filter is recursive, the time taken to process a new data has to wait until the previous iteration is complete. Therefore, the architecture is divided into two parts instead of three.

4.1 Parallelism

There are three major stages in the algorithm; stage (1) $P_{11}, P_{12}, P_{21}, P_{22}, G_{11}, G_{12}$, stage (2) $P_{33}, P_{34}, P_{43}, P_{44}, G_{32}, G_{42}$ and stage (3) Y1 and Y2. The architectures required for calculating stage 1 and stage 2 are same. Instead of implementing two different architectures, one architecture can be implemented and shared. Since the calculations, 1 and 2 are combined in a single architecture, the entire Kalman archi-

ture is divided into two parts and they run in parallel. The first part calculates the Kalman gain (G_{11}, G_{12}, G_{32} and G_{42}) and posterior error co-variance estimate ($P_{11}, P_{12}, P_{21}, P_{33}, P_{34}, P_{43}$ and P_{44}). The second part calculates the estimated output ($Y(k)$).

4.2 Pipelining

Both the first part and the second part of the architecture consist of arithmetic operations like addition, subtraction, multiplication and division. The first and second part architectures are shown in Fig. 1a and b, respectively. The multiplier and divider circuits are complex and slow. Therefore, pipeline registers are used to increase the speed of operation. As only two data sets are available for processing at a time and also taking into account the area constraints, a two-stage pipelining is used. In the data path, delays are introduced to synchronize the data from different blocks. The first part of the architecture is time shared and is explained in the next subsection. The second part of the architecture in Fig. 1b calculates the output estimate using the Kalman gain calculated from the first stage.

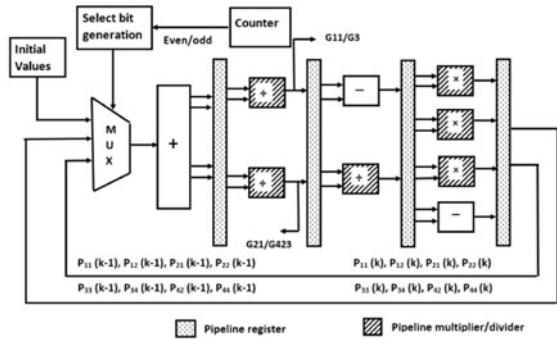
4.3 Time Sharing

Figure 1a is a time-shared architecture for calculating either $P_{11}, P_{12}, P_{21}, P_{22}, G_{11}, G_{12}$ or $P_{33}, P_{34}, P_{43}, P_{44}, G_{32}, G_{42}$. Initial values are provided to the system for the first iteration. A multiplexer selects the right input in accordance with the select bit. The select bit is generated according to the even and odd counting. For instance, during odd counting, $G_{11}, G_{12}, P_{11}, P_{12}, P_{21}, P_{22}$ are calculated. Similarly, when the counter value is even $G_{32}, G_{42}, P_{33}, P_{34}, P_{43}, P_{44}$ are calculated using same resources. The second part runs in parallel with this architecture. Necessary delays are introduced in the data path to synchronize the data. The total Kalman filter architecture consists of nine adders, three subtractors, six multipliers, two dividers and few pipelined registers and a control unit. The proposed architecture is described in Verilog HDL and implemented on Artix -7 FPGA with a data width of 16 bits.

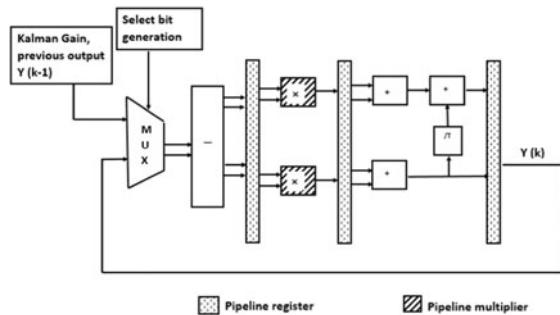
5 Results and Discussions

For functionality verification, a parabolic path is used. A random noise is added to the parabolic path. The simulation results for the tracking performance of the filter are shown in Fig. 1c. An accuracy error of 1.98% is obtained between the actual and the estimated path. This error is mainly due to the limitation of the hardware. Truncation of input data width to 8 bits to avoid overflow is one of the reasons for the error.

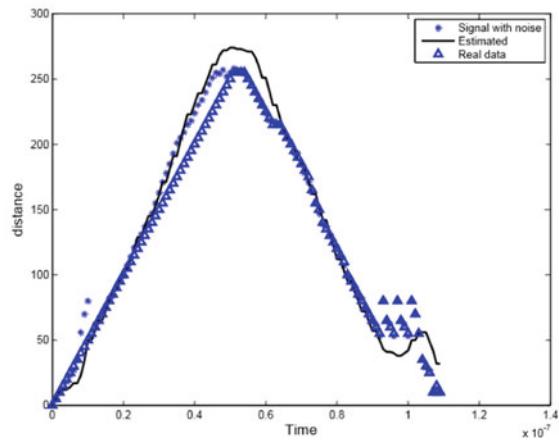
Fig. 1 Architecture and tracking performance



(a) Kalman gain and error co-variance estimate with pipelining and time sharing architecture



(b) Architecture for output estimate of Kalman filter



(c) Tracking performance with parabolic path

By increasing the word lengths of the input and output data, further the error can be reduced. The proposed reduced scalar Kalman filter algorithm requires four subtractors less than the algorithm proposed in [13]. The time-shared pipelined parallel architecture implemented using the proposed algorithm requires two subtractors less than the architecture proposed in [14]. The worst case computational delay obtained is 6.881 ns with a maximum clocking rate of 145 MHz. The total power consumption at 100 MHz is 0.128 W, and the leakage power consumed is 0.043 W.

6 Conclusion

A reduced scalar Kalman filter algorithm is proposed to meet the growing demands in the field of tracking applications. The simplified way of estimating posterior error co-variance estimate reduces the computational complexity and area cost for real-time tracking systems. The time-shared pipelined parallel architecture increases the speed of processing and sharing of the hardware resources greatly reduces the power and area requirement. The architecture requires nine adders, three subtractors, six multipliers, two dividers, a few pipelined registers and a control unit. The tracking error between the actual and the estimated output for a parabolic path is found to be 1.98%.

References

1. J. Gunnarsson, L. Svensson, L. Danielsson, F. Bengtsson, Tracking vehicles using radar detections, in *Proceedings of IEEE Intelligent Vehicles Symposium* (2007), pp. 296–302
2. S. Blackman, R. Popoli, *Design and Analysis of Modern Tracking Systems* (Artech House, Norwood, MA, 1999)
3. Y. Bar-Shalom, W.D. Blair, *Multitarget-Multisensor Tracking Volume III: Applications and Advances* (Artech House, Norwood, MA, 2000)
4. R.E. Kalman, A new approach to linear filtering and prediction problems. *J. Basic Eng.* **82D**, 34–45 (1960)
5. A. Honarbach, P. Rauschert, A. Kummert, Data fusion in wireless sensor networks: a new application for Kalman filtering, in *TENCON 2004. IEEE Region 10 Conference, 2004*, vol. 1 (2004), pp. 511–514
6. C. Hongyang, D. Ping, X. Yongjun, L. Xiaowei, A robust algorithm with biased extended Kalman filtering of TDOA data for wireless sensor networks, in *WEEE Proceedings, WCNM* (2005)
7. S. Ci, H. Sharif, Performance comparison of Kalman filter based approaches for energy efficiency in wireless sensor networks, in *IEEE Conference on Computer Systems and Applications* (2005)
8. D.-J. Wan, J.-C. Fang, Q. Wang, *Theory of GPS Dynamic Filtering and Its Application* (Jiangsu Science and Technology Press, Nanjin, 2000)
9. G. Welch, G. Bishop, An introduction to the Kalman filter, in *SIGGRAPH, Course 8* (2001)
10. K.D. Rao, G. Sridhar, Improving performance in pulse radar detection using neural networks. *IEEE Trans. Aerosp. Electron. Syst.* **31**, 1193–1198 (1995)

11. L.P. Maguire, G.W. Irwin, Transputer implementation of Kalman filters, in *IEEE Proceedings* (1991), pp. 355–362
12. S.M. Bozic, *Digital and Kalman Filtering: An Introduction to Discrete-Time Filtering and Optimum Linear Estimation* (Edward Arnold, 1995)
13. C.R. Lee, Z. Salcic, A fully-hardware-type maximum parallel architecture for Kalman tracking filter in FPGAs, in *IEEE Proceedings on Information, Communication, and Signal Processing* (1997)
14. Z. Mehri, M. Ghantous, M. Bayoumi, M. Bayoumi, A El Desouki, A fully-pipelined parallel architecture for Kalman tracking filter, in *IEEE Proceedings of CAMP* (2006), pp. 81–86

Shape Optimization of Beams with Scaled Boundary Finite Element Method and B-Splines



Kandula Eswara Sai Kumar, A. Sundaramoothy, Sundararajan Natarajan, and Sourav Rakshit

Abstract Shape optimization is a method to find the optimal shape of a structure by minimizing the objective function, i.e., volume and/or compliance under the design and limit constraints. The boundary of the structure is discretized using B-splines which allow local control and these control points are taken as design variables for the optimization formulation. The scaled boundary finite element method (SBFEM) is a semi-analytical approach, which has the combined advantage of both finite element method and the boundary element method. The SBFEM discretizes the boundary of the element, which reduces the computational domain size by one. Both the B-splines and the SBFEM discretizes the boundary of the structure, so the control points of the B-spline and the nodes of the SBFEM discretization can coincide to form the design variables. In this paper, we design a cantilever beam by using the shape optimization with the B-splines and the SBFEM. The cantilever beam is designed for minimum volume under a UDL and a point load with displacement constraint. The results obtained by the FEM and the SBFEM are compared.

Keywords Shape optimization · B-splines · Finite element method (FEM) · Scaled boundary finite element method (SBFEM)

1 Introduction

The shape optimization is the process of finding the optimal geometry of the structure, such that structure response is fulfilled best under some design constraints. The design objective and the constraints are typically weight, compliance, maximum allowable

K. E. S. Kumar · A. Sundaramoothy · S. Natarajan · S. Rakshit (✉)
Indian Institute of Technology, Chennai 600036, India
e-mail: srakshit@iitm.ac.in

K. E. S. Kumar
e-mail: me15d417@smail.iitm.ac.in

S. Natarajan
e-mail: snatarajan@iitm.ac.in

stress, and maximum displacement of the structural component [1]. Shape optimal design plays a vital role in industries since the performance efficiency is improved by choosing the optimal shape. The important applications of shape optimal designs are the automobile industries, aircraft industries, etc., where weight reduction is the most important objective to achieve the high efficiency of power transfer and less fuel consumption [2].

Haftka et al. [3] have given a treatise of shape optimization of structures. They discussed several steps in the shape optimization process, such as geometric model description, selection of the objective functions and design variables, finite element model generation and optimization algorithms. Bhavikatti et al. [4] used polynomial functions where the coefficient of polynomials is taken as the shape design variables. These variables describe the shape of structures. Ding [5] reviewed that use of the polynomials with control nodes for shape representation can reduce the total number of shape variables, but can result in oscillatory boundary shape due to numerical instability with higher-order curves. Wall et al. [6] developed a method by combining isogeometric analysis with shape optimization. This idea eliminates the problem of properly connecting the design model. Mukesh et al. [7] formulated an aerodynamic shape optimization process and solved starting from NACA 2411 airfoil. They have validated the formulation by conducting the experiments. Bandara et al. [8] solved the isogeometric shape optimization for shell structures using multiresolution subdivision surfaces. They have shown that the formulation prevents the appearance of sub-optimal jagged structures.

The present work focuses on finding the optimal shape of cantilever under different loading conditions. The boundary of the structure is represented using open uniform B-spline curve. Both the finite element method (FEM) and scaled boundary finite element method (SBFEM) is used to solve the equilibrium equation. We use MATLAB fmincon function to solve the shape optimization problem. The optimal shape is obtained for volume minimization under the displacement constraint using both the FEM and SBFEM. We compare the results obtained using these two methods.

2 Problem Formulation

This section defines the shape optimization problem and gives a brief introduction on B-splines and the SBFEM.

2.1 Structural Shape Optimization

Shape optimization is the process of finding the shape design variable \mathbf{x} , while minimizing the objective function $W(\mathbf{x})$ under the constraint functions $g_i(\mathbf{x}), h_i(\mathbf{x})$. In this paper, the objective function $W(\mathbf{x})$ is volume of the beam and the constraint is on maximum deflection of the beam. The shape optimization problem can be stated as

$$\begin{aligned}
& \min W(\mathbf{x}) \\
& \text{subject to } g_i(\mathbf{x}) \leq \bar{g}_i(\mathbf{x}) \\
& h_j(\mathbf{x}) = \bar{h}_j(\mathbf{x}) \\
& \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U
\end{aligned} \tag{1}$$

2.2 Boundary Representation with B-spline Curve

The design boundaries are represented by open uniform and uniform B-spline curves. It has advantages like degree of the curve not depend on number of control points and it offers local control. A B-spline curve is represented by blending functions and set of control points

$$P(t) = \sum_{i=0}^n N_{i,k}(t) P_i \quad (t_{k-1} \leq t \leq t_{n+1}) \tag{2}$$

where P_i is the set of control points and $N_{i,k}$ is blending function given by de-Boor algorithm [9]

$$N_{i,k}(t) = \frac{(t - t_i)N_{i,k-1}(t)}{t_{i+k-1} - t_i} - \frac{(t_{i+k} - t)N_{i+1,k-1}(t)}{t_{i+k-1} - t_{i+1}} \tag{3}$$

and

$$N_{i,k}(t) = \begin{cases} 1, & t_i \leq t \leq t_{i+1}, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

where n is a number of control points, k is order of B-spline curve. In this work, a knot vector-based approach is used to construct the B-spline. The B-splines are assumed to be weighted sum of the control points P_i .

$$P(t) = \sum_{i=0}^n P_i B_{n,i}(t) \tag{5}$$

where $B_{n,i}$ is weight function and $P(t)$ is traced when parameter t varies over the interval of unit one from integer value K to next integer value $K + 1$. The K values are called knots of the B-spline curve [10].

2.3 Scaled Boundary Finite Element Method (SBFEM)

The SBFEM is formulated to take advantage of analytical techniques available to solve ordinary differential equations. The SBFEM discretizes the boundary of the element and has the advantage of both the finite element method and the boundary element method. It reduces the order of computational domain size by one. The scaled boundary finite element method reduces the governing partial differential equations to a system of ordinary differential equations, which can be solved analytically [11]. Figure 1 is a polygon structure whose boundary is discretized by scaled boundary 1D elements. The scaled boundary equation in displacement for two dimensions is written as,

$$\mathbf{E}_0 \xi^2 \mathbf{u}(\xi)_{,\xi\xi} + (\mathbf{E}_0 - \mathbf{E}_1 + \mathbf{E}_1^T) \xi \mathbf{u}(\xi)_{,\xi} - \mathbf{E}_2 \mathbf{u}(\xi) = 0 \quad (6)$$

where $\mathbf{E}_0, \mathbf{E}_1, \mathbf{E}_2$ are the coefficient matrices obtained by assembling the elemental coefficient matrices and these are defined as

$$\mathbf{E}_0 = \int_{-1}^{+1} \mathbf{B}_1(s)^T \mathbf{D} \mathbf{B}_1(s) |J(s)| ds$$

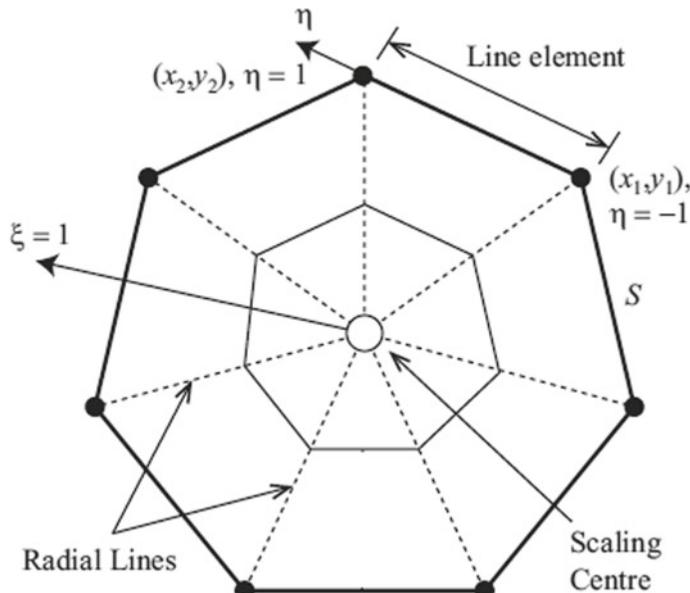


Fig. 1 Polygonal domain having seven scaled boundary line elements along the boundary

$$\begin{aligned}\mathbf{E}_1 &= \int_{-1}^{+1} \mathbf{B}_2(s)^T \mathbf{D} \mathbf{B}_1(s) |J(s)| ds \\ \mathbf{E}_2 &= \int_{-1}^{+1} \mathbf{B}_2(s)^T \mathbf{D} \mathbf{B}_2(s) |J(s)| ds\end{aligned}\quad (7)$$

From the Eq. 6, the Hamiltonian matrix \mathbf{Z} is written as,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{E}_0^{-1} \mathbf{E}_1^T & -\mathbf{E}_0^{-1} \\ -\mathbf{E}_2 + \mathbf{E}_1 \mathbf{E}_0^{-1} \mathbf{E}_1^T & -\mathbf{E}_1 \mathbf{E}_0^{-1} \end{bmatrix} \quad (8)$$

Now find the eigenvalues λ_i and eigenvectors ϕ_i of the \mathbf{Z} matrix. \mathbf{Z} has ‘ $2n$ ’ modes, in that the first ‘ n ’ modes correspond to the modal displacements and next ‘ n ’ modes are for the modal force vector. First ‘ n ’ modes of the nodal displacements are designated by ϕ , and the next ‘ n ’ modes are designated by \mathbf{Q} [12]. The stiffness matrix is calculated as:

$$\mathbf{K} = \mathbf{Q} \phi^{-1} \quad (9)$$

2.4 Mesh Convergence Study

Figure 2 is a cantilever beam with length ‘ l ’ = 1 m, height ‘ h ’ = 0.4 m, and load ‘ P ’ = 100 kN. This beam is solved using both the FEM and SBFEM to understand the convergence behavior of the methods. The convergence curve is shown in Fig. 3. The displacement is compared between the FEM and SBFEM for different number

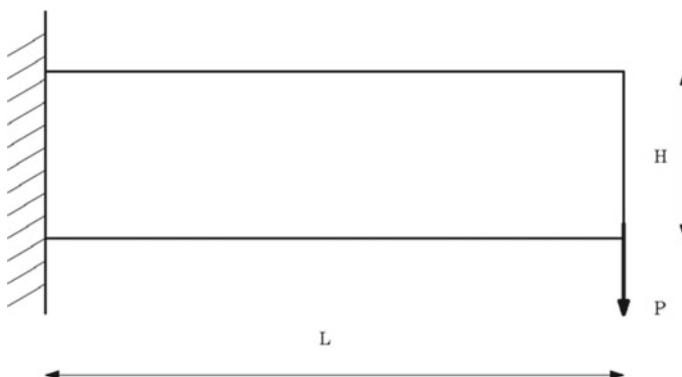


Fig. 2 Cantilever beam with length ‘ l ’ = 1 m, height ‘ h ’ = 0.4 m and load ‘ P ’ = 100 kN

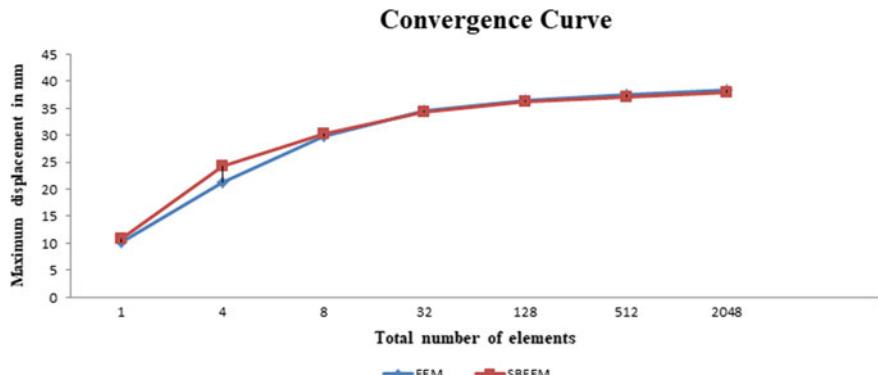


Fig. 3 Convergence study of the cantilever beam for FEM and SBFEM

of elements. We observed that, the relative error between FEM and SBFEM is less than 1.2% after 8 elements. Both the FEM and SBFEM results are matching after 32 elements.

2.5 Algorithm

Figure 4 shows the algorithm of shape optimization problem. The structural domain undergoes scaled boundary analysis once it discretizes the domain. The governing equation is solved by the SBFEM for displacements. The objective function and the constraints are calculated using the displacements. The fmincon takes these as input to solve the shape optimization problem. The convergence check is done for each iteration. We plot the optimal shape after it satisfies the convergence check.

3 Results

The shape optimization with SBFEM is applied to the cantilever beam under uniformly distributed loading conditions for volume minimization, under the plane stress condition. The material properties of the beam are Young's modulus $E = 200$ GPa, Poisson's ratio $\nu = 0.3$.

3.1 Example—Cantilever Beam Under UDL

Figure 5 shows the cantilever beam under UDL. The geometric details of the beam are: length ' l ' = 1 m, height ' b ' = 0.1 m, the load of 10 kN/m is applied with the distance ' d ' = 0.2 m.

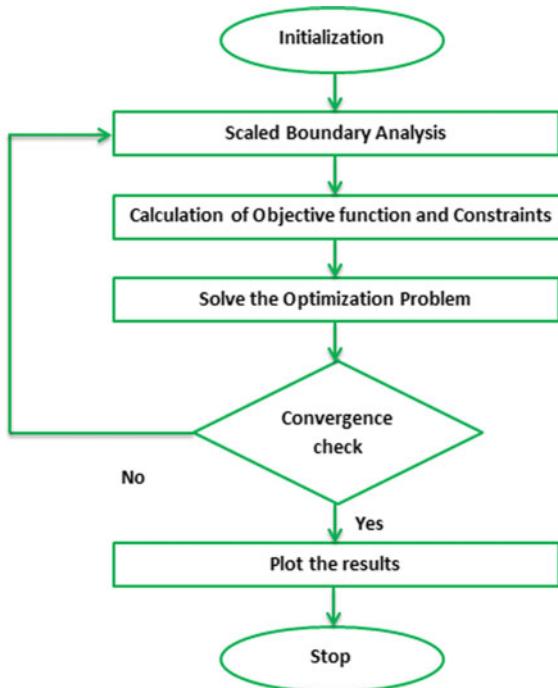


Fig. 4 Flowchart for solving shape optimization problems using the SBFEM

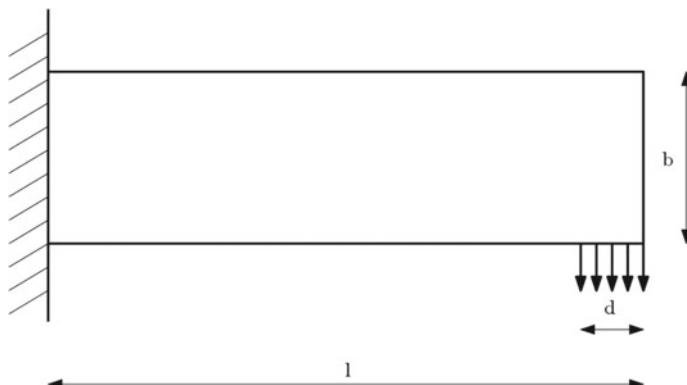


Fig. 5 Cantilever beam with length ' l ' = 1 m and height ' b ' = 0.1 m under a UDL of 10 kN/m with ' d ' = 0.2 m distance from the free end of cantilever beam

Figure 6 shows SBFEM mesh representation of cantilever beam. The beam is taken as a sub-domain and its boundary is discretized by a set of one-dimensional line elements. The boundary of the beam is discretized with 28 1D scaled boundary elements. The upper surface of the beam is controlled by the B-spline curve. The control points are updated in each iteration and the mesh coordinates are interpolated from the parametric equation of the splines as in Eq. 5. The cantilever beam has 11 design variables located on the upper surface. Figure 7 shows the finite element discretization of the domain with 80 2D elements. The shape optimization problem with the displacement constraint of cantilever beam under UDL is defined in Eq. 10. The objective is to minimize the total volume of the structure by satisfying the governing equation along with the displacement constraint. The maximum displacement is constrained to 10 cm. The shape optimization problem is:

$$\begin{aligned} \min \quad & V(\mathbf{x}) \\ \text{subject to } & \mathbf{Ku} = \mathbf{F} \\ & \max(|u|) \leq 0.01 \\ & 0.04 \leq |x| \leq 0.1 \end{aligned} \quad (10)$$

Figure 8 shows the optimal shape of the cantilever beam obtained using the FEM. The optimal solution obtained after 27 iterations, and reduced the volume to 74.7% of its initial volume. Figure 9 shows the optimal shape of the cantilever beam under UDL obtained using the SBFEM. The optimal solution obtained after 24 iterations, and reduced the volume to 65.97% of its initial volume.

Fig. 6 Representation of cantilever beam by the SBFEM mesh with 11 design variables on the top surface of the cantilever beam. The SBFEM mesh has 28 1D elements

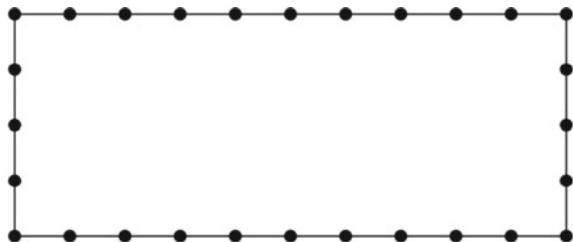
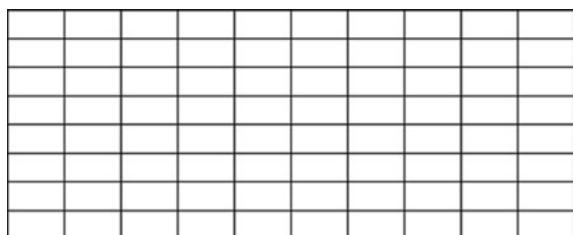


Fig. 7 Representation of cantilever beam by the FEM mesh with 11 design variables on the top surface and 80 2D elements for entire domain



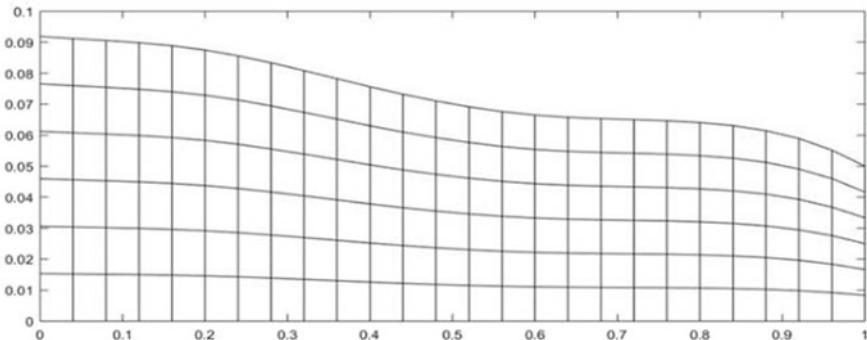


Fig. 8 Optimal shape of the cantilever beam under UDL obtained using FEM with 25.3% of volume reduction

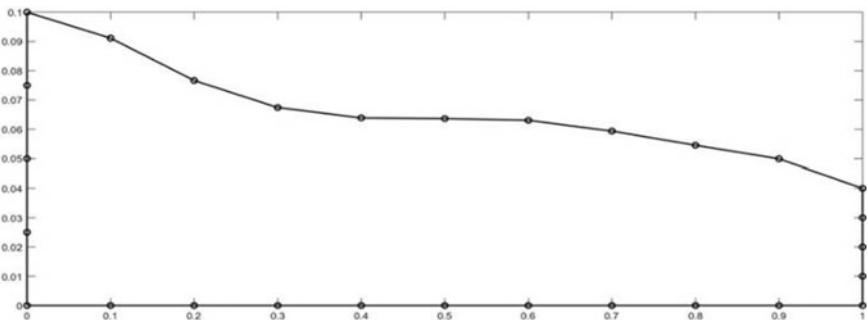


Fig. 9 Optimal shape of the cantilever beam under UDL obtained using SBFEM with 34.03% volume reduction

3.2 Example—Cantilever Beam with Point Load

Figure 10 shows the cantilever beam with point load. The geometric details of the beam is: length ‘ l ’ = 1 m and height ‘ h ’ = 0.2 m. The point load ‘ P ’ = 100 kN is applied at the middle point of the right end. The upper and lower surface of cantilever beam has 42 design variables. Figure 11 shows the mesh representation of the cantilever beam using SBFEM. The boundary of the beam is discretized by 48 1D elements while allowing both the top and bottom surfaces for design. The FEM discretization is shown in Fig. 12. The domain has 160 2D FE elements.

$$\begin{aligned}
 & \min \quad V(x) \\
 & \text{Subject to } \mathbf{Ku} = \mathbf{F} \\
 & \max(|u|) \leq 0.01 \\
 & 0.02 \leq |x| \leq 0.1
 \end{aligned} \tag{11}$$

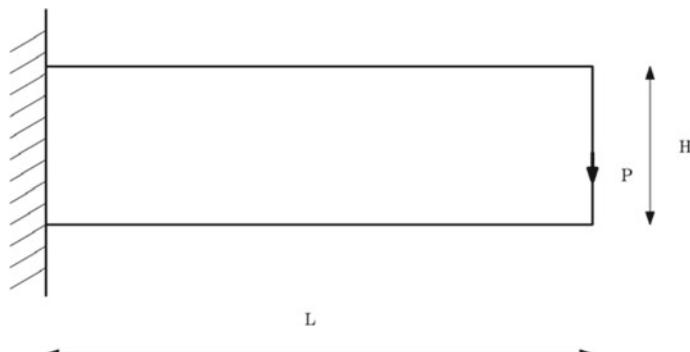
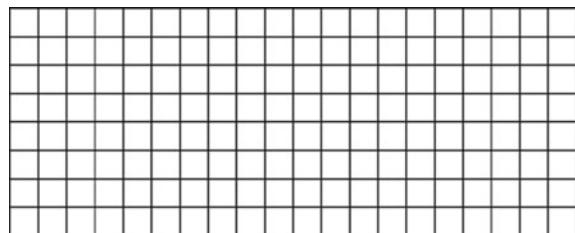


Fig. 10 Cantilever beam with length $L = 1$ m, height $H = 0.2$ m under a point load $P = 100$ kN

Fig. 11 SBFEM mesh representation of the cantilever beam with 48 1D elements for 42 design variables. The top and bottom surfaces of the cantilever beam are allowed to design



Fig. 12 FEM mesh representation of the cantilever beam with 160 2D elements for 42 design variables on both the top and bottom surfaces



The shape optimization problem for the example 2 is shown in Eq. 11. The objective is to minimize the total volume of the structure under the displacement constraint, i.e., the allowable displacement in the beam is 10 cm while satisfying the governing equation.

Figure 13 shows the optimal design of the cantilever beam under point load using SBFEM. The optimal solution has 41.74% volume reduction. Figure 14 is the optimal design obtained using FEM. The FEM has a volume reduction of 33.95%.

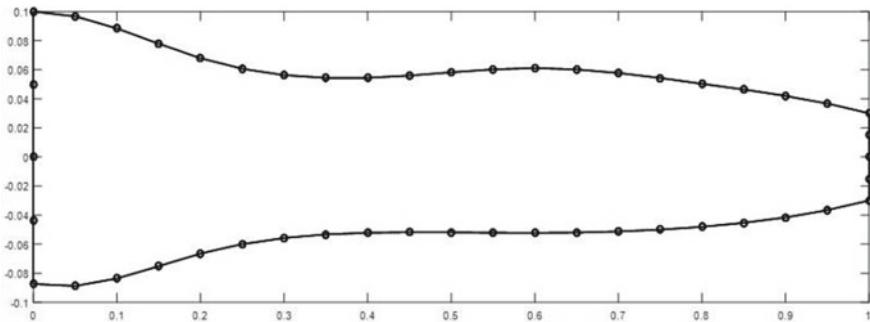


Fig. 13 Optimal design of cantilever beam under point load at end obtained using SBFEM with 41.74% volume reduction

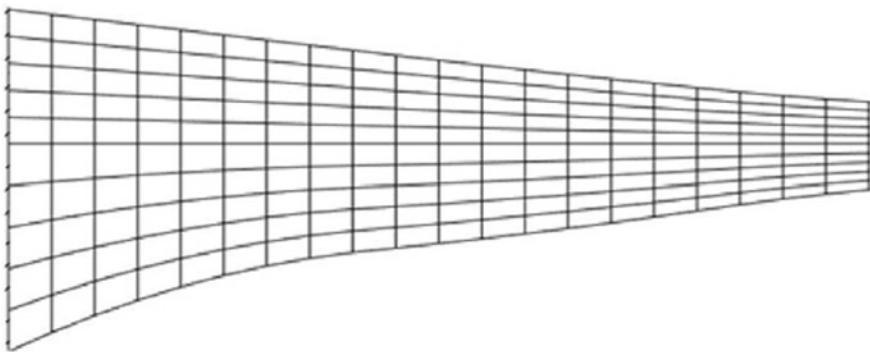


Fig. 14 Optimal design of cantilever beam under point load at end obtained using FEM with 33.95% volume reduction

4 Conclusions

The shape optimization with SBFEM is used to design the cantilever beam under UDL and point load. The value of volume reduction obtained by both SBFEM and FEM for the cantilever beam under UDL and point load is listed in Table 1. From Table 1, we observed that the values of volume reduction obtained using the SBFEM is better

Table 1 Comparison between the FEM and the SBFEM for the shape optimization of the cantilever beam

S. No.	Method	Problem definition	% of volume reduction	Time in sec
1	FEM	UDL	25.30	6.0498
2	SBFEM	UDL	34.03	3.8139
3	FEM	Point load	33.95	7.2115
4	SBFEM	Point load	41.74	5.2934

than the FEM for both the examples. We conclude that the SBFEM performs better for shape optimization of the beams for minimizing the volume under displacement constraint.

References

1. S.N. Skninner, H. Zare-Behtash, State-of-the-art in aerodynamic shape optimisation methods. *Appl. Soft Comput.* **62**, 933–962 (2018)
2. S. Yoshimoto, T. Stolarski, Y. Nakasone, *Engineering Analysis with ANSYS Software*. 2nd edn. (Butterworth-Heinemann Publications, 2018)
3. R.T. Haftka, R.V. Grandhi, Structural shape optimisation—A surey. *Comput. Methods Appl. Mech. Eng.* **57**(1), 91–106 (1986)
4. S. Bhavikatti, C. Ramakrishnan, Optimum shape design of shoulder fillets in tension bars and t-heads. *Int. J. Mech. Sci.* **21**(1), 29–39 (1979)
5. Y. Ding, Shape optimization of structures: a literature survey. *Comput. Struct.* **24**(6), 985–1001 (1986)
6. W.A. Wall, M.A. Frenzel, C. Cyron, Isogeometric structural shape optimization. *Comput. Methods Appl. Mech. Eng.* **197**, 2976–2988 (2008)
7. R. Mukesh, K. Lingadurai, U. Selvakumar, Airfoil shape optimization using non-traditional optimization technique and its validation. *J. King Saud Univ. Eng. Sci.* **26**(2), 191–197 (2014)
8. K. Bandara, F. Cirak, Isogeometric shape optimisation of shell structures using multiresolution subdivision surfaces. *CAD Comput. Aided Des.* **95**, 62–71 (2018)
9. R.H. Bartels, J.C. Beatty, B.A. Barsky, An introduction to splines for use in computer graphics and geometric modeling. (Morgan Kaufmann Publications, 1987)
10. D. Salomon, Curves and surfaces for computer graphics. (Springer Science and Business Media Publications, 2007)
11. S. Natarajan, E.T. Ooi, I. Chiong, C. Song, Convergence and accuracy of displacement based finite element formulations over arbitrary polygons: Laplace interpolants, strain smoothing and scaled boundary polygon formulation. *Finite Elem. Anal. Des.* **85**, 101–122 (2014)
12. A.J. Deeks, J.P. Wolf, A virtual work derivation of the scaled boundary finite element method for elastostatics. *Comput. Mech.* **28**(6), 489–504 (2002)

Innovative Design of Hydraulic Actuation System for Operator Fatigue Reduction and Its Optimization



Sukrit Mittal , Divyam Aggarwal , and Dhish Kumar Saxena

Abstract This paper deals with the proposition of a new design of hydraulic actuation systems (HAS), its modelling and subsequent optimization. The purpose of HAS is to provide the operator, a linkage for actuating large equipment like construction and manufacturing machines, responsible for doing mechanical jobs such as lifting tons of materials and high-force/pressure applications. Owing to the government regulations and economic and reliability constraints, the use of simple hydromechanical interfaces is prevalent in the industry. Furthermore, to ease out the operator's effort, another interface called as pilot control HAS is dominant among the industrial practices. However, pilot control prevents the operator from controlling the equipment directly, affecting the precision of the operation, which further requires costly electronic interventions. The proposed HAS, which accounts for operators fatigue (by providing hydraulic assistance) and precision (by directly controlling the HAS), provides a cost-effective solution. The proposed HAS, with multiple settings of design parameters, fulfils the varying input effort requirements in a variety of applications. Due to government regulations around safety and operability, it is required to provide a minimum input effort for a particular equipment/application but the effort reduction can be achieved with respect to the user preferences in the proposed HAS. Moreover, it is known that a human mind (operator) could anticipate/calibrate the movements varying only in a linear fashion. The novelty here relates to the modelling of proposed system design and exploration of relationship between operator fatigue and additional cost of manufacturing through single-and/or multi-objective optimization.

S. Mittal · D. Aggarwal · D. K. Saxena

Department of Mechanical & Industrial Engineering, Indian Institute of Technology Roorkee,
Roorkee, India

e-mail: smittal1@me.iitr.ac.in

D. Aggarwal

e-mail: daggarwal@me.iitr.ac.in

D. K. Saxena

e-mail: dhish.saxena@me.iitr.ac.in

Keywords Directional-control valve • Hydraulic actuation system • Hydraulic assistance • Operator fatigue reduction • Optimization

1 Introduction

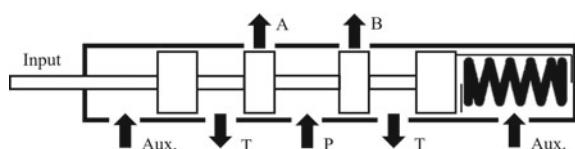
Directional-control valve (DCV) is the most fundamental component of any hydraulic actuation system, which lays the foundation of any hydraulic/pneumatic equipment. The most commonly used actuation system is a typical three-position DCV, attached to a double-acting hydraulic cylinder and/or to a dual-direction hydraulic motor. A basic schematic of a manually actuated DCV (along with auxiliary chambers for hydraulic actuation) is shown in Fig. 1. The topic of interest in this mainstream but widely used component is the spring which makes the spool return to its neutral position on the release of the applied force [3]. In principle, this return of valve motion is governed by a dual-acting pre-compressed spring, i.e. it gets compressed regardless of the direction of the spool motion from the neutral position.

There are a number of ways in which a DCV can be actuated, such as manual (as shown in Fig. 1), mechanical (using cams, rollers, etc.), hydraulic (pilot-controlled valves) and electronic (using electromechanical solenoids). Pilot control valves use secondary spool valves to direct the fluid towards main spool valve chambers (as shown in Fig. 1) and govern their motion as per requirement. But, there are limitations to each of these mechanisms as enlisted below:

1. *Manual*: High operating force, as the spring reaction force increases with compression.
2. *Mechanical*:
 - (a) Their application is limited to mechanisms which operate in cycles.
 - (b) Cams tend to wear.
3. *Hydraulic*: Pilot control valves lose direct control over the final operation being done, i.e. the operator becomes unable to precisely control the required operation.
4. *Electronic*: The solenoids are unable to generate high actuation forces, i.e. their application is limited.

Due to cost considerations, the commonly and most widely used mechanism is still manual actuation, regardless of the operator's fatigue due to high operating force. Owing to the government regulations and safety regards, the actuation force at neutral position, i.e. the spring stiffness or pre-compression cannot be modified to account for operator's efforts [6]. Motivated from this, some industries have replaced

Fig. 1 DCV schematic (P : Input from pump; T : Exhaust to tank; A & B : Outputs for operation; $Aux.$: Input to auxiliary chambers (caps))



manually actuated DCVs with pilot control valves, thus increasing the product price as well. As mentioned earlier, the operator loses direct control over the operation in case of pilot-controlled HAS. This is inherited from design as the signal is first transmitted from operator's lever to the secondary valve and then to the main spool valve. To overcome this particular aspect, there have been a few developments where electronic actuation is involved at the intermediate stage [8]. The idea is to sense the operator's signal (hand/lever movement) and accordingly adjust the pilot (secondary) valve to have a precise control over the final hydraulic movement/operation.

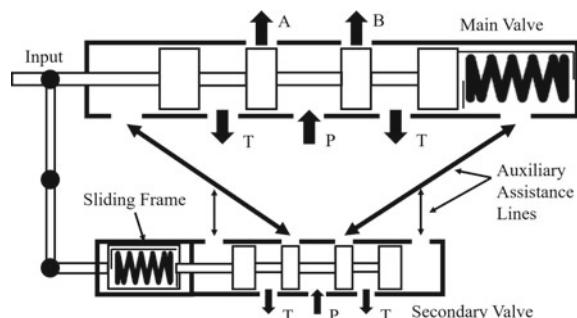
2 Proposed System Design

The design intent is to meet the following requirements:

5. Provide hydraulic assistance, i.e. lower actuation/operating force.
6. Maintain direct control of the operator over the function.
7. Maintain initial actuation force as per safety standards.

In this paper, the design of a new HAS, i.e. a system to actuate a DCV is proposed. It consists of two interconnected spool valves, i.e. the main spool valve and the secondary spool valve. The main spool is fixed with a pre-compressed spring which compresses with motion in either of the directions and has caps (chambers) on both sides through which assistance pressure may act on it. The secondary spool also holds a similar configuration, but is like a small-scale version. Also, as shown in Fig. 2, the compression spring of the secondary spool is contained in a sliding frame which can slide freely along the spool movement axis. The spring chamber is attached to the secondary spool on one side and is mechanically linked to the main spool on the other side, through a lever mechanism (acting as a fulcrum). The auxiliary lines are shown by dual-sided arrows in Fig. 2, which connect the outputs of secondary valve to the aux. chambers of both main as well as secondary spool valves. The complete system can be operated directly by the operator through a lever attached to the input (as marked in Fig. 2). The secondary spool takes pressure input from a reduced pressure line (P , available from the pump) and directs the required pressure

Fig. 2 New HAS design



to the main spool caps. Also, the secondary spool has the caps on both sides as well subjected to the same pressure as the main spool caps for direct feedback sensing.

2.1 Working

The motion of the spool can be done on the either sides, though the response mechanism acts in a similar way for both. When the main spool shifts towards right, the main spool spring is compressed. Due to the lever attachment, the sliding spring frame shifts towards the left side, and consequently the secondary spool. This allows the system line pressure to be directed into the left side assistance gallery. Eventually, the left side chambers on both the main and secondary spools are exposed to pressure, which tries to move both the spools towards the right. On the main spool side, this pressure acts like an assistance since its original motion was also towards the right side. Whereas for the secondary spool, this pressure acts against the motion as the original spool motion was towards the left side. This compresses the secondary spring and makes the secondary spool move towards the right side till the assistance gallery is completely blocked again. Thus, the secondary spool acts as a pressure control valve allowing the pressure to be built as per the movement (and thus, the assistance requirement). Hence, the design (in principle and in itself) takes care of the assistance requirement according to the main spool movement and provides consistent performance through real-time feedback sensing.

2.2 Mathematical Modelling

The formulation of the optimization problem for the proposed system design consists of creating a mathematical model of the system by shaping the objective function(s) and defining the constraint(s). The proposed system design is aimed at maximizing the force reduction (assistance provided to the user), while minimizing the additional cost of manufacturing. This leads to the formulation of a multi-objective optimization problem (MOOP). For better clarity, the design parameters used in this problem formulation are mentioned in Table 1 while the design variables are mentioned in Table 2.

Table 1 Design parameters used in the problem

Symbol	Term	Value
k_1	Main spring stiffness (N/m)	3383.31 N/m
d_1	Main spool diameter (mm)	18.00 mm
x_1	Main spring pre-compression (m)	20.54 mm
y_1	Main spool displacement (m)	6.00 mm

Table 2 Design variables used in the problem

Symbol	Term
k_2	Main spring stiffness (N/m)
d_2	Main spool diameter (mm)
x_2	Main Spring pre-compression (m)
m	Main spool displacement (m)

The values of the above-mentioned design parameters are taken from the state-of-the-practice main spool valve. Using these design parameters, variables and force analysis [7], the first objective function, i.e. minimization of force F at input (or maximization of the hydraulic assistance), is formulated as follows:

$$\begin{aligned} \text{Min } F(d_2, x_2, k_2, m) = & y_1 \left(k_1 + \frac{k_2}{m} \left\{ \frac{1}{m} - \left(\frac{d_1}{d_2} \right)^2 \right\} \right) \\ & + k_1 x_1 + k_2 x_2 \left\{ \frac{1}{m} - \left(\frac{d_1}{d_2} \right)^2 \right\} \end{aligned} \quad (1)$$

The second objective function is the minimization of the additional cost of manufacturing which is directly dependent on the additional material and processing requirements which, in turn, depends upon the size of the secondary spool. Hence, the formulation of second objective function is given by

$$\text{Min } c(d_2, m) = a + b \left(\frac{d^2}{m} \right) \quad (2)$$

where a and b are constants and can be neglected while minimizing the cost function.

In principle, the proposed system design is capable of providing a full user assistance, i.e. $F = 0$, which is not desirable as there has to be a realizable force against which the user acts. Hence, to restrict the force reduction (first objective), a nonzero user-preferred force value, f , is used. To demonstrate the efficacy of the proposed system design, this force value f is assumed to be 30 N.¹ Thus, the only design constraint possible for the optimization problem at hand is as follows:

$$F - f \geq 0 \quad (3)$$

Further, the resulting side constraints (estimated from the state-of-the-practice manufacturing capabilities) for governing the design variables are given in (4).

¹A sensitivity analysis of this force on the optimization process could be carried out in the future work.

$$\begin{cases} 6 \text{ mm} & \leq d_2 \leq 18 \text{ mm} \\ 2 \text{ mm} & \leq x_2 \leq 5 \text{ mm} \\ 1000 \text{ N/m} & \leq k_2 \leq 2000 \text{ N/m} \\ 0.75 & \leq m \leq 3 \end{cases} \quad (4)$$

2.3 Computational Experiments

The resulting MOOP for the proposed system design is comprised of two objective functions, given in Eqs. (1) and (2), and design and side constraints, given in Eqs. (3) and (4), respectively. This MOOP is optimized using NSGA-II [1], which is one of the most widely adopted multi-objective evolutionary algorithm (MOEA).

The reasons for preferring a genetic algorithm over traditional gradient-based algorithms are given as follows:

1. In second objective function (Eq. 2), it can be observed that the constants a and b are unknown. Deploying a classical optimization technique would have required these constants to be known, since it converts the original problem into a single-objective optimization problem using weighted sum method [9]. On the other hand, a pure multi-objective optimization method allows these constants to be neglected since it does not affect the trade-off solution [5].
2. In the formulated problem statement, the first objective is a force function while the second objective is a cost function, both having different physical significance. Also, their scales cannot be compared, which rules out the weighted sum approach [2, 4].

The GA parameters used to in this problem are as follows: No. of Generations = 1000; Population Size = 100; Crossover Probability = 0.9; Mutation Probability = 0.1. To observe the effect of random number seeds on the optimal solution, experiments are performed with 20 different random number seeds (varying uniformly between 0 and 1). The experimental settings and results are summarized in Table 3, and the Pareto front (PoF) achieved in one of the runs is shown in Fig. 3.

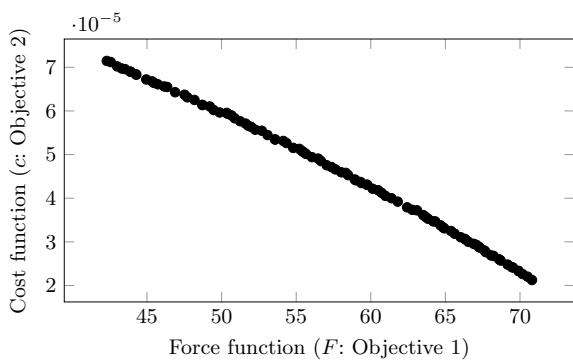
3 Conclusion

The aforementioned work presents a novel design for hydraulic actuation and assistance mechanism and a systematic method to optimize its design parameters. In state-of-the-practice DCV (without assistance mechanism), the actuation force at the neutral position is required to be 69.5 N. This value remains preserved after the implementation of proposed design, since there is no assistance at neutral position. The major contribution of the proposed design is linked to the operator's fatigue in

Table 3 Two extreme solutions (A for min. force and B for min. cost) observed over 20 different seed runs. For each solution, the top row shows the avg. values while the bottom row shows the best values

Sol.		F (N)	c (X e-05 units)	d_2 (mm)	x_2 (mm)	k_2 (N/m)	m
A	$x^- \pm \sigma$	42.309 ± 0.007	7.197 ± 0.168	7.346 ± 0.008	2.000 ± 0.004	1017.259 ± 11.226	0.750 ± 0.008
	Best	42.301	7.104	7.299	2.000	1009.245	0.750
B	$x^- \pm \sigma$	70.803 ± 0.001	2.126 ± 0.007	7.987 ± 0.001	2.000 ± 0.003	1000.591 ± 0.417	2.999 ± 0.052
	Best	70.800	2.125	7.984	2.000	1000.059	2.999

Fig. 3 PoF on a random seed between force function and cost function



which the user effort (applied on the lever) is reduced continuously during the operation by providing assistance via the proposed secondary spool valve. The rationale for this is the tendency of the lever force to increase drastically with spring compression during the operation. Another contribution of this work relates to the multi-objective optimization of the user effort reduction with the additional cost of manufacturing for the proposed design. It is found that these objectives are conflicting in nature with respect to the parameters of the proposed design. The resulting PoF provides a set of non-dominated design solutions which could be adopted as per the user preferences for the maximum effort reduction and the manufacturing cost. From the computational experiments, the design parameters are optimized in a way to assist $\sim 50\%$ of the state-of-the-practice user effort (~ 85 N) at the extreme position. This corresponds to the minimum force solution, as observed on the left-most end of the PoF.

References

1. K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **6**(2), 182–197 (2002). <https://doi.org/10.1109/4235.996017>
2. C.M. Fonseca, P.J. Fleming, An overview of evolutionary algorithms in multiobjective optimization. *Evol. Comput.* **3**(1), 1–16 (1995). <https://doi.org/10.1162/evco.1995.3.1.1>
3. K. Joshi, C. Rajsekhar, Study of hydraulic directional control valve with cylinder performance using matlab simulink. *Int. J. Innov. Eng. Technol. (IJIET)* **4**(1) (2014)
4. S.A. Khan, S. Rehman, Iterative non-deterministic algorithms in on-shore wind farm design: a brief survey. *Renew. Sustain. Energy Rev.* **19**, 370–384 (2013). <https://doi.org/10.1016/j.rser.2012.11.040>
5. O. Kramer, *Genetic Algorithm Essentials*, 1st edn. (Springer Publishing Company, Incorporated, Heidelberg, 2017)
6. S. Mandal, B. Ghosh, A. Prasad, Design of a spool type 4 way 3 position directional control valve for use in tractor hydraulics system. *Int. J. Appl. Sci. Eng. Res. (IJASER)* **5**(6) (2016). <https://doi.org/10.6088/ijaser.05043>
7. V. Mishra, S. Chandel, Modeling & parameterization of directional control valves using matlab-simulink/sim hydraulics. *Int. J. Eng. Technol. Sci. Res. (IJETSR)* **5**(3) (2018)

8. S. Mittal, R. Singh. A simplified approach towards draft control in hydraulic machines for component/cost reduction, in *SAE Technical Paper*. SAE International (2019). <https://doi.org/10.4271/2019-26-0102>
9. L. Xiujuan, S. Zhongke, Overview of multi-objective optimization methods. *J. Syst. Eng. Electron.* **15**(2), 142–146 (2004)

Design of FSAPDS Using Multidisciplinary Design Optimization (OpenMDAO)



Rishav Kanth and Basant Kumar Gupta

Abstract Multidisciplinary design analysis and optimization (MDAO) has achieved a fair amount of success in the aerospace industry catering to the need of integration of highly complex disciplines in the same. Such a complexity is shared to some extent in the designing of an ammunition. In this paper, we have tried to develop an MDAO framework for the design of an armor piercing fin stabilized discarding sabot (FSAPDS). The optimization framework contains a physics solver that needs to account for all the physics that can be modeled right from the combustion of the propellant to the target penetration. The modules that the framework in this paper integrates are design assembly, internal ballistics, in-bore travel, the trajectory of the ammunition modeled by two DOF equations and finally target penetration. These individual modules have been developed in-house. The final integration and the development of the framework have been performed using Python. Optimization has been performed using OpenMDAO. The structure of the framework has been depicted using an extended design structure matrix (XDSM). This paper explores the possibility of attempting to model such a complex system and to use optimization tools to narrow down the huge design space generated by the large number of design variables present in the physical system.

Keywords MDO · OpenMDAO · FSAPDS · Kinetic energy projectile · Optimization · Genetic algorithm · Design space search

1 Introduction

Fin stabilized armor piercing discarding sabot (FSAPDS) is a kinetic energy projectile launched from a smooth-barrel gun, used against armored targets. The purpose of the FSAPDS is to hit the target and penetrate its armor. The ammunition assembly consists of three basic components—the penetrator, the sabot and the fin. The penetrator is a heavy tungsten alloy component meant to pierce the target armor, while the

R. Kanth (✉) · B. K. Gupta
Zeus Numerix Private Limited, Pune, India
e-mail: rishav.kanth@zeusnumerix.com

fin provides the required in-flight stability. Being a sub-caliber ammunition, the sabot is not a part of the projectile, rather it provides the surface area required for the force generation within the barrel and transmits it to the penetrator via buttress threads. At the time of ignition, the gaseous pressure at the base of the penetrator and sabot pushes the assembly forward. During the in-bore travel, obturation is provided by a driving band, while a centering band is used to prevent any eccentricity between the center of application of pressure and the centerline of the assembly. At the muzzle exit, the sabot gets discarded, while the penetrator-fin subassembly is projected forward. For the purpose of the exercise presented in this paper, the penetrator has been divided into three sections—fore, sabot–penetrator–interface and aft, in order to assign separate design variables to these sections. The design variables for the assembly are shown in Figs. 1 and 2 and in Table 1. The sabot profile has been defined in a way that it consists of five curves, revolved about the centerline of the

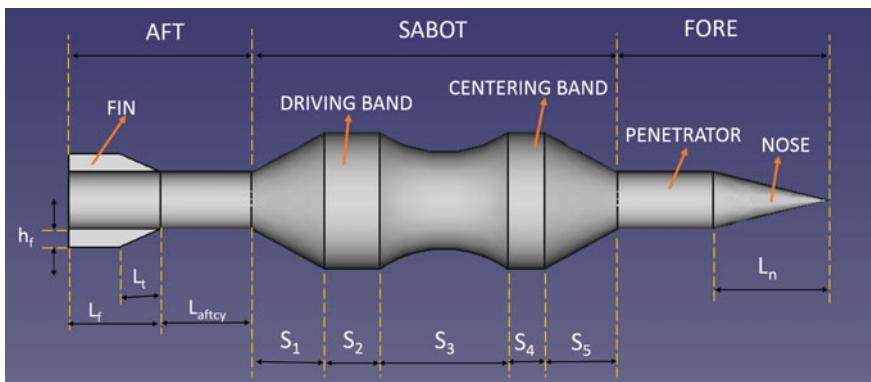


Fig. 1 Design variables from the side view of the FSAPDS assembly

Fig. 2 Design variables from the back view of the FSAPDS assembly

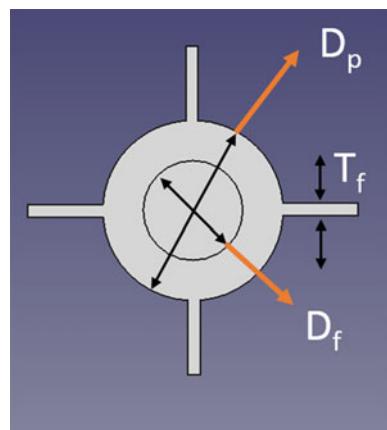


Table 1 Description of design variables

	Symbol	Description
1	L_n	Length of the nose of penetrator
2	L_f	Length of the fin
3	L_t	Length of the tapered section of fin
4	H_f	Height of the fin
5	T_f	Thickness of fin
6	D_p	Diameter of the penetrator
7	D_f	Diameter of cylindrical section of the fin
8	S_1	Length of the first straight line non-parallel to x -axis
9	S_2	Length of first straight line parallel to x -axis
10	S_3	Length of the circular curve
11	S_4	Length of second straight line parallel to x -axis
12	S_5	Length of second straight line non-parallel to x -axis
13	L_{aftcy}	Length of penetrator aft excluding the fin

sabot, two straight lines parallel to x -axis, two straight lines with some slope to x -axis and one circular curve passing through three predetermined points on the sabot profile. Only the lengths of these curves on the x -axis are the design variables for the sabot, shown as S_1, S_2, S_3, S_4 and S_5 in Fig. 1. In total, there are 13 design variables which seem enough to define a simplistic model of FSAPDS assembly.

2 Multiple Disciplines

There are four disciplines present in the MDAO framework—internal ballistics, in-bore travel, two-DOF and target penetration. For a given FSAPDS geometry, internal ballistics calculates the pressure–time history at the base of the assembly during its trajectory within the barrel. This solver is based on a lumped parameter model and a propellant combustion model derived from the literature [1]. The in-bore travel, using the pressure–time history from the internal ballistics, iteratively calculates the various stresses generated in each section of every component of the assembly inside the barrel to ascertain the factors of safety of each component. It has been derived from the models present in the literature [2]. The two-DOF calculates the trajectory of the projectile using the initial velocity from the internal ballistics and a typical drag profile of the projectile, obtained from high-fidelity CFD analysis. Only two degrees of freedom have been considered in-flight to simplify the in-flight dynamics. Target penetration, as the title suggests, calculates the depth of penetration of the projectile into the target armor and checks whether the target has been defeated or not using the impact velocity of the projectile from the two-DOF module, target thickness and target obliquity. The methodology for such a physics module has been derived

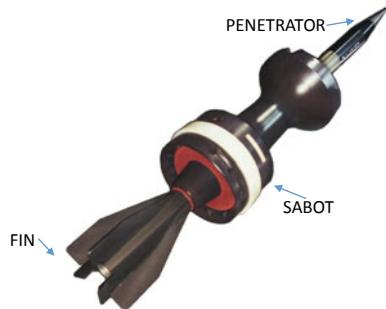
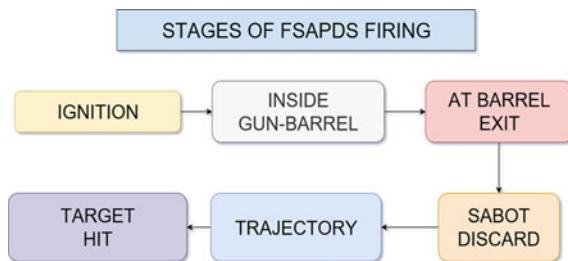


Fig. 3 Fin stabilized armor piercing sabot discarding KE projectile (*Source https://battlefield.fandom.com/wiki/APFSDS-T_Shell*)

Fig. 4 Block diagram of the stages of operation of an FSAPDS



from the literature [3]. All these multiple disciplines were developed in-house using Python and C++. Apart from the physics modules present for analysis of any given FSAPDS design, there is a geometrical module that takes the design variables and calculates the dependent geometrical and inertial parameters like area, volume, mass, moment of inertia for the relevant sections and components. It also generates a CAD file of the design using FreeCAD (Figs. 3 and 4).

3 System Structure

The major headings of the optimization framework are the optimizer, the design assembly and the MDA. MDA is the physics module accessing various sub-modules for analysis. Design assembly takes the design variables from the optimizer and calculates dependent geometrical parameters, volume, mass, center of mass and moment of inertia of each component. There is a separate module for CAD generation of the design created using FreeCAD macros. Once the design is created, the MDA module is called which further calls internal ballistics, structural analysis, two-DOF and finally the target penetration sub-modules. Extended design structure matrix

Table 2 Description of keys to the XDSM framework

Symbol	Description
x^0	Vector of initial guess of design variables
x_0	Vector of design variables for an iteration
Des	Vector of design variables and dependent geometrical parameters for a iteration
Mate	Vector of material properties of components
Prop	Vector of properties of the propellant
Gun	Vector of geometry and properties of the combustion chamber and the gun barrel
Aero	Vector of aerodynamic loads
Area	Vector of cross-sectional area of bore and penetrator
Mass	Mass of each component, projectile and assembly
Target	Thickness and obliquity of the target
P_a, P_s, P_p	Ambient pressure, shot start pressure and peak pressure
St _{pk} , St _{ex} , St _{im}	States (position, linear and angular velocities, linear and angular accelerations and orientations) at peak, muzzle exit and impact
FOS, Vel _{res} , Range	Factors of safety of each component, residual velocity and range of flight
**	The optimal value of the physical quantity

describes the process and data flow in the framework, shown in Fig. 2. The key to the symbols used in the XDSM [4, 5] is provided in Table. 2. The framework was developed using the concepts and methodologies described in the literature [6–9] (Fig. 5).

4 Optimization Problem

There are four performance parameters obtained from the MDA. The difference of the impact velocity and ballistic limit of the target (beyond this value the target fails) called the residual velocity (V) is the first performance parameter followed by the range of flight of the projectile (R), factors of safety of either compressive or tensile failure in the penetrator (FOS_p) and sabot (FOS_s).

The present MDAO model contains a constrained optimization problem. The objective function can be defined based on two different applications of the model—it can be used to obtain better performance for a fixed range of target (say 1 km) and to obtain maximum possible range of targets the projectile can defeat. So, there are two cases of optimization possible.

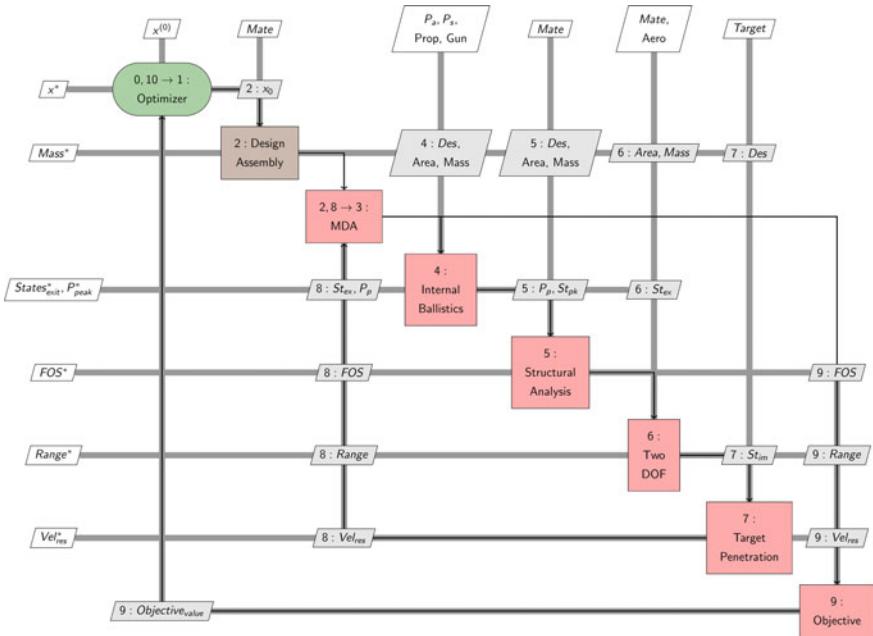


Fig. 5 Extended design structure matrix of FSAPDS MDAO framework

Case—1, titled ‘Maximizing performance for a fixed target range’, is defined as follows

Maximize	V	(1)
Subject to the constraints	$999.9 \text{ m} \leq R \leq 1000.1 \text{ m}$	(2)
	$FOS_p > 1$	(3)
	$FOS_s > 1$	(4)
	$S_2 > S_4$	(5)
	$\frac{l_{\text{sabot}}}{l_{\text{penetrator}}} > 0.5$	(6)
	$D_f < D_p$	(7)

Case—2, titled ‘Maximizing range of flight with target defeat’, is defined as follows

Maximize	R	(8)
Subject to the constraints	$V > 0$	(9)
	$FOS_p > 1$	(10)
	$FOS_s > 1$	(11)
	$S_2 > S_4$	(12)

(continued)

(continued)

$\frac{l_{\text{sabot}}}{l_{\text{penetrator}}} > 0.5$	(13)
$D_f < D_p$	(14)

where l_{sabot} is the total length of the sabot and $l_{\text{penetrator}}$ is the total length of the penetrator. Rest all other variables are either design variables or performance parameters. The bounds on the design variables were decided based on an initial design space search and on high-fidelity structural and CFD analysis that were run on a set of FSAPDS designs.

5 The Optimizer and OpenMDAO Setup

The framework for MDAO has been self-developed along the lines of object oriented programming in Python. OpenMDAO [10] only provides the necessary driver for the optimization purpose. It has multiple optimizers from multiple sources—‘SimpleGADriver’, ‘ScipyOptimizeDriver’ from Python module `scipy` and ‘pyOptSparseDriver’ from `mdolab`. From the open source options at OpenMDAO, only three optimizers support constrained optimization—COBYLA, SLSQP and SimpleGA. In this paper, optimization was performed using COBYLA and SimpleGA.

The physics section of the framework is a separate class in the OOP setup of the MDAO which inherits members and attributes from other classes design assembly, internal ballistics, structural analysis, two-DOF motion and target penetration. It contains a function that acts as the physics module to optimize for the OpenMDAO.

For OpenMDAO to work, a separate class is defined which inherits the ‘Explicit-Component’ class of ‘openmdao.api’. This is the section of MDAO framework that communicates with the actual physics module mentioned earlier, and the I/O of the MDAO framework is defined here using the `add_input()` and `add_ouput()` functions. The inputs include all the design variables in the optimization problem, while the outputs contain the variables defining the objective function and the constraints.

Figure 6 describes the way `openmdao` was configured to set up the problem statement and solve it. Basically, two interconnected objects are created—one for optimization problem (`prob`) and the other one for the physics model (`mod`). Every design variable and the physics class (`ExplicitComponent`) defined within the OpenMDAO setup was added to the physics model object (`mod`) as a separate subsystem using the `add_subsystem()` option. Connection between the design variables in the physics model object (`mod`) and the object of the physics subsystem is set up using the `connect()` function. Bounds on the design variables were provided using the `add_design_var()` function. The function `add_objective()` was used to define the objective function, while the constraints were defined using the `add_constraint()` function. An object (`prob.driver`) was created for the optimization driver being called

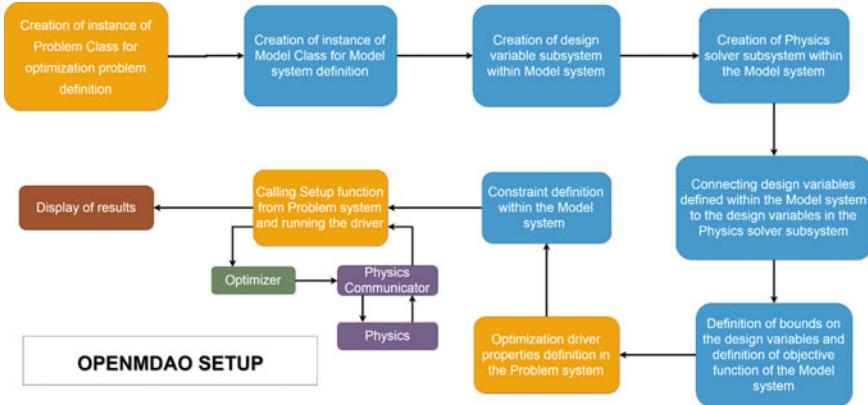


Fig. 6 Block diagram of OpenMDAO setup

by the program. It was further used to define the optimizer options. The final problem setup and the run were performed using the optimization problem object (prob).

For SimpleGA, the mutation rate was taken as 0.015, the crossover rate was 0.5, penalty parameter was 10, and the penalty exponent was kept 1. The bit values for each of the design variables ranged from 14 to 22.

6 Results

Both the cases of optimization experiments were performed with the COBYLA optimizer. The complete execution of COBYLA optimizer took about 30 min to 1 h, each iteration of the physics module consuming 2.56–6 s on an 8-core, 16 GB ram windows system running in serial mode. An initial guess to the design variables was provided, and the performance obtained for that is termed as ‘Input Performance’. The results from this set of optimizations are shown in Table 1. Fig. 7 shows the screenshot of the CAD output from Case—1 of the COBYLA optimizer.

Case—2 type of optimization was performed with SimpleGA, and the average fitness of the population was monitored after every generation. The population size selected was 140, while number of generations was kept 500. The constraints

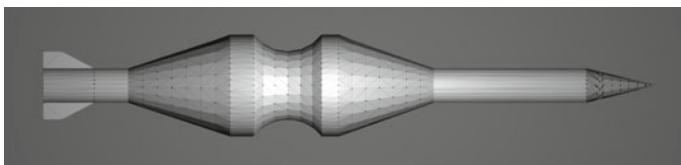


Fig. 7 CAD model pertaining to the results from Case—1 of COBYLA

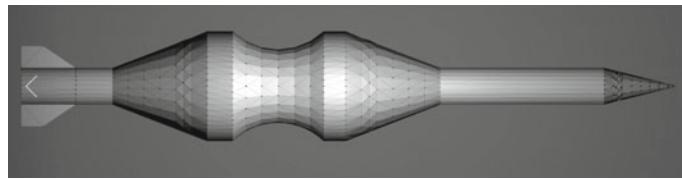


Fig. 8 CAD model pertaining to the results from SimpleGA for the first set of initial guess of design variables

mentioned earlier were provided to the optimizer as penalty functions. It was found that the average fitness kept increasing slowly and it became stagnant in a small region of fitness value after about 30th generation. So, keeping the number of generations 30 and population size the same, Case—1 optimization problem was performed with two different sets of initial guesses to design variables as input. The performance of the input set selected and the performance of the design variable set at the end of 30 generations are shown in Table 2. Fig. 8 shows the screenshot of the CAD output from the first results of the SimpleGA as shown in Table 2.

7 Discussion

From Table 3, none of the performance constraints in Case—1 and Case—2 of COBYLA-based optimization have been violated even with a constraint as rigid as the one present on the range of flight in Case—1. From Table 4, even with a totally

Table 3 Results from the COBYLA optimizer

		Case—1	Case—2
	Input performance	Output performance	
FOS_p	0.78	1.31	1.50
FOS_s	0.60	1.23	1.78
V (m/s)	−25.71	106.24	40.69
R (m)	1590.49	999.92	1113.54

Table 4 Results from the SimpleGA optimizer

	Input performance—1	Output performance—1	Input performance—2	Output performance—2
FOS_p	0.78	1.60	0.85	1.6
FOS_s	0.60	1.83	0.72	1.85
V (m/s)	−25.71	38.89	84.03	44.73
R (m)	1590.49	1105.30	744.10	1108.90

different set of values of initial guess of design variables, SimpleGA ended with results that were in neighborhood of each other in the design space. This exercise was performed numerous times to confirm the conclusion. Results were also compared by varying the optimizer options, i.e., penalty parameter, penalty exponent, mutation rate and crossover rate. After this comparison, the current values to the optimizer options were obtained.

Figures 8 and 9 depict the way design optimization took place using genetic algorithm. Each frame represents a generation with population size of 140 represented by a single band of color. The color bar on the side represents the fitness function increasing upwards. As the generation proceeds, a greater number of designs in a generation shift toward higher fitness value. This can be seen by the number of deep blue colored bands in the first generation to the 30th generation. Generation progresses from left to right.

8 Conclusion

The above attempts prove that such an exercise is possible with a physics model that includes the complexity of four modules from multiple disciplines. The value of the design variables obtained at the end of each type of optimization run, resulted in a feasible FSAPDS design. On top of that, no commercial software was used for the either the development of physics modules or the framework or the optimizer.

With SimpleGA, even though the results were feasible and the design space was narrowed down by a huge extent, there are methods that help in shrinking the design space further. The current design space was narrow enough for a practical application.

Keeping the simplicity intact, if one tries to improve the accuracy of the physics modules by using surrogate models over data accumulated from high-fidelity softwares for CFD and structural analysis, it might be possible to perform a similar experiment for a highly detailed design optimization which is what we will be working on. The only need would be to keep the framework same and just ‘plug-in’ the surrogate models in place of the current models. Introduction of balloting during exit, six-DOF, dispersion analysis, detailed modeling of buttress threading to capture realistic force transfer, effect of sabot discard on the overall design, manufacturability, etc., will be attempted by us in the future.

A lot of memory, data and process handling is required. To account for unpredictability, probability distributions of the possible errors can be introduced, and the design performances can be rated based on the probability of their performance. Hence, a stochastic model seems fit than a deterministic one if an accurate system is going to be attempted.

Also, visualization of the designs during optimization played a key role in ascertaining the methodology (Fig. 10).

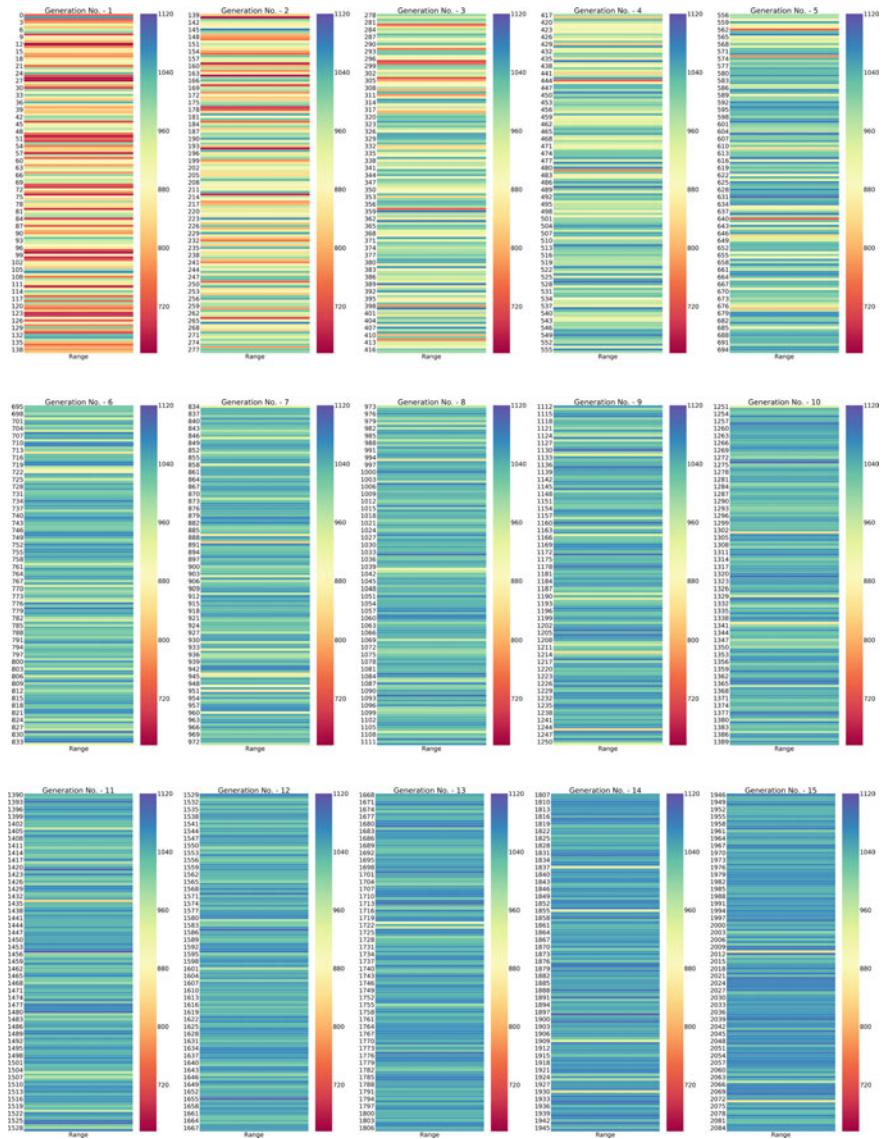


Fig. 9 Generation wise evolution of design using simple genetic algorithm from generation—1–15

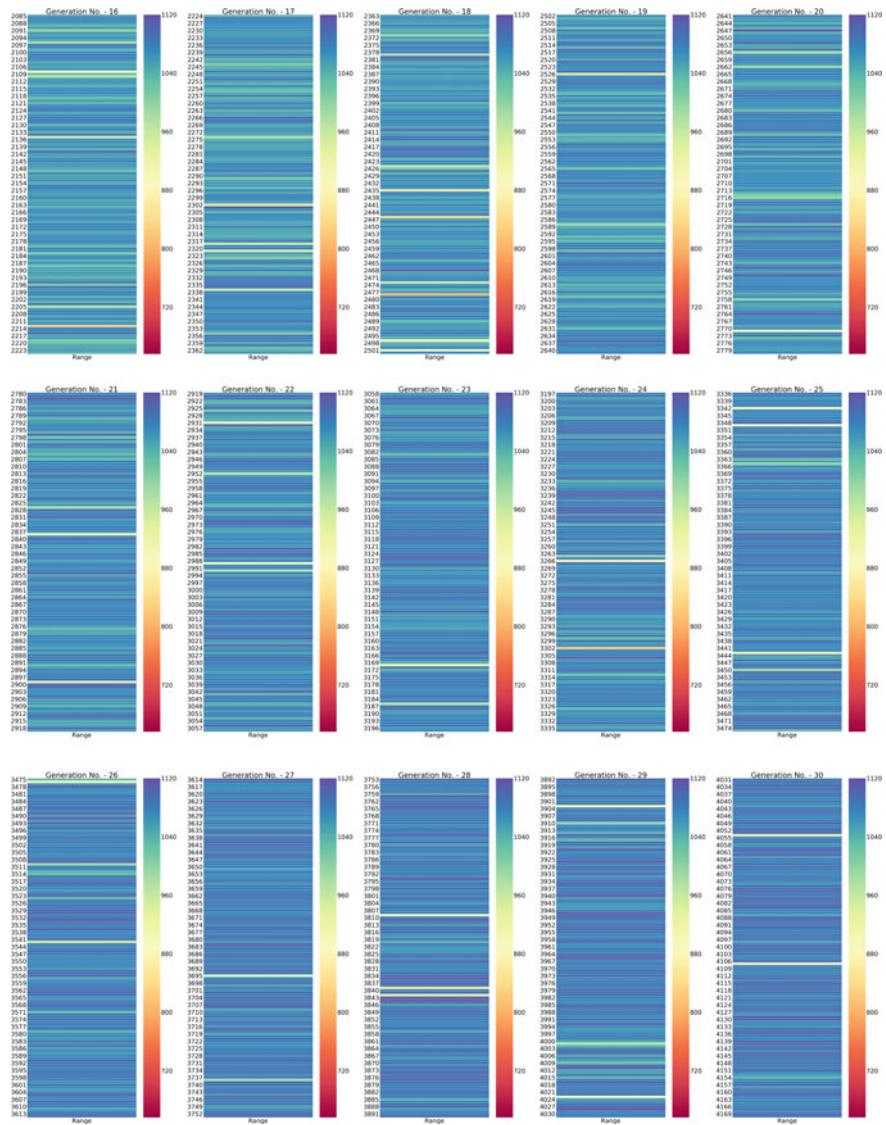


Fig. 10 Generation wise evolution of design using simple genetic algorithm from generation—16–30

References

1. G. Dyckmans, *Fundamentals of Ballistics*, Department of Weapons System Royal Military Academy of Belgium (2015)
2. D.E. Carlucci, S.S. Jacobson, *Ballistics Theory and Design of Guns and Ammunition*. CRC Press (2008). <https://doi.org/10.1201/b22201>

3. W.H. Drysdale, *Design of Kinetic Energy Projectiles for Structural Integrity*, Technical Report ARBRL-TR-02365 (1981)
4. A.B. Lambe, J.R.R.A. Martins, Extensions to the design structure matrix for the description of multidisciplinary design, analysis, and optimization processes. *Struct. Multi. Optim.* **46**(2), 273–284 (2012). <https://doi.org/10.1007/s00158-012-0763-y>
5. R.R.A. Joaquim, Martins: a short course on multidisciplinary design optimization. *Aerosp. Eng. Univ. Michigan* (2012). <https://doi.org/10.1016/j.cma.2012.04.007>
6. J. Sobiesczanski-Sobieski, A. Morris, M.J.L. van Tooren, G. La Rocca, W. Yao, *Multidisciplinary Design Optimization Supported by Knowledge Based Engineering* (2015). <https://doi.org/10.1002/9781118897072>
7. C.T. Leondes, *Multidisciplinary Engineering Systems: Design and Optimization Techniques and their Application* (1993)
8. P. Breitkopf, R.F. Coelho, *Multidisciplinary Design Optimization in computational Mechanics* (2013)
9. H.W. Engl, P. Kügler (auth.), V. Capasso, J. Périaux, *Multidisciplinary Methods for Analysis Optimization and Control of Complex Systems*, vol. 6. Mathematics in Industry (2005)
10. J.S. Gray, J.T. Hwang, J.R.R.A. Martins, K.T. Moore, B.A. Naylor, OpenMDAO: an open-source framework for multidisciplinary design, analysis, and optimization. *Struct. Multi. Optim.* (2019). <https://doi.org/10.1007/s00158-019-02211-z>

Reducing RCS of Aerospace and Marine Vehicles by Shaping



Sumit Jana, Mohd. Naimuddin, and Jason Jose

Abstract Radar systems have come a long way from World War II era, when detection and ranging was the only goal. Over the decades, radar technology has advanced rapidly; now it focuses on the target imaging, parameter estimation and radar target recognition. However, notwithstanding these developments, reducing radar cross section (RCS) through shaping remains an important and valuable design tool because most tools Yeh et al. (Sensors 16:125, 2016 [1]) assume that (a) RCS is more than some critical value and (b) the target is a metallic. It can be observed that reducing RCS through shaping as an important analysis/design activity to minimize the radar visibility and increase survivability of aircraft and missiles. The objective of this work is to utilize suitable numerical approach to reduce RCS of some canonical shapes, understand the basis for reduction in RCS, and then apply this basis for reducing RCS of real-life shapes such as aircraft, missiles and ships.

Keywords Radar · RCS · CEMExpert · PO-PTD-SBR · Ship · Optimization · Azimuth

1 Introduction

The RCS of a complex body such as an aircraft is due to echoes of scattered electromagnetic (EM) waves from all its surfaces. The scattered EM waves get generated from surface currents, if they are electrically conducting, when immersed in incident EM field. The areas which produce significant surface currents due their exposure to EM waves are called hot spots. Whereas the regions in space, as sensed

S. Jana (✉) · Mohd. Naimuddin · J. Jose (✉)
Zeus Numerix Pvt. Ltd, Pune, India
e-mail: sumit@zeusnumerix.com

J. Jose
e-mail: jason.jose@zeusnumerix.com

Mohd. Naimuddin
e-mail: mohd.naimuddin@zeusnumerix.com

by the receiving antenna, contributing the RCS significantly are called scattering centers. Finding out scattering centers requires analysis of radar echoes, in time, at the receiving antenna; it requires complex digital signal processing. Both hot spots and scattering centers are function of aircraft geometry, frequency and polarization of illuminating EM wave and aspect angle. The focus of current work is to locate hot spots and reduce the magnitude of surface currents which are a prerequisite to inhibit target recognition.

Broadly speaking, the EM simulation methods can be classified as (a) full-wave simulation methods which require solution of Maxwell's curl equations six in numbers and (b) ray-based simulation methods, which models EM field through rays undergoing propagation along a straight line in space and reflecting and scattering when obstructed by solids. Ray-based simulation methods are computationally efficient; i.e., they are in order of magnitude faster with an acceptable loss of accuracy. Their computational efficiency increases as frequency of EM field increases. They are a natural choice for analysis stealth. PO-PTD-SBR, a ray-based tool in CEMExpert is used in this study. MPI/Open MP-enabled CEMExpert produces hot spots for a given complex realistic geometry, such as an aircraft, missile or even a ship, modeled using millions of triangles at a given aspect angle in minutes on a parallel computer.

PO-PTD-SBR is known to consider primary (specular reflection) and secondary scattering phenomena. However, since the aim of the paper is to optimize the shape for stealth, rather than find out very accurate RCS value for a given frequency and aspect angles, we need to only concentrate on the principle scattering mechanism and PTD option can be disabled. The argument can be easily defended because current radar technology uses high frequencies, and for these frequencies, the size of the aircraft is an order of magnitude larger, and PO-PTD_SBR becomes the method of choice. More importantly, scattering becomes more localized as frequency increases which helps in attacking problem by altering the geometry locally.

2 Methodology

Surface currents are formed on the surface of the vehicle which is exposed to the incident EM field. Areas of large surface currents are called hot spots. The surface currents in turn produce scattered field considering interference of EM field as the various contributions that go in and out of phase with each other. The interference could be due to the surfaces far away from each other or due to one surface. Naturally, the methodology to reduce RCS is: (a) locate hot spots, (b) order the hot spots as per their relative contribution to the total RCS, (c) reduce contribution of the identified hot spot one by one by changing the geometry locally so that induced currents on them goes down. The above methodology requires answers to three questions: (a) what mechanism can be used to reduce surface currents, (b) how much reduction is expected and (c) implementation issues while working with realistic geometry. The first two questions are dealt with in the next section.

Any shape can be considered as an assembly of several flat surfaces. In some real-life examples, such as a ship, the geometry can be approximated by a closed non-overlapping collection of large flat surfaces, small in numbers. On the other hand, aerospace geometries, such as a missile, can be approximated by small flat triangles (say) large in numbers. We postulate that aircraft geometry can be reasonably modeled using moderately large-sized flat surface sufficiently small in numbers (but still running in thousands). The last question is answered for a ship in the same section. A ship is used as an example as geometric modeling of ship poses lesser constraints compared to other shape such as aircraft.

Experiments with standard shapes and a real-life vehicle: As stated above, one can consider a flat triangular or a trapezoidal shaped to see how hot spots are formed and how their intensity can be reduced. As stated above in Sect. 2 (theory and scope of work), scattered EM field or surface currents are affected by aspect angles for a given frequency.

We therefore find out surface currents scattered field and RCS for the range of azimuth and elevation angles (0° – 90°). This study using CEM_Expert is reported here. A number of flat surfaces with regular geometric shapes, (i) square plate of size $1\text{ m} \times 1\text{ m}$, (ii) rectangular plate of $0.5\text{ m} \times 1\text{ m}$, (iii) triangular plate of 1 m base $\times 1\text{ m}$ height, (iv) trapezoidal plate of widths 0.5 m and 1 m) and height 1 m , were tested. All the surfaces are considered as metallic PEC and flat in shape.

RCS of a square plate is shown in Fig. 1 for trapezoidal plate for an azimuthal, and elevation sweeps at 1 Ghz is shown. It can be seen that RCS changes rapidly

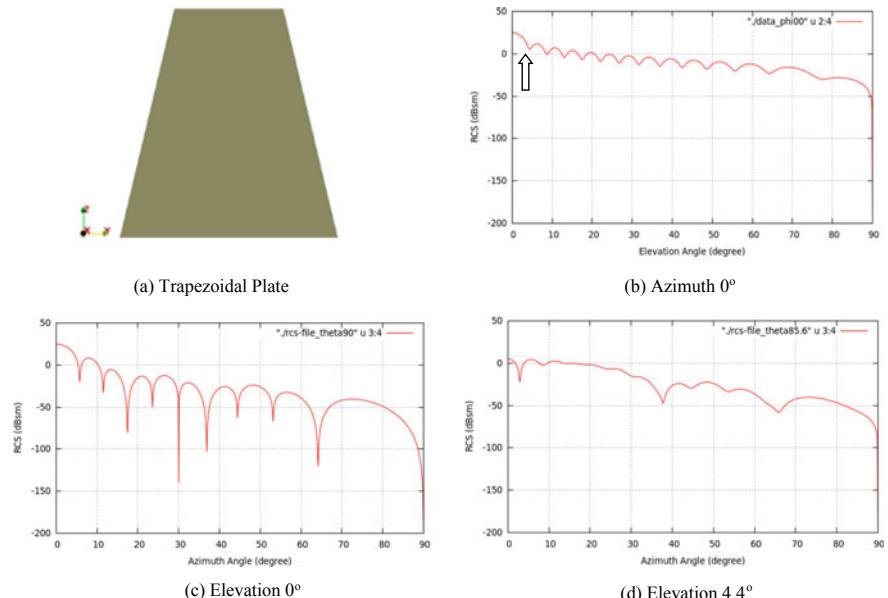


Fig. 1 RCS pattern of a trapezoidal plate elevation and azimuth sweep 0° – 90°

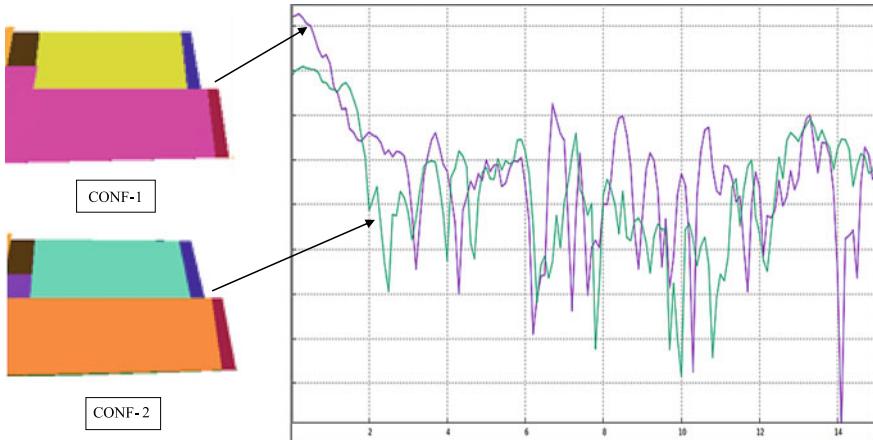


Fig. 2 RCS comparison of a ship with and without surface inclination correction (for head on incidence 0° to 15°)

but in a complex manner as plate changes angle with respect to the incident field. Figure 1b shows the RCS plot in elevation plane at azimuth is 0° . We find the first dip in RCS value at 4.4° as indicated by the arrow. Figure 1c, d shows the RCS of azimuth sweep for elevation 0° and 4.4° . A flat surface radiates maximum energy when all surface points are in the same phase of the incident ray. It could be easily concluded that only two basic variables matter: the projected size of the plate and the frequency of illumination. However, irrespective of their magnitudes, what matters is the number of bands of surface currents. Smaller the number of surface current bands, small is the scattered field and smaller is RCS value.

The above philosophy was used for a ship. A total of 15 flat surfaces were identified and ordered based on their relative contribution to RCS. Each surface was shaped (made inclined in vertical direction) at various angles one after the other to reduce the contribution of each surface. The median value for RCS could be brought down several orders.

Figure 2 shows the difference in RCS values for with and without surface inclination optimization. CONF-1 is the initial design, and CONF-2 is the corrected one. A couple of degrees of correction in the inclination results into a considerable improvement in overall RCS character.

3 Conclusion

Canonical shapes (quadrilaterals) were studied using PO/PTD for understanding the reinforcing/interference of scattered EM as produced by surface currents. Slanting of faces with respect to vertical direction was used to ensure that interference produced the minimum possible RCS. Optimization methodology established using flat-plate

scatterers was used to improve the stealth characteristic of a large vehicle for various frequencies between 2 GHZ to 30 GHz and azimuthal angle of $\pm 10^\circ$ nose-on. Shaping exercise could bring down RCS by 10 dBSm.

References

1. C.-M. Yeh, W. Zhou, L. Yao-Bing, J. Yang, Non-cooperative target imaging and parameter estimation with narrowband radar echoes. Sensors **16**, 125 (2016). <https://doi.org/10.3390/s16010125>
2. E.F. Knott, Radar cross section measurements. Book (1993). <https://doi.org/10.1007/978-1-4684-9904-9>
3. EMC Studio/EMCoS Antenna VLab Application node #19: Analysis of RCS problems in ships. Application Note
4. J. Khan, W. Duan, S. Sherbaz, Radar cross section prediction and reduction for naval ships. J. Marine Sci. Appl. **11**(2), 191–199 (2012). <https://doi.org/10.1007/s11804-012-1122-5>
5. Stoyanov, Y.J.: American society for engineering education, AD-A186 107, ISBN: B00CQCVDII

Performance Analysis and Design Optimization of Two-Baffle Muzzle Brake of 155 mm Artillery Gun



Anubhav Tiwari and Vighnesh Pawar

Abstract Muzzle brake is a device fitted at the muzzle end of the gun barrel to reduce the recoil energy by venting some amount of propellant gases sideways as the shot is fired. The focus of the study is a two-baffle muzzle brake which has its vast application in most of the 155 mm bored barrel of advanced artillery guns of the current generation. The objective of the study is to analyse and then optimize the current design of two-baffle muzzle brake developed for 155 mm barrel of advanced towed artillery gun system. The performance characteristics of muzzle brake is measured in terms of its efficiency, sound pressure levels (SPL) generated in the crew zone, mechanical strength and durability, considering the limitation on its weight which imposes instability at the muzzle end and further effects the accuracy of the impact of shot on the target. The indigenous computational methods and tools developed and validated in-house are used for the analysis and design optimization cycles. The inviscid, parallel flow solver developed was used to solve time-dependent fluid flow simulations of propellant gases. The solver is capable of executing dynamic simulations with rigid bodies in relative motions that solve the motion of shot from all burn point (ABP) of solid propellant to the instant of achievement of ambient conditions inside the combustion chamber of barrel. The peak pressure and the muzzle velocity of the shot attained for intense firing condition is 450 MPa and 890 m/s, respectively. The achievement of the study is a three-baffle muzzle brake design with efficiency 20% greater than current two-baffle muzzle brake, and other performance characteristics remained unaltered.

Keywords Muzzle brake · Brake efficiency · Stress distribution · Sound pressure level (SPL)

A. Tiwari · V. Pawar
Zeus Numerix Pvt. Ltd, Pune, India
e-mail: anubhav.tiwari@zeusnumerix.com

1 Introduction

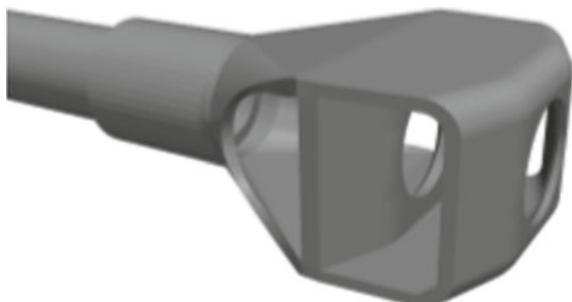
Muzzle brake is a device fitted at the muzzle end of the gun barrel to reduce the recoil impulse and energy. Muzzle brake reduces the recoil energy by deflecting the propellant gases after exit from the muzzle end in the backward direction by its baffles, as shown in Fig. 1. The configuration of muzzle brake like the number of baffle stages, angle of deflection of the propellant gases and width of baffles governs the efficiency of muzzle brake. The increase in number of baffle stages or higher deflection angles boosts the efficiency of muzzle brake but also makes the configuration bulky which is undesirable as it effects the stability of barrel during firing. Hence to optimize the mass and efficiency of the muzzle brake, there is a need to carry out the performance analysis and design optimization of the current muzzle brake configuration.

The current muzzle brake configuration has two baffle stages with mass of 180 kg fitted at the muzzle end of 155 mm × 52 cal barrel of artillery gun, as shown in Fig. 2.

Fig. 1 Propellant gases are vented sideways by the muzzle brake as the shot exits from the muzzle end
(Source <https://warriorlodge.com/pages/m198-howitzer>)



Fig. 2 Double-baffle muzzle brake configuration



The indigenous computational methods and tools developed and validated in-house are used for the performance analysis and design optimization study [1–5]. The inviscid, parallel flow solver developed was used to solve time-dependent fluid flow simulations of propellant gases inside the barrel chamber and the interaction of escaping propellant gases with the muzzle brake, as the shot is fired. The effectiveness and performance of the muzzle brake is measured in terms of brake efficiency and sound pressure level (SPL). The first expression is the overall brake efficiency, which is the per cent reduction in the recoil impulse on the recoil assembly due to the action of the muzzle brake. The recoil assembly consists of barrel, muzzle brake and breech mechanism that slides on the rails provided on the undercarriage of the gun. The brake efficiency, represented by φ , is defined as follows:

$$\varphi = \frac{I_{R_{WO}} - I_{R_W}}{I_{R_{WO}}} \cdot 100\% \quad (1)$$

$I_{R_{WO}}$ symbolizes the total recoil impulse without the brake and I_{R_W} is the total recoil impulse with the brake. The total recoil impulse exerted by propellant gases on the recoil assembly is given by

$$I_R = \int_0^{t_R} F_R \, dt \quad (2)$$

where F_R is the net pressure force exerted by the propellant gases on the barrel and muzzle brake assembly and t_R is the total recoil time. The free recoil velocity is expressed as:

$$V_R = \frac{I_R}{M_W} \quad (3)$$

where M_W is the mass of the recoil assembly.

2 Simulation Methodology

The analysis includes internal ballistics and CFD simulations using in-house developed computational tools and modules [6, 7]. The objective of the internal ballistics simulation is to obtain the time-varying flow conditions of propellant gases from initiation to ignition to all burn point (ABP) inside the barrel chamber. On the other hand, the objective of CFD simulations is to simulate the coupled shell motion and fluid dynamics of escaping propellant gases from the muzzle end of the gun barrel, calculate the pressure forces exerted on the muzzle brake and measure peak overpressure generated by the muzzle blast in the vicinity of breech end of barrel. The coordinate system adopted for the fluid dynamics simulation is shown in Fig. 3. The

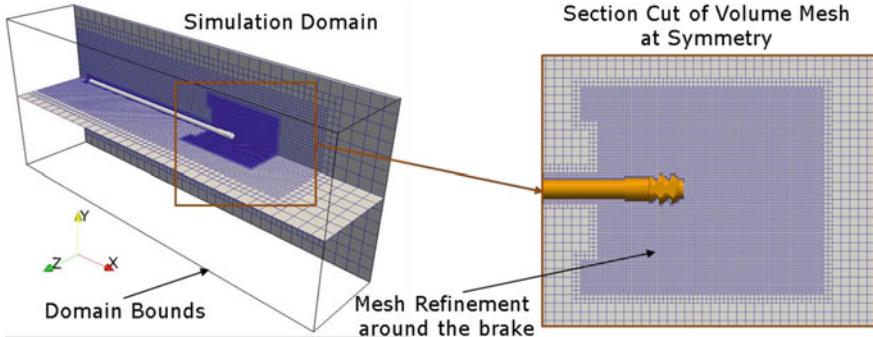


Fig. 3 Unstructured volume mesh

Euler formulations are adopted for the simulations; thus, fluid viscosity of propellant gases is not taken into account [8]. Shell is assumed to have only one degree of freedom (1 DOF) with its motion in positive x-direction. Leakage of gases through the gap between shell and barrel inner wall is not allowed in the solver. Friction factor that takes into account the friction between shell outer surface, and barrel internal grooves are also implemented in the dynamic module. Mass of the recoil assembly and shell is to be 3040 kgs and 45 kgs, respectively. Except the volume between breech end and rear end of the shell, rest of the computational domain is initialized at ambient condition.

3 Computational Domain

Since, the geometry of the recoil assembly is collectively having planar symmetry about the z-axis, only half of the body is considered for simulation, x-axis being the barrel central axis. Sectional view of the volume mesh generated around half of the body is shown in Fig. 3. Figure 3 shows the section view seen from the z-direction at the plane of symmetry of the barrel, and the section seen from x-direction shows only half of the body present inside the computational domain.

The list of geometrical inputs used for modelling is given in Table 1.

Table 1 List of geometrical inputs

Diameter of barrel	155 mm
Length of barrel	8053 mm
Mass of muzzle brake	180 kg
Mass of recoil assembly	3040 kg

4 Simulation Analysis and Discussion

An accurate prediction of the overall brake efficiency and dynamic efficiency of the muzzle brake is an aim of this study. Figure 4 shows the pressure variation inside the barrel at the breech end, and Fig. 5 shows the velocity of the shell throughout its travel. It then proceeds towards high rate of combustion of propellant and formation, expansion of gases located at the rear of the shell, with shell being moving towards the muzzle end.

The shell exits from the muzzle end of the barrel at approximate 25 m/s, as shown in Fig. 5. Figure 6a shows the velocity contours of fluid around the muzzle brake at an instant shell has left the brake. The gases are diverted sideways and rearwards by the muzzle brake at a certain angle with reference to barrel axis, as shown in Fig. 6b.

The performance characteristics of the double-baffle muzzle brake configuration are given in Table 2.

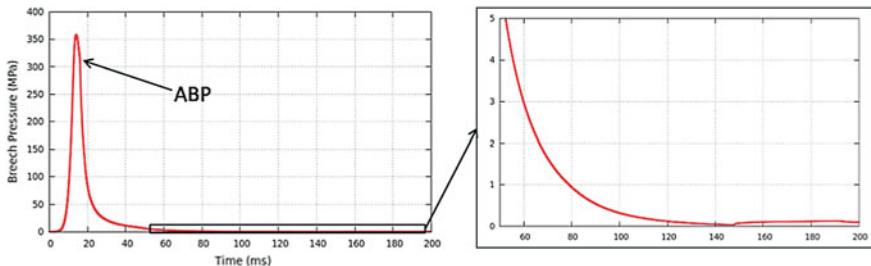


Fig. 4 Variation of barrel pressure at breech end with respect to time

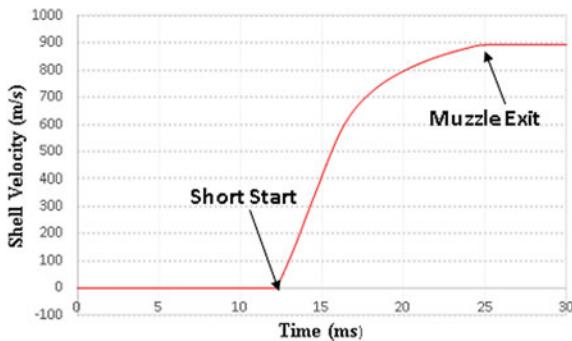


Fig. 5 Shell velocity with respect to time. Shell muzzle velocity = 890 m/s

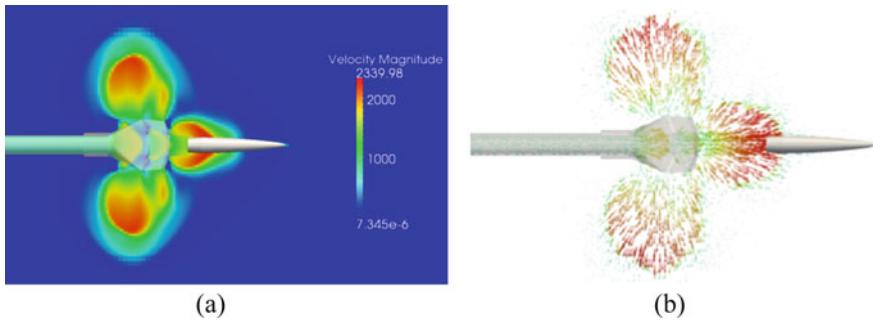


Fig. 6 **a** Velocity contours and **b** velocity vectors at the symmetry plane during shell exit from the muzzle end

Table 2 Performance parameters of double-baffle muzzle brake configuration

Shell muzzle velocity	892.9 m/s
Overall brake efficiency	19.7%
Kinetic energy based efficiency	39.4%

5 Sound Pressure Level (SPL) Measurement

Noise exposure from a typical sound source is measured in terms of sound pressure level or acoustic pressure level. SPL is a logarithmic measure of the effective pressure of sound relative to a reference value

$$L_P = \ln \frac{P}{P_0} = 2 \log_{10} \frac{P}{P_0} = 20 \log_{10} \frac{P}{P_0} \quad (4)$$

where p is the effective sound pressure, p_0 is the reference sound pressure (for air: 20 μ Pa). Two probes were positioned 4 and 10 m from the muzzle end. First probe was placed at an angle of 90° with the barrel axis and second probe at an angle of 180°. The time-varying measurement of static pressure at Probe 1 and Probe 2 is shown in Fig. 7.

The peak SPL measured at Probe 1 and Probe 2 for double-baffle muzzle brake configuration is given in Table 3.

6 Design Optimization

The efforts involved in design optimization study aims in achieving design that is more efficient in performance and lighter than the current double-baffle muzzle brake configuration. The study also considers the constraints imposed due to the sound pressure levels generated in the operating zones of the gun and the strength of the muzzle

Fig. 7 Time variation of static pressure measured at Probe 1 and Probe 2

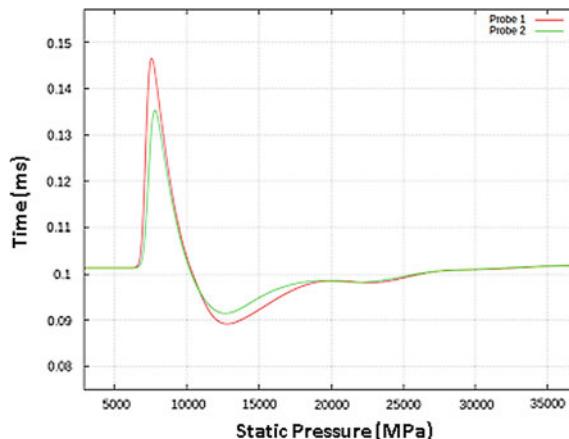


Table 3 SPL measurement

Probe number	Linear position (m)	Angular position (°)	Double baffle SPL (dB)
1	4	90	186
2	10	180	165

design. The optimization methodology used for this study was design trial and error. Depending upon the design parameters and the design constraints of the muzzle brake initial inputs were specified to some of its design parameters and its performance characteristics were evaluated until the desired output is achieved. Design parameters includes number of baffles, inlet angle, first baffle distance, spacing between first and second baffle, spacing between second and third baffle and the outlet angle. Once the desired output is achieved, the final revision is concluded as the optimized design. This procedure involved 15 design revisions to achieve the optimized design.

The flow work of design optimization procedure is shown in Fig. 8a. The cut section of last three design revisions is shown in Fig. 8b. The optimized design (Case B15) is a triple-baffle muzzle brake configuration which is 5 kg lighter and 5% more efficient than the double-baffle muzzle brake configuration. The sound pressure levels recorded for triple-baffle muzzle brake configuration in operating zone are equal in magnitude as double-baffle muzzle brake.

7 Structural Analysis

Using the pressure forces exerted by the propellant gases on the muzzle brake, as loading conditions, the transient structural analysis of the muzzle brakes was carried out in order to estimate the structural strength of double-baffle and triple-baffle muzzle brake configuration.

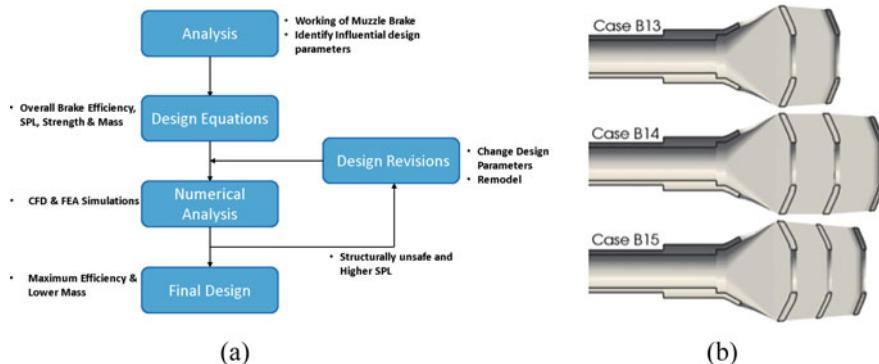


Fig. 8 **a** Design optimization process and **b** cross-sectional view of optimized design of double baffle and triple baffle configuration

The region of maximum stress is at the junction between the first baffle and the upper flat plate. The stresses for 558 MPa pressure is around 1972 MPa which are at surface level as shown in Fig. 9. FEM analysis of the optimized design is carried out for weight reduction and allowable stress level (Fig. 10).

The linearized stress on both the paths across thickness of baffle plate and maximum value of linearized stress are found to be 600 MPa which is less than allowable strength (940 MPa) of the muzzle brake. Hence, this design is structurally safe.

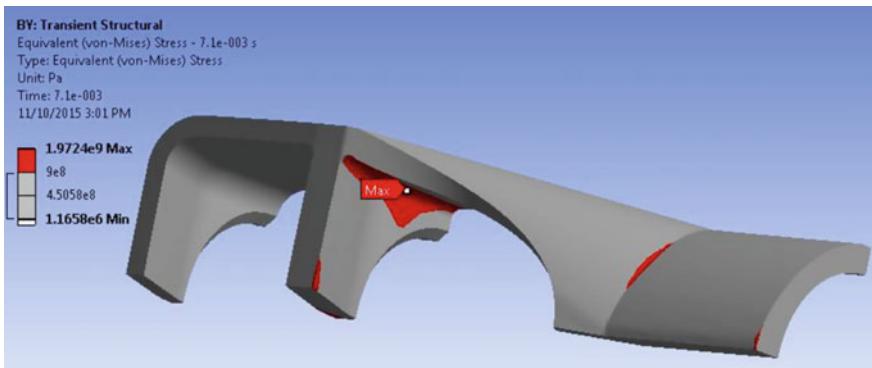


Fig. 9 Equivalent stress of double-baffle muzzle brake

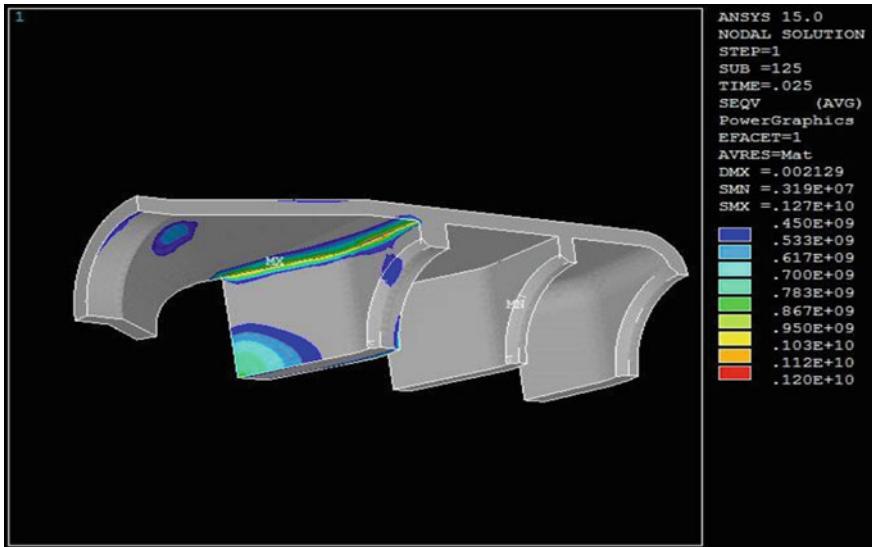


Fig. 10 FEM analysis result of optimized design (triple baffle)

8 Conclusion

The advanced computational tools and mathematical models were adopted for the performance analysis of current double-baffle muzzle brake configuration. The overall brake efficiency of double-baffle muzzle brake obtained from the analysis is 19.7%. The analysis further involved measurements of sound pressure levels generated in the operational zone of the gun which is found intolerable and harmful to exposed human hearing senses during firing. The time-dependent pressure forces on muzzle brake obtained from CFD simulations were used as loading conditions for structural analysis which resulted into prediction of safe design. The procedure of design optimization cycle of double-baffle muzzle brake configuration accumulated 15 design revisions by design trial and error and resulted into achievement of triple-baffle muzzle brake configuration (optimized design) which is nearly 5% more efficient and 5 kg lighter than the double-baffle muzzle brake. The similar levels of sound pressures and maximum equivalent stresses were observed in CFD analysis and structural analysis of both the muzzle brake configurations.

Acknowledgements The authors thank and acknowledge many discussions during the course of this research with Prof. Gopal Shevare, IIT Bombay, and ARDE, DRDO Laboratories, Pune, for sharing their rich experimental experiences.

References

1. V. Pawar, *Development of Customized Software for CFD analysis of Gas Operated Weapon*. Zeus Numerix Technical Report, January, 2011
2. S.M. Murman, M.J. Aftosmis, M.J. Berger, Implicit approached for moving boundaries in a 3D cartesian method. 41st AIAA Aerosp Sci Meet. (2003)
3. D.H.M.M.W.S. Lennart Schneiders, An accurate moving boundary formulation in cut cell methods. *J. Comput. Phys.* (2012)
4. M.J.B.J.E.M.M.J. Aftosmis, Robust and efficient cartesian mesh generation for component-based geometry. *AIAA Paper* (1997)
5. S.A. Bayyuk, K.G. Powell, B. Vanleer, Computations of flows with moving boundaries and fluid structure interactions, in *13th AIAA Computational Fluid Dynamics Conference* (1997)
6. G. Dyckmans, Fundamentals of ballistics—eBook, <http://e-ballistics.com> (2007–2012)
7. J. Blazek, *Computational Fluid Dynamics: Principles and Applications*, 1st edn. (2001)
8. C.-W. S. Sirui Tan, A high order moving boundary treatment for compressible inviscid flows. *J. Comput. Phys.* **230**, 6023–6036 (2011)

Neural Networks and Machine Learning in Optimization

Inverse Response Surface Method for Structural Reliability Analysis



M. Nagesh and A. S. Balu

Abstract Reliability-based design of complex structural systems is a computationally tedious task. In order to reduce the computational effort, approximation methods, such as classical response surface method, Kriging model and artificial neural network, can be adopted. Response surface model is a conventional method, where the limit state function is approximated using a suitable surrogate model. For the construction of response surface, variables of stochastic model should be known well in advance. However, the design parameters are unknown during initial stages of reliability-based design optimization (RBDO). For such structural design cases using RBDO, an adaptive inverse response surface procedure is proposed in this paper. The procedure is developed by coupling the adaptive response surface method with suitable experimental design (Halton low-discrepancy sequence sampling) for estimating reliability indicators and artificial neural network-based inverse reliability method for design optimization. The validity and accuracy of the proposed method are tested on example with explicit nonlinear limit state function.

Keywords Artificial neural network · Inverse reliability analysis · Reliability-based design optimization · Response surface methods

1 Introduction

Structural design is carried out in order to ensure the reliable working condition of any newly built structure throughout its lifetime. Reliability index or probability of failure for the structure is to be evaluated for a particular limit state. For complex systems, evaluation of these indicators is computationally expensive task. To reduce the computational effort, approximation methods or metamodels are employed as an

M. Nagesh (✉) · A. S. Balu

Department of Civil Engineering, National Institute of Technology Karnataka,
Surathkal, India

e-mail: nageshm807@gmail.com

A. S. Balu

e-mail: asbalu@nitk.ac.in

appropriate solution without compromise in the accuracy. Approximation methods replace the original limit state function (LSF) with a simpler function which are less time consuming. Classical simulation methods are used to estimate probability of failure with the approximated function. One of the popular methods of approximation is the response surface method (RSM) [1] where the polynomial function is used more often to replace LSF. Some other methods are Kriging metamodel [2], artificial neural network (ANN) [3], support vector machine [4].

Each variable in the stochastic model is to be known well in advance during construction of a response surface. During the initial stages of the structural design process, the design parameters are unknown subjected to RBD0 procedure. To evaluate these design parameters, adaptive inverse response surface method with Halton low-discrepancy sequence (LDS) sampling for small sample simulation is utilized in the proposed method of this paper.

2 Response Surface Method

The representation of model behavior of any system by a response surface can be considered as independent of the stochastic properties of the basic variables. Reliability analysis of the model can be improved substantially in both accuracy and efficiency, by taking into account the prior knowledge of the stochastic properties, i.e., mean values and standard deviations of the variables involved. Additional information need not be precisely available such as distribution type and correlation between the variables. The set of basic random variables including the load and system conditions represented by \mathbf{X} is considered to denote the limit state condition as given by Eq. (1).

$$g(\mathbf{X}) \leq 0 \quad (1)$$

This original LSF can be replaced by a second-order polynomial function as an approximation:

$$\tilde{g}(\mathbf{X}) = a + \sum_{i=1}^n b_i X_i + \sum_{i=1}^n \sum_{j=1}^n c_{ij} X_i X_j \quad (2)$$

where a, b_i, c_{ij} are the unknown regression coefficients of the approximation function, $i, j = 1, 2, \dots, n$. The regression coefficients are evaluated by performing number of numerical experiments by selecting a suitable experimental design for input variables. The number of experiments required during evaluation of Eq. (2) is $1 + n + n(n + 1)/2$. Bucher and Bourgund [1] further simplified the approach by using polynomial function without the mixed terms $X_i X_j$ in the form of Eq. (3).

$$\tilde{g}(\mathbf{X}) = a + \sum_{i=1}^n b_i X_i + \sum_{i=1}^n c_i X_i^2 \quad (3)$$

The number of numerical experiments reduces to $2n + 1$ without mixed terms. Depending on the problem, sometimes the response surface neglecting mixed terms does not yield accurate approximations. At such situation, mixed terms are included as stated in Eq. (2). In the next step, based on the assumption of uncorrelated Gaussian variables, the function $\tilde{g}(\mathbf{X})$ is used to obtain an estimate of the design point X_D for the surface $\tilde{g}(\mathbf{X}) = 0$. Once X_D is found, $\tilde{g}(X_D)$ is evaluated and a new center point X_M for interpolation is chosen on a straight line from the mean vector X to X_D , i.e.,:

$$X_M = \bar{X} + (X_D - \bar{X}) \frac{\tilde{g}(\bar{X})}{\tilde{g}(\bar{X}) - \tilde{g}(X_D)} \quad (4)$$

The same approximation using Eq. (3) is repeated using X_M as the new center point, and the total number of LSF evaluations becomes $4n + 3$. Updating the polynomial results leads to an accurate response surface.

3 Artificial Neural Network-Based Inverse Reliability Method

Assessment of design parameters for desired reliability is referred to as inverse reliability analysis (i.e., structural design). Parameters may be deterministic or random in nature, related to the structure, load acting on the structure or surrounding environment. The vector of basic random variables $\mathbf{X} = \{X_1, X_2, \dots, X_n\}$, the vector of design deterministic parameters $\mathbf{d} = \{d_1, d_2, \dots, d_n\}$, and the vector of the design parameters of random variables $\mathbf{r} = \{r_1, r_2, \dots, r_p\}$. Z_j represents safety margins and target failure probabilities $P_{f,j}$ or reliability indices β_j , where $j = 1, 2, \dots, m$ is number of LSFs. The inverse reliability method in the form of RBDO problem is generally stated as:

$$\begin{aligned} &\text{Given: } P_{f,j} \text{ or } \beta_j \\ &\text{Find: } \mathbf{d} \text{ and/or } \mathbf{r} \\ &\text{Subject to: } Z_j = g(\mathbf{X}, \mathbf{d}, \mathbf{r})_j = 0 \text{ for } j = 1, 2, \dots, m \end{aligned} \quad (5)$$

ANN is used as a surrogate model of unknown inverse function describing relation between the design parameters and corresponding reliability indicators [5]. ANN builds the relation as in Eq. (6) between $\mathbf{P} = (\mathbf{d} \cup \mathbf{r})$ the vector of all design parameters and $\mathbf{I} = \boldsymbol{\beta}$ (or $\mathbf{I} = \mathbf{P}_f$) the vector of reliability indicators.

$$\mathbf{P} = f_{\text{ANN}}^{-1}(\mathbf{I}) \quad (6)$$

During ANN-based analysis, the selection of appropriate structure of ANN is important, which includes number of hidden layers and number of neurons, respectively. Kurukova [6] suggested that two hidden layers with sufficient number of neurons are sufficient to solve inverse reliability problems. A suitable sampling technique, i.e., Halton LDS sampling [7], is used in this work for training set preparation. After ANN training, the design parameters can be evaluated corresponding to desired reliability.

4 Inverse Response Surface Method

Response surface is an approximation to the original LSF as discussed in Sect. 2. To construct the response surface in the forward approach, desired design variables are to be known well in advance. For this purpose, an inverse response surface method [8] is proposed. The method utilizes ANN and Halton LDS sampling technique for robust, efficient and feasible solution for solving time-consuming structural design problems. An iterative scheme shown in Fig. 1 to upgrade the response surface and, at the same time, to accomplish the inverse reliability analysis [9–11] is proposed as follows:

1. Initial values of the design parameters help to construct the initial response surface using polynomial model in the forward approach.
2. Based on the constructed response surface, the ANN-based inverse reliability analysis is carried out and a new estimate of design parameters is obtained as well as the design point.
3. New anchor point is calculated from the design point using Eq. (4). The new anchor point with the previously obtained design parameters is used for the response surface update. For the updated response surface, ANN-based inverse reliability analysis is carried out again to seek the new design parameters and the design point.
4. The process is to be repeated till the design parameter converges to an acceptable level.

5 Numerical Example

An explicit nonlinear LSF [12] has been selected from the literature to validate the proposed method as follows.

$$g(\mathbf{X}) = \exp[0.4(X_1 + 2) + 6.2] - \exp(d) - 200 \quad (7)$$

where X_1 is the standard normal variable and d is an unknown deterministic parameter.

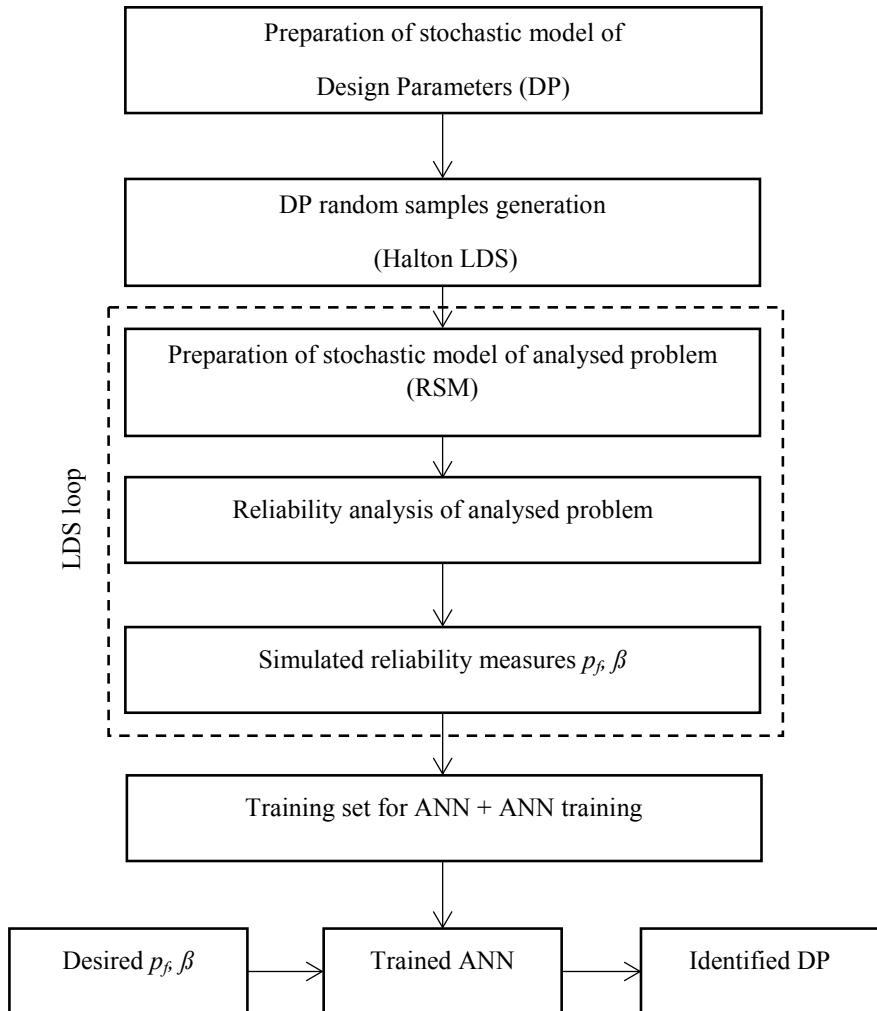


Fig. 1 Inverse response surface procedure

Monte Carlo simulation (MCS) with 10 million samples performed on Eq. (7) with target value of $d = 5.163$ yields target reliability index value $\beta = 2.688$. Randomization of d is considered as uniform distribution with parameters as given in Table 1, and these values are used for the construction of response surface and to perform ANN-based inverse reliability analysis.

Table 1 Randomization of design parameter

Variable	Mean	Standard deviation	Min	Max
d	6	1.155	4	8

Table 2 Results of iterative process

Parameter	Identification			Target value
	Iteration 1	Iteration 2	Iteration 3	
d	5.470	5.198	5.159	5.163
β	2.661	2.673	2.686	2.688

The proposed methodology is utilized to determine design parameter value in Eq. (7). Utilized ANN during inverse reliability analysis consisted of two nonlinear neurons in a hidden layer and a linear output neuron corresponding to the design parameter d . Table 2 represents the values of reliability index and design parameter during iteration process. Accuracy of the design parameter estimated is more than 98% for the particular problem.

6 Conclusions

From the results obtained, the iterative procedure significantly improves the quality of utilized response surface in a couple of iterations during structural reliability-based design process. In general, the initial response surface approximation coupled with inverse reliability analysis cannot result in sufficiently accurate design parameter. A number of iterations and successful convergence of the process depend on the original LSF and the design parameter range chosen during initial stages. Second-degree polynomial function may become insufficient for nonlinear problems, and use of a more complex surrogate model may be necessary. For complex systems, the RSM is the only way to approach both forward analysis and inverse analysis, due to unavailability of other method which can give the solution with an acceptable level of computational effort. To ensure accuracy of reliability-based design and reliability assessment, the iterative procedure should be performed while using RSM for inverse problems.

References

1. C.G. Bucher, U. Bourgund, A fast and efficient response surface approach for structural reliability problems. *Struct. Saf.* **7**(1), 57–66 (1990). [https://doi.org/10.1016/0167-4730\(90\)90012-E](https://doi.org/10.1016/0167-4730(90)90012-E)
2. I. Kaymaz, Application of Kriging method to structural reliability problems. *Struct. Saf.* **XXVII**(2), 133–151 (2005). <https://doi.org/10.1016/j.strusafe.2004.09.001>
3. D. Lehký, M. Šomodíková, Reliability calculation of time-consuming problems using a small-sample artificial neural network-based response surface method. *Neural Comput. Appl.* **XXVIII**(6), 1249–1263 (2017). <https://doi.org/10.1007/s00521-016-2485-3>

4. J.E. Hurtado, An examination of methods for approximating implicit limit state functions from the viewpoint of statistical learning theory. *Struct. Saf.* **XXVI**(3), 271–293 (2004). <https://doi.org/10.1016/j.strusafe.2003.05.002>
5. D. Lehký, D. Novák, Solving inverse structural reliability problem using artificial neural networks and small-sample simulation. *Adv. Struct. Eng.* **XV**(11), 1911–1920 (2012). <https://doi.org/10.1260/1369-4332.15.11.1911>
6. V. Kurukova, Kolmogorov's theorem and multilayer neural networks. *Neural Netw.* **5**(3), 501–506 (1992). [https://doi.org/10.1016/0893-6080\(92\)90012-8](https://doi.org/10.1016/0893-6080(92)90012-8)
7. H. Dai, W. Wang, Application of low-discrepancy sampling method in structural reliability analysis. *Struct. Saf.* **31**(1), 55–64 (2009). <https://doi.org/10.1016/j.strusafe.2008.03.001>
8. David Lehky, Martina Somodikova, Inverse response surface method in Reliability-based design. *Trans. VŠB: Tech. Univ. Ostrava, Civil Eng. Ser.* **17**, 37–42 (2017). <https://doi.org/10.1515/tvsb-2017-0025>
9. H. Li, R.O. Foschi, An inverse reliability method and its application. *Struct. Saf.* **XX**(3), 257–270 (1990). [https://doi.org/10.1016/S0167-4730\(98\)00010-1](https://doi.org/10.1016/S0167-4730(98)00010-1)
10. A.S. Balu, B.N. Rao, Inverse structural reliability analysis under mixed uncertainties using high dimensional model representation and fast Fourier transform. *Eng. Struct.* **37**, 224–234 (2012). <https://doi.org/10.1016/j.engstruct.2011.12.043>
11. J. Cheng, Q.S. Li, Application of the response surface methods to solve inverse reliability problems with implicit response functions. *Comput. Mech.* **43**(4), 451–459 (2009). <https://doi.org/10.1007/s00466-008-0320-0>
12. S.H. Kim, S.W. Na, Response surface method using vector projected sampling points. *Struct. Saf.* **19**(1), 3–19 (1997). [https://doi.org/10.1016/S0167-4730\(96\)00037-9](https://doi.org/10.1016/S0167-4730(96)00037-9)

Smart Thermal Data Match Using Machine Learning



Krishna Nelanti Suresh Thiraviyam Naresh Nidamanuri and Raviraj Barapu

Abstract Building an analytical thermal model to simulate actual working conditions of a jet engine is quite challenging. Many approximations creep into thermal analysis in the process of simulating real-time working environment. Therefore, it is mandatory to validate analytical temperature predictions against test engine data (thermal couple measured temperatures). Thermal boundary conditions (BCs), i.e., convective heat transfer coefficients (HTC), are estimated using empirical correlations which are derived from experiments conducted in controlled environments. Traditionally, BCs are varied within permissible limits to account the inability of empirical correlations in capturing physics. This approach requires large number of iterations to match analytical model against engine data. This requires manual efforts and consumes huge cycle time. Fine tuning BCs for the validation of analytical model varies from engine to engine based on its architecture and flow system. Rule-based algorithms would not help in data matching analytical models with less number of iterations. Machine learning (ML) algorithms would be an appropriate choice as it can use lessons from the legacy data matched models. A methodology is developed using regression-based ML algorithm which uses data from legacy thermal data matched models to identify the influence of convective HTCs on component temperatures. 70% of available data is used to train the ML model and the remaining 30% of data is used for validation. Further to improve the results, hyperparameters were tuned along with the k-fold cross validation. Several attempts have been made to improve prediction accuracy by employing various algorithms like Lasso, Ridge, Elastic Net, and XGBoost. Among all, multiple linear regression has yielded lowest

K. Nelanti · S. Thiraviyam · N. Nidamanuri · R. Barapu
Cyient Limited, Hyderabad, India
e-mail: Krishna.Nelanti@cyient.com

S. Thiraviyam
e-mail: Suresh.Thiraviyam@cyient.com

N. Nidamanuri
e-mail: Naresh.Nidamanuri@cyient.com

R. Barapu
e-mail: Raviraj.Barapu@cyient.com

mean squared error. An ensembling approach has also been applied with the best resulting algorithms among above. Though the results seem to be interesting, the objective of identifying weights could not be achieved with ensembling technique. Thus, the objective function is constructed with the results obtained from multiple linear regression and solved using simplex method to identify optimal scale factors to convective heat transfer coefficients. With this approach, user intervention is mostly eliminated. 50% of cycle time reduction is achieved.

Keywords Thermal data match machine learning · Heat transfer coefficient · Regression · Mean square error (MSE)

1 Introduction

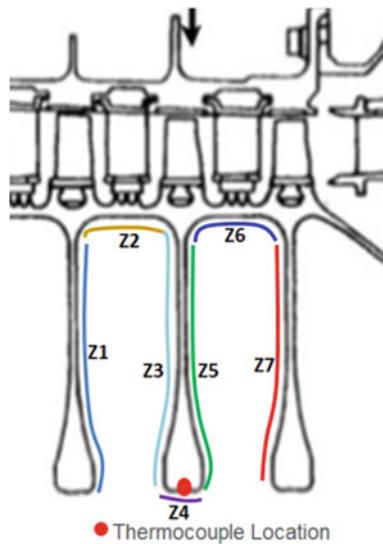
High-pressure compressor module in gas turbine engine comprises of many rotating cavities formed by adjacent rotating disks. Estimating the thermal boundary conditions in rotating cavities is extremely challenging as the cavities have through flow, unstable recirculation, buoyancy effects, and they have significant impact on metal temperatures. Accurate prediction of steady state and transient temperature distribution is critical in estimating component life and clearance requirements. Thermal data match plays a crucial role in validating analytical predictions which ensures exact physics is captured in the thermal model. Conventional data match process is consuming huge amount of time which impacts overall design cycle time.

2 High-Pressure Compressor Rotating Cavities Geometry

Typical high-pressure compressor rotating cavity is considered for this study as shown in Fig. 1. Boundary condition includes flow, convective heat transfer coefficients, and heat generations.

3 Thermal Boundary Conditions

Single high-power steady-state point representative of engine conditions is chosen for this analysis. Forced and free convection heat transfer coefficients play a major role in achieving thermal data match. Figure 1 shows the thermal zone map (z1 to z7) associated with the geometry. Thermocouple is instrumented at the rotor bore location close to z4. Table 1 lists independent variables for each zone and its corresponding dependent variable. X1, X2 through X13 are the scale factors on base heat transfer coefficients (independent parameters) and Y are the delta in temperature between engine data and analysis (dependent variable).

Fig. 1 HPC rotating cavity**Table 1** Sample data points

Z1		Z2		Z3		Z4	Z5		Z6		Z7		Y
X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	
2.7	1.2	2.2	2.4	1.6	1.6	1.3	1.3	1.8	0.7	2.1	2.6	1.8	-23
2.6	2.8	0.6	0.6	2.8	2.2	2.2	2.3	2.7	2.8	2.7	2.1	1.2	-32
2.3	1.4	2.9	1.3	2.9	1.2	0.7	0.7	1.8	0.8	2.8	2.4	1.2	-15
2.4	0.6	1.6	2.9	1.2	2.6	1.2	2.4	1.7	0.8	2.4	2.1	1.3	-16
0.8	0.8	2.8	2.8	0.6	1.3	1.4	2.8	0.8	1.9	2.8	0.9	1.6	-18
1.3	0.7	0.7	0.7	1.3	2.4	0.6	0.6	1.8	1.2	1.9	1.6	2.6	-10
1.2	2.6	1.3	2.7	2.1	0.7	2.1	2.2	1.1	2.9	2.9	2.1	1.6	-31
0.8	1.4	2.6	1.2	2.9	2.9	1.2	2.4	2.3	2.4	2.8	1.3	1.8	-17
0.7	1.4	0.8	2.2	1.6	2.9	0.9	2.4	2.1	1.3	1.8	2.8	1.2	-10
1.4	2.2	0.8	2.4	1.2	2.6	2.7	1.7	1.9	1.3	0.9	2.8	1.8	-35
2.8	2.1	1.2	2.6	1.8	2.4	1.4	2.9	2.7	2.1	1.4	0.6	0.6	-20
0.9	0.7	0.8	1.7	0.9	1.8	0.6	0.6	1.9	2.2	1.7	1.4	0.7	-7
1.6	2.4	1.1	2.2	2.3	1.3	1.8	1.4	1.8	0.6	2.2	2.6	1.7	-30
2.6	1.1	2.7	2.4	2.7	2.2	2.1	2.1	1.7	2.9	1.7	2.2	1.1	-30

4 Conventional Thermal Data Match Approach

Conventional thermal data match is performed manually by varying boundary conditions with the use of scale factors (Examples: X1 through X13 for this case), which are to be justified with factual physics. Standard empirical HTC correlation may not

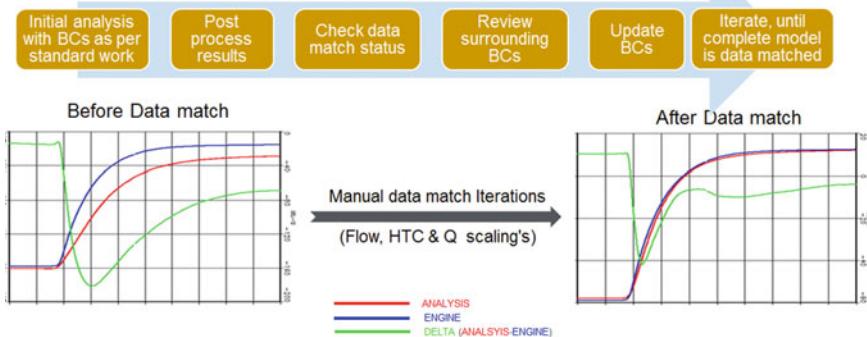


Fig. 2 Schematic diagram of conventional method

be appropriate for all real-time physical conditions; hence, scale factors are inevitable to achieve thermal data match. BCs are updated manually based on previous iteration data match status. This approach requires human intervention after every iteration as shown in Fig. 2, which results in huge data match cycle time.

5 Smart Thermal Data Match—ML Approach

The proposed method involves use of machine learning algorithms and optimization technique to achieve data match in less cycle time.

The proposed method includes following steps.

- Gather data from legacy thermal data matched models
- Data mining and exploratory data analysis
- Selection of suitable machine learning algorithms
- Fine tune ML algorithm for robustness
- Build optimization algorithm to predict independent variables.

Based on the above steps, the architecture for the problem was built, the schematic of which is shown in Fig. 3.

5.1 Legacy Data

Nearly, 30,000 historical data points were acquired from compressor rotor thermal data match analyses performed in last 15 years. Among which 70% of data is used to train the ML model and the remaining 30% as test set.

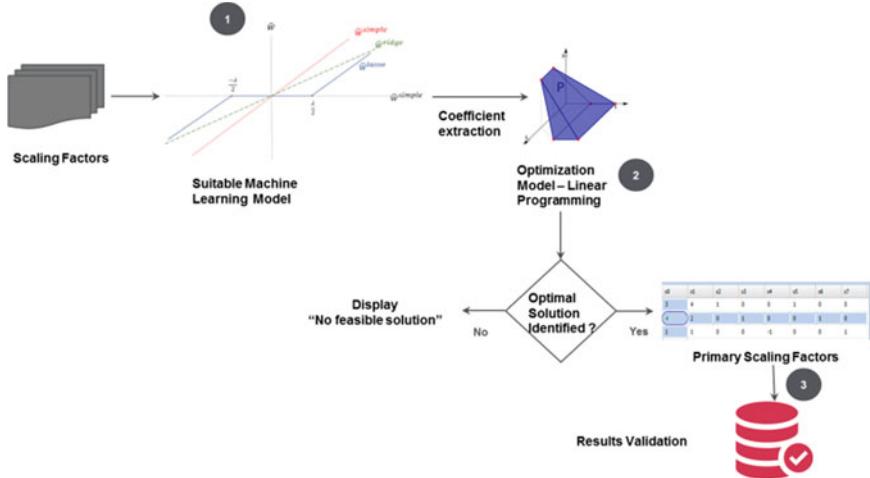


Fig. 3 Solution architecture

5.2 Exploratory Data Analysis

In order to elect the suitable machine learning technology, exploratory data analysis was performed to understand the data and the correlation across different variables. It is observed from data that both dependent and target variables were continuous which in turn justifies the selection of supervised algorithm.

Figures 4 and 5 show the data summary which explains that, for all the independent variables and dependent variables, there is an existence of skewness. The inference from this analysis is that the variables considered are not following normal distribution; thus, one can expect a less positive results if linear regression is employed.

One would be more interested to know the relation between target and predictors; thus, correlation analysis was performed and the results were visualized in Fig. 6. The plot suggests strong relationship only exists between X7 and Y, and there is no multicollinearity in the data.

Model	X1	X2	X3	X4	X5	X6	X7
A1:30000	Min. :0.600	Min. :0.600	Min. :0.60	Min. :0.600	Min. :0.600	Min. :0.600	Min. :-0.600
	1st Qu.:1.100	1st Qu.:1.200	1st Qu.:1.20	1st Qu.:1.100	1st Qu.:1.100	1st Qu.:1.100	1st Qu.:1.200
	Median :1.700	Median :1.800	Median :1.80	Median :1.800	Median :1.800	Median :1.700	Median :1.800
	Mean :1.751	Mean :1.754	Mean :1.76	Mean :1.748	Mean :1.749	Mean :1.747	Mean :1.753
	3rd Qu.:2.400	3rd Qu.:2.400	3rd Qu.:2.40	3rd Qu.:2.300	3rd Qu.:2.300	3rd Qu.:2.300	3rd Qu.:2.400
	Max. :2.900	Max. :2.900	Max. :2.90	Max. :2.900	Max. :2.900	Max. :2.900	Max. :2.900

	X8	X9	X10	X11	X12	X13	Y
	Min. :0.600	Min. :-46.00					
	1st Qu.:1.100	1st Qu.:1.200	1st Qu.:1.200	1st Qu.:1.100	1st Qu.:1.100	1st Qu.:1.100	1st Qu.:-34.00
	Median :1.700	Median :1.800	Median :1.800	Median :1.800	Median :1.700	Median :1.700	Median :-29.00
	Mean :1.745	Mean :1.754	Mean :1.756	Mean :1.753	Mean :1.746	Mean :1.744	Mean :-25.77
	3rd Qu.:2.300	3rd Qu.:2.400	3rd Qu.:2.400	3rd Qu.:2.400	3rd Qu.:2.300	3rd Qu.:2.300	3rd Qu.:-19.00
	Max. :2.900	Max. :6.00					

Fig. 4 Data summary

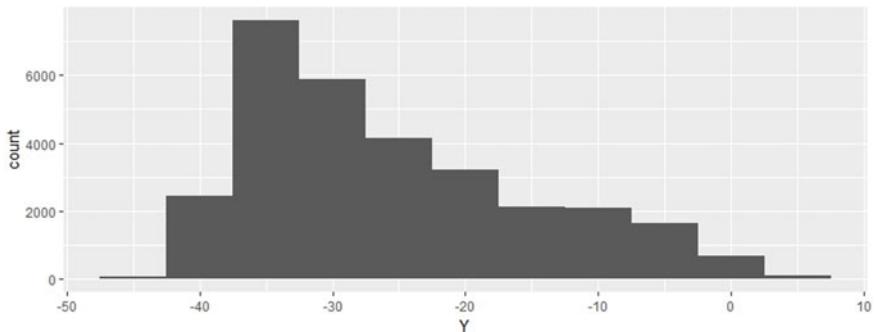


Fig. 5 Target variable distribution

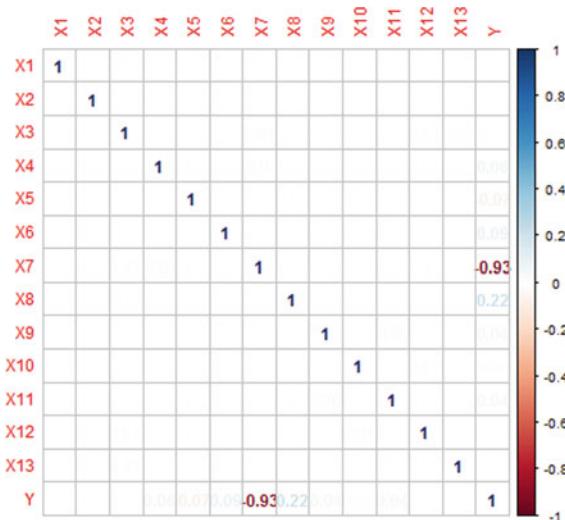


Fig. 6 Correlation analysis

After the thorough exploratory data analysis, need for transformations and feature selection seems to be inevitable. However, the engineering constraints and the objective of the problem would not allow doing so, as the procedure should yield weights. Thus, regression-type supervised learning methods are adopted to perform the analysis.

5.3 Multiple Linear Regression

One can realize the importance of the weight of each zone contributing to the temperature data match at the required location; thus, the solution of the problem in its clear vicinity should have to establish the weight for the surrounding zones followed by optimization to acquire required scaling factors for the given temperature. To accommodate multiple predictors, each predictor has given a separate slope coefficient in a single model; thus, our model takes the form as shown below,

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_j X_j + \epsilon \quad (1)$$

(a) Relationship Between the Response and Predictors

In order to determine whether there is a relationship between the response and the predictor in the multiple regression with p predictors, check whether all of the regression coefficients are zero. The hypothesis test is performed by computing the F-statistic.

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_p = 0 \quad (2)$$

Alternative,

$$H_a : \text{at least one } \beta_j \text{ is non zero}$$

The F-statistic provided the compelling evidence against null hypothesis as shown in Table 2. In other words, the large F-statistic [1] suggests that at least one of the thirteen predictor variables must be related to response.

Do all the predictors help to explain Y, or is only a subset of the predictors useful?

P-value is one of the important factors for feature selection process; all the variables were considered into the model and identified that few variables with largest p-value which is also called least statistically significant.

(b) Model Fit

Two of the most common numerical measures of model fit are the RSE and R^2 , the fraction of variance explained. An R^2 value close to 1 indicates that the model explains a large portion of the variance in the response variable. 92% variance of the

Table 2 Evaluation metrics

Quantity	Value
RSE	3.001
R^2	0.9194
F-statistic	1.842e + 04

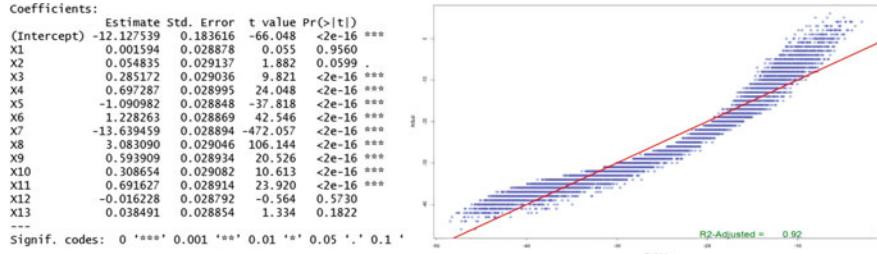


Fig. 7 Multiple linear regression—Results

dependent variable is explained by the model and 8% explained by the error term for this model as is evident in Fig. 7.

(c) Assumptions Validation

Performing multiple linear regression on the data set allowed us to establish few conclusions. Four out of the thirteen predictor variables do not have statistical significance in predicting the response variable, existence of nonlinear relationship, heteroscedasticity and correlation among error terms are evident. Thus, the basic assumptions of linear regression stays void, to sort them out one should adopt transformations and or other ML algorithms to establish required coefficients.

5.4 Lasso, Ridge and Elastic Net

After modeling the data through multiple linear regression, it become inevitable to build the model with other algorithms to make sure to have unbiased coefficients and eliminating the features which are not statistically significant. To do so, the so called shrinkage methods Lasso, Ridge and Elastic Net [2] were employed and acquired the results as given in Fig. 8.

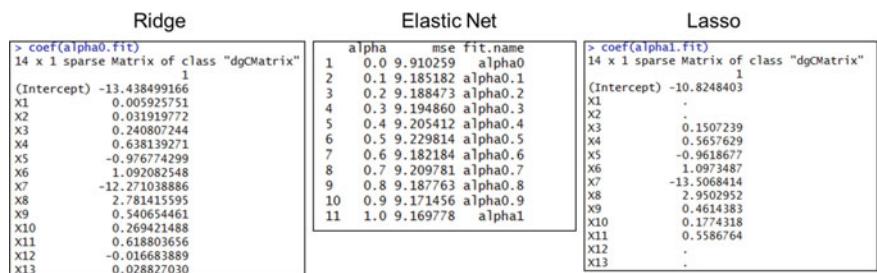


Fig. 8 Lasso, Ridge and Elastic Net—Results

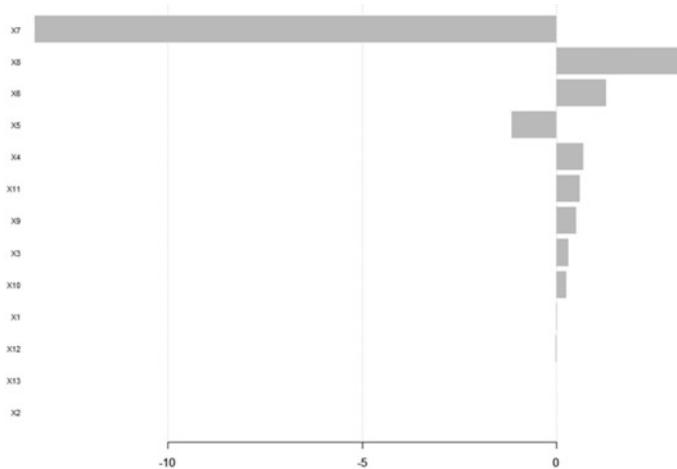


Fig. 9 XGBoost—Feature importance

It is evident from the results that Lasso is giving least mean squared error compared to Ridge or Elastic Net methods. As Lasso works on feature selection, the idea of obtaining weights for all independent variables cannot be achieved.

5.5 XGBoost

Boosting algorithms assist in converting weak learners into strong. The data set has been experimented with XGBoost algorithm and the feature importance is shown in Fig. 9. The results were assessed with mean squared error.

5.6 Multiple Algorithms and Quantitative Assessment

Several modeling approaches were tried on the required dataset and the prediction results were compared with the metric MSE. However, the traditional engineering problems allow the user the predictions to be in prescribed tolerance limit. One of the important metric for business justification is the prediction accuracy within the tolerable limit; hence, the results from different algorithms were compared.

Across all the applied algorithms, two factors have been considered for the assessment. One is MSE, lower the value better the accuracy and other is the count of test observation falling within the defined tolerance limits (1° , 2° , 3° temperature). Based on Fig. 10, it is concluded that multiple linear regression would be the best choice in the perspective of MSE. If the tolerance limit is to be chosen for the assessment, ridge would be an appropriate algorithm as is evident in Fig. 11.

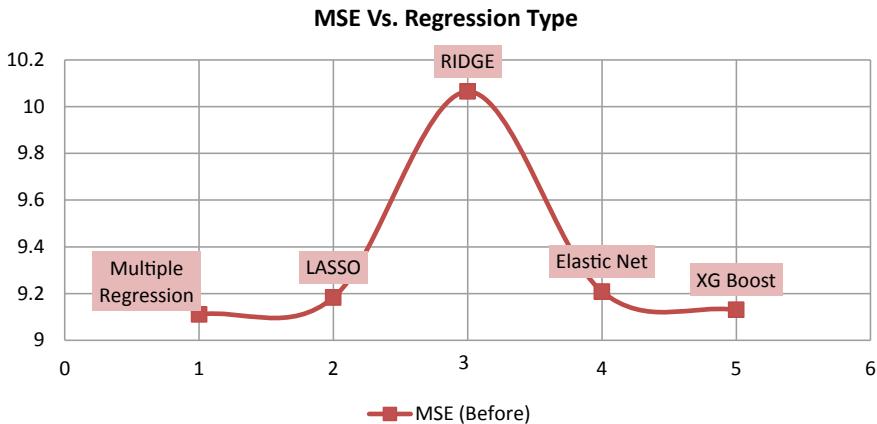


Fig. 10 MSE across various algorithms

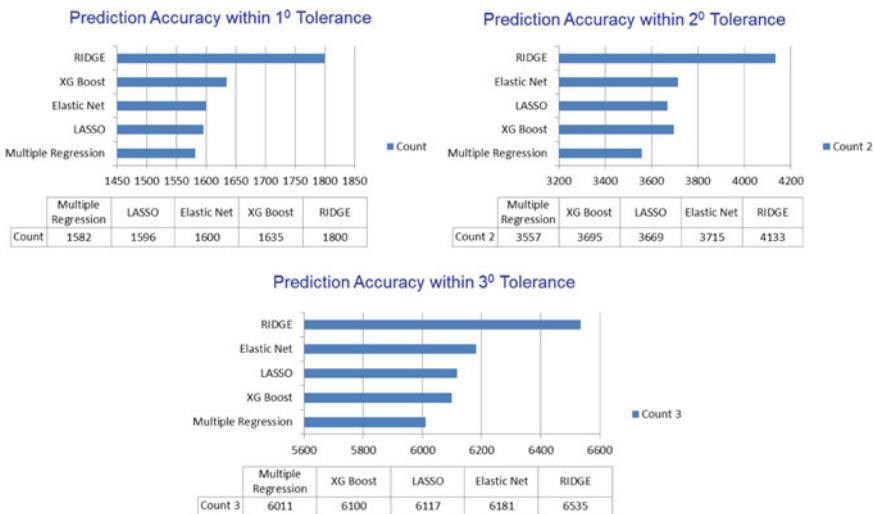


Fig. 11 Prediction accuracy within various tolerance limits

5.7 Ensembling Approach and Business Justification

Based on above conclusions, in order to incorporate the best features of above algorithms, ensembling methodology is implemented with multiple combinations. Results are assessed by considering both the metrics MSE and observations count within defined tolerances.

Table 3 Ensemble methodology results

Attempt	Base layer algorithm	Top layer algorithm	MSE	Prediction accuracy in 1 degree tolerance
1	LASSO	XG Boost	1.847	5741
	RIDGE			
	Multiple linear			
2	XG Boost	Multiple linear	0.219	8705
	LASSO			
	RIDGE			

Table 3 shows the results of ensemble method. XGBoost, Lasso and Ridge as base layer and multiple linear as top layer algorithm produced better results in terms of MSE and observations count within 1 degree tolerance.

Though the results are quite interesting, the ensemble methodology or a black box model could only provide us the coefficients of algorithm where as our need to identify the weight of each boundary condition remains blank. However, this methodology allows user to work on one alternate approach, i.e., for a given values of boundary condition scale factors for each zone, the ensemble model will let us know the precise value of target variable, i.e., data match status.

5.8 Model Selection

Across the applied algorithms, ensembling approach is performing best in all aspects but the objective of obtaining weights could not be achieved and same is the case with XGBoost. Though Ridge is yielding higher quantity of observations within the defined tolerance limits, its MSE is not so interesting. Lasso, however, depends on the feature elimination and is not ideal for the problem statement. Therefore, with least MSE and the option to attain coefficients, multiple linear regression though it is deviating from the assumptions found to be the best suitable algorithm in lieu with the problem statement.

To improve the multiple linear regression algorithm prediction accuracy, k-fold cross validation has been employed and the intercept is chosen as a hyperparameter for fine tuning the algorithm by using super learner package in R programming.

6 Optimization

After selecting the multiple linear regression as the best algorithm, model is built to learn from the training observations and the coefficients are obtained for all the independent variables. Thus, the objective function is constructed using the results

Table 4 Results validation

Y_Target	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	Y_Analysis
1	0	0	0.5	0.5	0.5	0.5	0.5	1.9	0.5	0.5	0.5	0	0	2

of multiple linear regression and is represented in its canonical form as shown below,

$$y = \sum_{k=1}^n \beta_k x_k \quad (3)$$

Once the objective function is known, optimized the independent variables based on the defined constraints and known value of target variable. Thus, this method helped to obtain optimized boundary condition scaling factors for the required target variable, i.e., data match status; the constraints are defined based on the standard work.

The formulation for the problem takes the below form,

$$\min(z) = \sum_{k=1}^n \beta_k x_k \quad (4)$$

Subjected to, $0.5 \leq x_{i...n} \leq 3.0$

The objective function includes 13 decision variables subjected to 28 constraints. Results from linear programming are tested on the actual analytical model and achieved y_analysis is 2 degree against y_target of 1 degree as given in Table 4.

7 Conclusions

For estimating optimized thermal boundary conditions to validate analytical model against test engine results, various algorithms like multiple linear regression, Lasso, Ridge, Elastic Net, and XGBoost are explored. An ensembling method is also attempted. Multiple linear regression in conjunction with linear programming method seems to be appropriate with the problem statement and is useful to carry out thermal data match. This method eliminated manual intervention and reduced the cycle time by 50% for steady-state analysis and can be extended to full-cycle transient analysis. Results can be further improved by addressing the nonlinearity in the data and or by employing feature elimination methods, also adopting the AI methods like ANN, CNN may be helpful.

References

1. G. James, D. Witten, T. Hastie, R. Tibshirani, *An Introduction to Statistical Learning* (Springer, USA, 2013)
2. T. Hastie, R. Tibshirani, J. Fried Man, *The Elements of Statistical Learning* (Springer, USA, 2008)

Identification of Aerodynamic Derivatives of a Re-entry Module



C. Priyant Mark and K. Srinivasan

Abstract Blunt body shape is used for re-entry modules to withstand the aerodynamic heating loads in the hypersonic regime. During the terminal phase of descent, it encounters the low subsonic regime. In this phase, stable descent and favorable attitude for deceleration system deployment are of paramount importance. For ensuring the requisite state, the dynamic stability of the body during the period needs to be characterized. Captive experiments do not provide accurate results due to sting effects. Hence, free-flight tests are performed using a dynamically scaled model. But only the accelerations and angular rates during the flight can be measured. To estimate the dynamic stability derivatives from the data, the system must be identified. It is considered as a nonlinear least squares problem, which can be solved using an unconstrained nonlinear method. The total coefficients are extracted using an analytical approach which solves the 6-degree of freedom (6-DOF) equations of motion. A set of neural networks are designed to map the coefficients to the influencing parameters. The networks are trained and optimized using Levenberg–Marquardt backpropagation algorithm, which interpolates between Gauss–Newton and gradient descent methods. The parameters are perturbed to obtain the stability derivatives. The data from real-time free-flight tests of a re-entry module is used to estimate the parameters, using the proposed method.

Keywords Blunt body stability · Free-flight tests · 6-DOF · System identification · Neural networks

C. P. Mark

Department of Instrumentation and Control Engineering, Manipal Institute of Technology,
Manipal Academy of Higher Education, Manipal, KA, India
e-mail: priyant.mark.c@gmail.com

K. Srinivasan

Wind Tunnel Data Division, Wind Tunnel Group, Aeronautics Entity,
Vikram Sarabhai Space Centre, Thiruvananthapuram, KL, India
e-mail: k_srinivasan@vssc.gov.in

1 Introduction

A human-rated spacecraft designed for a mission to space consists of a re-entry module, a service module, and a launch abort system. The re-entry module is used in the re-entry phase, which is the treacherous part of the mission profile. The module must endure prolonged periods of hypersonic air velocity, scorching temperatures, and immense vibrations during re-entry. When it has dropped enough altitude, it reaches the lower, denser atmosphere where the module will face its most turbulent stage. This phase is classified into the low subsonic regime, where dynamic instabilities diverge and promote disastrous effects, such as uncontrolled tumbling. In this phase, the module must be in the required attitude for the deceleration systems to be deployed. In order to gather the data during this phase and understand the state in the region, a scaled model test must be performed. Even though various captive test techniques, ballistic range, and vertical wind tunnel-based test can be performed, none is close enough in simulating the real flight conditions. A method which can re-create the same conditions to obtain accurate data has to be explored. Hence, a sublime method must be been designed and developed for testing and determining the dynamic stability of a re-entry module at the low subsonic regime.

Aubuchon et al. [1] use wind tunnel experimental techniques to determine dynamic stability of blunt bodies. This method will have a lot of sting effects which the proposed method addresses. Murman and Aftosmis [2] use CFD and in-flight data for stability assessment. The proposed method will give far accurate results. Potturi and Peroomian [3] use data from CFD as well as wind tunnels to characterize the aerodynamics of a return capsule. These methods do not reflect the flight conditions during testing. Nomura et al. [4] use numerical methods to analyze the dynamic stability of a re-entry capsule. This method will not incorporate the uncertainties of the atmosphere. Albisser [5] uses free-flight tests inside a ballistic range, recorded by high-speed cameras and shadowgraphs. This method is useful for hypersonic test regime. Owens and Aubuchon [6] use sub-scale free-flight tests, free-to-oscillate, ballistic range, and forced oscillation techniques. These methods do not cater to the subsonic regime or not accurate. Inatani [7] uses a combination of experimental, analytical, and computational methods but do not use outdoor free-flight test. All indoor tests like vertical wind tunnels and ballistic range do not reproduce actual flight conditions. Nomura et al. [8] use a transonic wind tunnel to study the pitching oscillation and flow field around a blunt body. The proposed method will avoid wind tunnels due to stings and its influences. Kobayashi et al. [9] use the forced oscillation technique coupled with CFD. This method will not match the detail obtained in an open atmospheric test.

Youssef and Juang [10] uses neural networks for estimating the aerodynamics by involving the coefficients as outputs. The apriori fix of the coefficients will be utilized in the proposed method. Raol [11] use recurrent neural networks to estimate aerospace dynamic systems. The estimated model uses an indirect approach to estimate the parameters while the proposed method uses a direct approach. Hamel and Jategaonkar [12] give a description of the evolution of system identification

methods. The neural networks method seems to give the best result at that stage. [13–18] use the delta method and zero method of feedforward neural networks along with Gauss–Newton and fuzzy logic. These methods can be reliably used for estimating the derivatives. It estimates the derivatives separately for longitudinal and lateral-directional dynamics, but the proposed method is used for estimating all the derivatives at once.

It shows that the proposed method will be unique in its approach in more than one way and will give the best results to extract the aerodynamic derivatives.

2 Methodology

In order to characterize the dynamic stability of the re-entry module, a dynamically scaled model must be designed for testing. Then, it must be embedded with sensors and used in tests to obtain data. The data must be used in 6-DOF equations to extract the total coefficients. A set of neural networks are created to map the influencing parameters to the coefficients, and the delta method is used to identify the derivatives.

2.1 *Dynamically Scaled Model*

Dynamic scaling is performed to simulate the motions and forces of the original prototype on the model. It helps maintain the ratio of inertial and gravitational effects between them. The equation of motion is of the form: *inertial force* = *weight* + *aerodynamic force*. The scaling parameters directly obtained from the equation are Froude number and mass ratio. The scale factors for other parameters must be derived by non-dimensionalizing the equations of motion using Buckingham Pi theorem. The respective parameters must be scaled by matching Reynolds number and Froude number [19]. The model also has geometric and mass distribution similarity. Once the scale is obtained, it is modeled and fabricated for testing.

2.2 *Experiments*

The dynamically scaled model is embedded with sensors (accelerometers and gyroscopes) and supporting electronics and is used in low altitude drop tests. The data (accelerations and rates) for each test profile is downloaded and processed for system identification. The data can also be obtained from any other test methods. The coordinate system identified is given in Fig. 1.

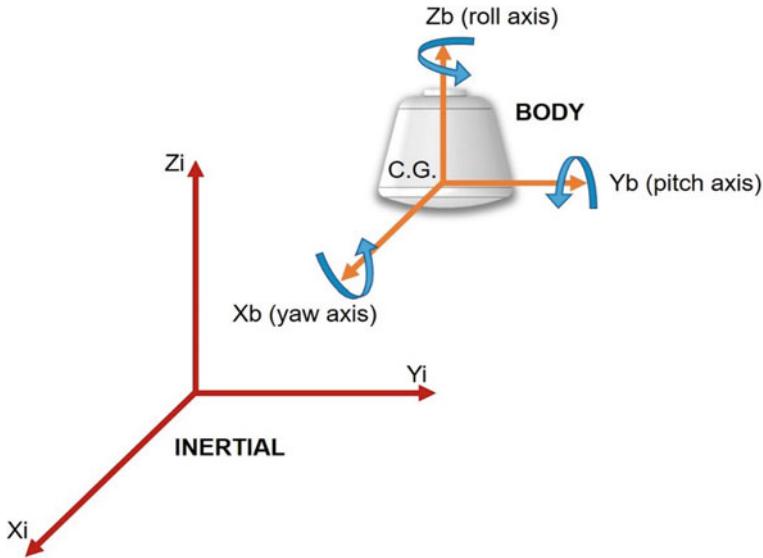


Fig. 1 Coordinate system

2.3 6-DOF Equations of Motion

To extract the total coefficients, the 6-DOF equations [20] must be solved. The force coefficients are determined by Eqs. (1)–(3).

$$C_x = \frac{m(\dot{u} - g_x)}{\bar{q} A_{\text{ref}}} \quad (1)$$

$$C_y = \frac{m(\dot{v} - g_y)}{\bar{q} A_{\text{ref}}} \quad (2)$$

$$C_z = \frac{m(\dot{w} - g_z)}{\bar{q} A_{\text{ref}}} \quad (3)$$

where C_x , C_y , and C_z are the force coefficients. \dot{u} , \dot{v} , and \dot{w} are the acceleration in the inertial frame. g_x , g_y , and g_z are the acceleration due to gravity in the inertial frame. m is the mass of the model. \bar{q} is the dynamic pressure. A_{ref} is the reference area of the model.

The moment coefficients are determined by Eqs. (4)–(6).

$$C_l = \frac{\dot{p} I_{xx} - (I_{yy} - I_{zz})qr}{\bar{q} A_{\text{ref}} d} \quad (4)$$

Table 1 Neural network parameters

Parameters	Value
Number of hidden layers	15
Number of hidden nodes	7
Data scaling range	-0.5 to 0.5
Nonlinear function slopes	0.6
Learning rate parameter	0.1
Momentum parameter	0.4
Initial random weights	0.5

$$C_m = \frac{\dot{q} I_{yy} - (I_{zz} - I_{xx}) pr}{\bar{q} A_{\text{ref}} d} \quad (5)$$

$$C_n = \frac{\dot{r} I_{zz} - (I_{xx} - I_{yy}) pq}{\bar{q} A_{\text{ref}} d} \quad (6)$$

where C_l , C_m , and C_n are the yawing, pitching, and rolling coefficients, respectively. p , q , and r are the angular rates. \dot{p} , \dot{q} , and \dot{r} are the angular accelerations. The moments of inertia are denoted by I_{xx} , I_{yy} , and I_{zz} . d is the reference diameter of the model.

2.4 Neural Network Training

A set of feedforward neural networks is designed for each output with the optimum properties listed in Table 1. The networks are described in Fig. 2. Levenberg–Marquardt backpropagation algorithm [21] is used for training and optimizing the networks. In the performance function, the Hessian computation is skipped to achieve the second-order training speed. It is made possible by approximating the Hessian, and a Jacobian is used in the gradient, which holds errors with respect to weights and biases. Due to the approximation, when the weight is large, it becomes a gradient descent method and when it is small, it becomes Newton's method. Using this weight, the optimizer can be switched to either of the methods to find the solution in the fastest possible manner. At each iteration, the performance function is reduced to find the optimal solution.

2.5 Aerodynamic Coefficients Model

The aerodynamic coefficients model is postulated in Eqs. (7)–(12). This is a simplified model, used only for demonstrating the concept. A higher complexity model considers various noises.

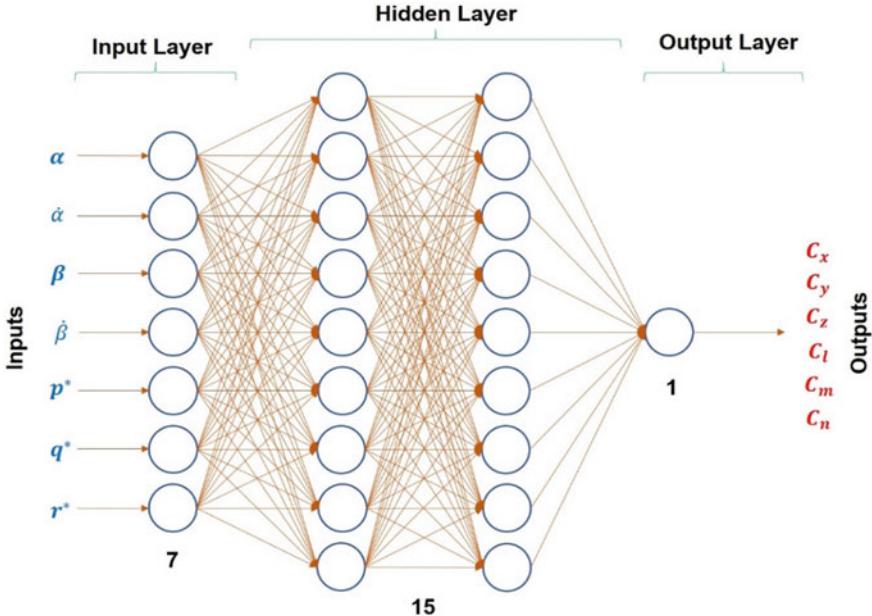


Fig. 2 Designed feedforward neural network(s)

$$C_x = C_{x\alpha}\alpha \quad (7)$$

$$C_y = C_{y\beta}\beta \quad (8)$$

$$C_z = C_{z\alpha}\alpha \quad (9)$$

$$C_l = C_{l\beta}\beta + C_{l\dot{\beta}}\dot{\beta} + C_{lp}p^* \quad (10)$$

$$C_m = C_{m\alpha}\alpha + C_{m\dot{\alpha}}\dot{\alpha} + C_{mq}q^* \quad (11)$$

$$C_n = C_{n\beta}\beta + C_{nr}r^* \quad (12)$$

where $C_{x\alpha}$, $C_{z\alpha}$, and $C_{y\beta}$ are the force coefficients due to the angle of attack and sideslip angle. α , β , $\dot{\alpha}$, and $\dot{\beta}$ are the angle of attack, sideslip angle, and their rates, respectively. $C_{l\beta}$, $C_{m\alpha}$, and $C_{n\beta}$ are the yaw coefficient due to side slip angle, pitch coefficient due to the angle of attack, and roll coefficient due to side slip angle, respectively. $C_{l\dot{\beta}} + C_{lp}$, $C_{m\dot{\alpha}} + C_{mq}$, and C_{nr} are the yaw, pitch, and roll damping coefficients, respectively. p^* , q^* , and r^* are the normalized angular rates and are given by Eqs. (13)–(15). The stability derivatives are summarized in Table 2.

Table 2 Summary of variables

Coefficients	Dependent variable	Independent variables	Parameters
Force in X	C_x	α	$C_{x\alpha}$
Force in Y	C_y	β	$C_{y\beta}$
Force in Z	C_z	α	$C_{z\alpha}$
Yawing moment	C_l	$\beta, \dot{\beta}, p^*$	$C_{l\beta}, C_{l\dot{\beta}} + C_{lp}$
Pitching moment	C_m	$\alpha, \dot{\alpha}, q^*$	$C_{m\alpha}, C_{m\dot{\alpha}} + C_{mq}$
Rolling moment	C_n	β, r^*	$C_{n\beta}, C_{nr}$

$$p^* = p \frac{d}{2V} \quad (13)$$

$$q^* = q \frac{d}{2V} \quad (14)$$

$$r^* = r \frac{d}{2V} \quad (15)$$

where V is the velocity in the body frame.

2.6 Stability Derivatives Estimation

The stability derivatives are estimated from the extracted total coefficients by the delta method [22], which uses the trained neural networks. These derivatives are assumed to be the change in aerodynamic force and moment coefficients due to a variation in one of the motion variables while the rest are kept constant. Using the inputs and outputs of the neural network, the delta method is applied. To use it, one of the inputs is perturbed by two sets of a small change, while the others are kept unchanged. The perturbed coefficient due to $\alpha + \Delta\alpha$ and $\alpha - \Delta\alpha$ can be denoted as C_x^+ and C_x^- , respectively. Hence, the derivative is approximated as in Eq. (16).

$$C_{x\alpha} = \frac{C_x^+ - C_x^-}{2\Delta\alpha} \quad (16)$$

Similarly, the different derivatives are estimated for all the data range.

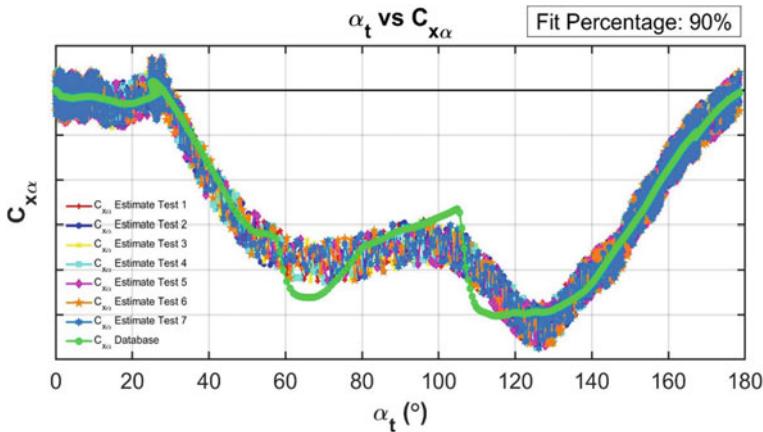


Fig. 3 Comparison of $C_{x\alpha}$

3 Results and Discussion

The above-discussed method can be applied to any real-time problem. A dynamically scaled re-entry module must be used in drop tests or any other method, and the measured data should be collected. The total coefficients should be extracted from the data, and the neural networks must be designed. The networks should be trained and optimized for an acceptable fit. Using the estimation method, the networks' inputs must be perturbed to obtain the stability derivatives. And hence, $C_{x\alpha}$, $C_{y\beta}$, $C_{z\alpha}$, $C_{l\beta}$, $C_{l\dot{\beta}} + C_{l\dot{\rho}}$, $C_{m\alpha}$, $C_{m\dot{\alpha}} + C_{m\dot{q}}$, $C_{n\beta}$, and C_{nr} can be determined for each instance. The results can be plotted with respect to α_t (total angle). A set of live test data was used to validate the method. The obtained static derivatives were compared with the static data from wind tunnel experiments database. They are shown in Figs. 3, 4, 5, 6, and 7.

From the results, it is found that the estimated derivatives are within the acceptable error criterion. These results will give an insight into the dynamic stability of a re-entry module in the low subsonic regime. The significance of the stability derivatives can also be analyzed from the results.

4 Conclusions

The pragmatic method to characterize the dynamic stability of a re-entry module at low subsonic regime has been investigated. This method is compatible with any type of experiment to obtain data. The mapping of the total coefficients using neural networks, along with the training and optimization, is explored. The identification of

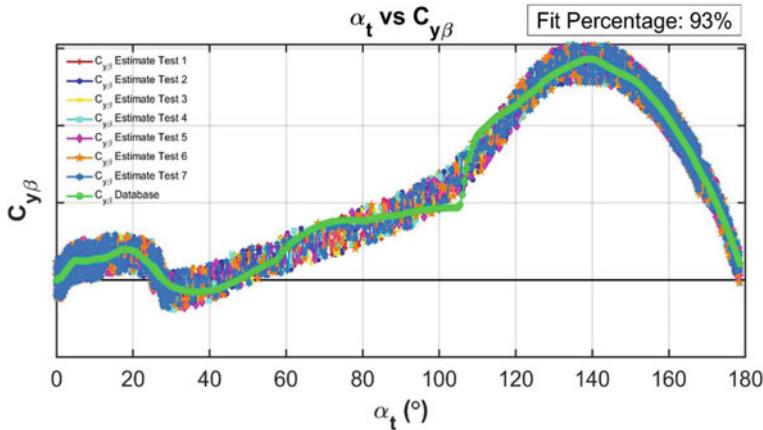


Fig. 4 Comparison of $C_{y\beta}$

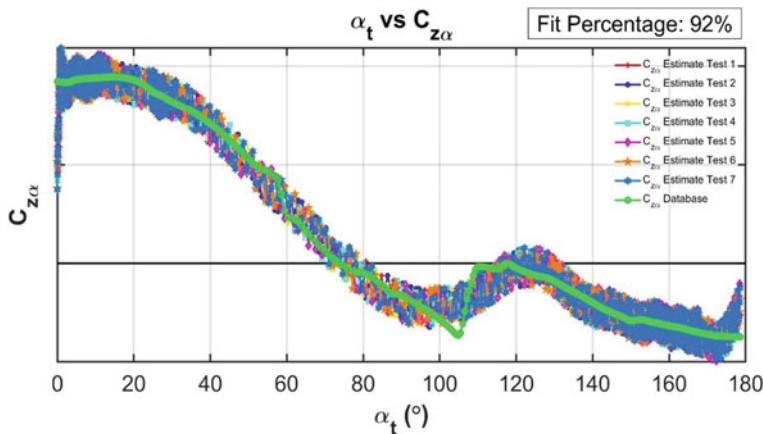


Fig. 5 Comparison of $C_{z\alpha}$

the aerodynamic derivatives using the trained networks can be evaluated using real-time data. A live test data was used to validate the method. This method is distinct because it can determine all the stability derivatives without de-coupling.

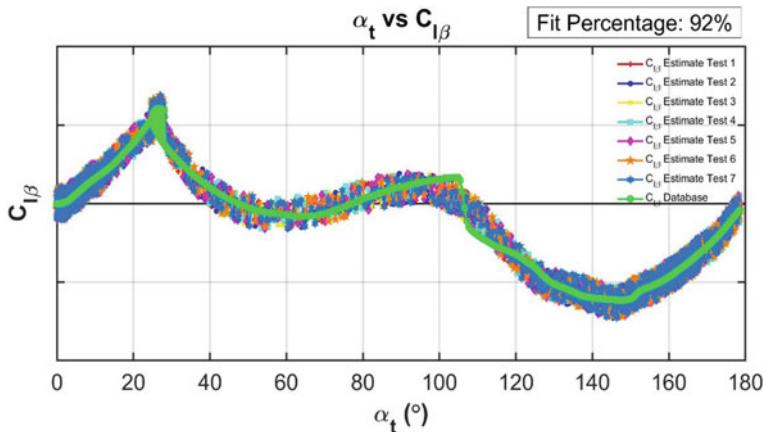


Fig. 6 Comparison of $C_{l\beta}$

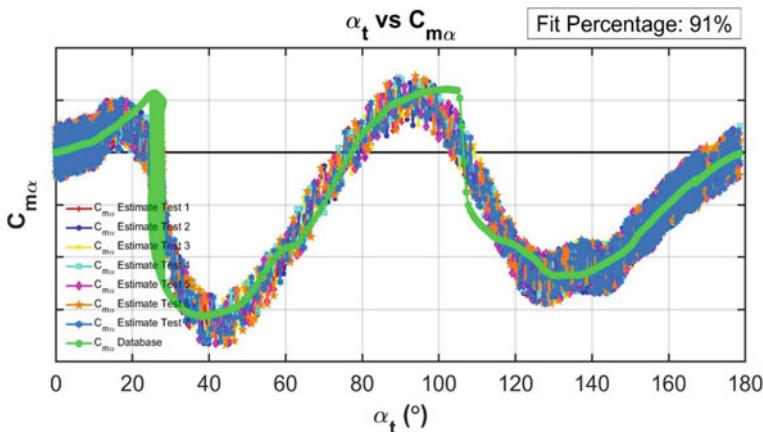


Fig. 7 Comparison of $C_{m\alpha}$

References

1. V.V. Aubuchon, D.B. Owens, O. Carvajal, S.B. Lumb, Dynamic stability of the boeing cst-100, in *AIAA Atmospheric Flight Mechanics Conference*, p. 1399 (2017). <https://doi.org/10.2514/6.2017-1399>
2. S. Murman, M. Aftosmis, Dynamic analysis of atmospheric-entry probes and capsules, in *45th AIAA Aerospace Sciences Meeting and Exhibit*, p. 74 (2007). <https://doi.org/10.2514/6.2007-74>
3. A.S. Potturi, O. Peroomian, Dynamic stability analysis of the orion crew module through computational fluid dynamics, in *46th AIAA Fluid Dynamics Conference*, p. 3200 (2016). <https://doi.org/10.2514/6.2016-3200>
4. M. Nomura, K. Fujita, H. Nagai, Computational study of aerodynamic characteristics on reentry capsule aft-body shape in transonic flow, in *2018 AIAA Atmospheric Flight Mechanics*

- Conference*, p. 0290 (2018). <https://doi.org/10.2514/6.2018-0290>
- 5. M. Albisser, Identification of aerodynamic coefficients from free flight data. Ph.D. dissertation, Universite de Lorraine (2015)
 - 6. D.B. Owens, V. Aubuchon, Overview of orion crew module and launch abort vehicle dynamic stability, in *29th AIAA Applied Aerodynamics Conference*, p. 3504 (2011). <https://doi.org/10.2514/6.2011-3504>
 - 7. Y. Inatani, Reentry capsule for sample return from asteroids in the planetary exploration missions. *J. Phys. Conf. Ser.* **1005**(1), 012049 (2018). <https://doi.org/10.1088/1742-6596/1005/1/012049>
 - 8. M. Nomura, A. Robbe, K. Fujita, H. Nagai, Reentry capsule dynamic instability with different aft-body shape in transonic flow, in *AIAA Scitech 2019 Forum*, p. 0018 (2019). <https://doi.org/10.2514/6.2019-0018>
 - 9. K. Kobayashi, Y. Ohmichi, M. Kanazaki, Modal decomposition analysis of subsonic unsteady flow around an atmospheric Entry capsule with forced oscillation, in *AIAA Scitech 2019 Forum*, p. 1851 (2019). <https://doi.org/10.2514/6.2019-1851>
 - 10. H. Youssef, J.C. Juang, Estimation of aerodynamic coefficients using neural networks, in *Flight Simulation and Technologies*, p. 3639 (1993). <https://doi.org/10.2514/6.1993-3639>
 - 11. J. Raol, Neural network based parameter estimation of unstable aerospace dynamic systems. *IEE Proc. Control Theory Appl.* **141**(6), 385–388 (1994). <https://doi.org/10.1049/ip-cta:19941506>
 - 12. P.G. Hamel, R.V. Jategaonkar, Evolution of flight vehicle system identification. *J. Aircraft* **33**(1), 9–28 (1996). <https://doi.org/10.2514/3.46898>
 - 13. A. Kumar, S. Saderla, A.K. Ghosh, Aerodynamic parameter estimation using neuro-fuzzy model based method, in *2017 First International Conference on Recent Advances in Aerospace Engineering (ICRAAE)* (IEEE, 2017), pp. 1–5. <https://doi.org/10.1109/icraae.2017.8297220>
 - 14. D.J. Singh, N.K. Verma, A. Ghosh, J. Sanwale, A. Malagaudanavar, Aerodynamic parameter estimation using two-stage radial basis function neural network, in *2017 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)* (IEEE, 2017), pp. 461–467. <https://doi.org/10.1109/sdpc.2017.93>
 - 15. A. Kumar, A.K. Ghosh, Anfis-delta method for aerodynamic parameter estimation using flight data. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.*, 0954410018791621 (2018). <https://doi.org/10.1177/0954410018791621>
 - 16. A. Kumar, A.K. Ghosh, Data-driven method based aerodynamic parameter estimation from flight data, in *2018 AIAA Atmospheric Flight Mechanics Conference*, p. 0768 (2018). <https://doi.org/10.2514/6.2018-0768>
 - 17. S. Saderla, D. Rajaram, A. Ghosh, Lateral directional parameter estimation of a miniature unmanned aerial vehicle using maximum likelihood and neural gauss newton methods. *Aeronaut. J.* **122**(1252), 889–912 (2018). <https://doi.org/10.1017/aer.2018.36>
 - 18. M. Winter, C. Breitsamter, Nonlinear identification via connected neural networks for unsteady aerodynamic analysis. *Aerosp. Sci. Technol.* (2018). <https://doi.org/10.1016/j.ast.2018.03.034>
 - 19. T.G. Gainer, S. Hoffman, Summary of transformation equations and equations of motion used in free flight and wind tunnel data reduction and analysis. NASA Technical Report (1972)
 - 20. E. Bernard, *Dynamics of Atmospheric Flight* (Wiley, 1972)
 - 21. H. Demuth, M. Beale, *Neural Network Toolbox* (The MathWorks Inc., 1992)
 - 22. V.J. Ravindra, *Flight Vehicle System Identification: A Time-Domain Methodology* (American Institute of Aeronautics and Astronautics, Inc., 2015). <https://doi.org/10.2514/4.102790>

Topology Optimization Using Convolutional Neural Network



Baki Harish, Kandula Eswara Sai Kumar, and Balaji Srinivasan

Abstract Topology optimization is a method to find the optimal material distribution of a structure by minimizing the objective function under the design and limit constraints. In this paper, we developed a deep learning-based machine learning algorithm to get the optimized structure for the given input conditions of a structure. We trained convolutional neural network (CNN)-based encoder–decoder architecture using the existing dataset as target images and input conditions modeled as input images. The target images are the optimized structures, developed using the MATLAB open-source topology optimization code, generated by varying the volume fraction from 5 to 95% with an increment of 5% and Poisson’s ratio varied from 0.01 to 0.49. The input conditions, i.e., the volume fraction and Poisson’s ratio are modeled as input images. In the present study, four types of input images and two encoder–decoder architectures are developed, and their performance is studied using identity mapping to obtain the optimized structure of a cantilever beam which is fixed at one end and a constant load is applied at the other end.

Keywords Topology optimization · Machine learning · Convolution–deconvolution network

1 Introduction

Structural topology optimization finds the optimal material distribution of a structure by minimizing the objective function under the design constraints. This method is widely used to design the low-weight structures which increase the efficiency of the

B. Harish (✉) · K. Eswara Sai Kumar · B. Srinivasan
Indian Institute of Technology Madras, Chennai, Tamil Nadu 600030, India
e-mail: harish.iiitn@gmail.com

K. Eswara Sai Kumar
e-mail: me15d417@smail.iitm.ac.in

B. Srinivasan
e-mail: sbalaji@iitm.ac.in

system. Sigmund et al. [1] provided free MATLAB source code to solve the topology optimization problem for compliance minimization.

Bruns et al. [2] first introduced the solid isotropic material with penalization (SIMP) method in topology optimization. The basic principle behind the SIMP method is it uses a density design variable that depends on material constitutive law and penalizes the intermediate density material in combination with an active volume constraint. Sosnovik et al. [3] first introduced the deep learning-based convolution–deconvolution network model to topology optimization and significantly improved the efficiency of the optimization process by considering the problem as an image segmentation task. In his model, intermediate results of SIMP are taken as inputs and final optimized structures are taken as targets. His model successfully mapped the intermediate results of SIMP with optimized final structures and decreased the computation time and cost significantly. However, his work did not consider the initial conditions for topology optimization, and the accuracy of the result heavily relies on the first few iterations. Thus, this method cannot completely eliminate the traditional SIMP method.

Banga et al. [4] have developed 3D convolution–deconvolution network, which takes intermediate 3D images from traditional solvers as inputs and final optimal 3D designs as targets. This method also reduced the time consumption, but this method also did not start from initial conditions.

In the present work, two types of deep learning-based encoder–decoder networks are developed, and four types of input images are modeled from input conditions. The present work focused on modeling the input conditions as input images to the proposed encoder–decoder network and mapping these input images with final optimal designs.

2 Methodology

2.1 Topology Optimization

The structural topology optimization is used to solve compliance minimization, eigenfrequency maximization, and design of compliant mechanisms of the structures. The compliance minimization problem is defined as:

$$\min C(\mathbf{x}) \quad (1)$$

$$\text{Subjected to } \mathbf{Ku} = \mathbf{f} \quad (2)$$

$$V(\mathbf{x}) \leq V^* \quad (3)$$

$$0.001 \leq \mathbf{x} \leq 1 \quad (4)$$

where \mathbf{x} is the design variable vector, \mathbf{K} is stiffness matrix, \mathbf{u} is displacement vector, \mathbf{f} is force vector, $V(\mathbf{x})$ is the design volume, and V^* is required volume.

In the present study, a 2D cantilever beam fixed at one end and a constant load applied at other end is considered to work on. The beam will have unique optimized design corresponding to volume percentage, Poisson's ratio, fixed boundary conditions, and application of force location. For the sake of simplicity, only volume percentage and Poisson's ratio are considered as variables for optimization.

2.2 Architecture of Encoder–Decoder Network

The encoder–decoder network is a deep neural network, which consists of two parts:

1. **Encoder network:** The input image is down-sampled by a series of convolution and pooling operations to a final vector of size 1024. The input image is reduced almost 20 times.
2. **Decoder network:** The output vector from encoder part is up-sampled by a series of deconvolution and upsampling operations to a final image of size same as the target image.

We have developed two types of encoder network. First network has two convolution operations and one fully connected operation as shown in Fig. 1. Second network has three convolution operations and one fully connected operations as shown in Fig. 2. In both networks, decoder network is the same. The convolution filters are squared matrices which are the weights we acquire after training. The pooling operation which we used here is average pooling, it reduces the convolved layer to the target size by doing averaging of elements. The convolution and pooling operations are said to be one convolution layer. The convolution and upsampling operations together said to be one deconvolution layer. In the upsampling operation, the convolved layer is expanded to a target size. Though explicitly not mentioned, ReLU is the activation function in every convolution and deconvolution operation and the mean square error is the cost function.

2.3 Preparing Target Data Set

We have generated optimized structures of the cantilever beam by varying the aspect ratio from 5 to 95% with an increment of 5% and Poisson's ratio from 0.01 to 0.49 with an increment of 0.01 using the MATLAB code. These images are used as target dataset for encoder–decoder network. Each image in the target dataset has a size of (100×100) .

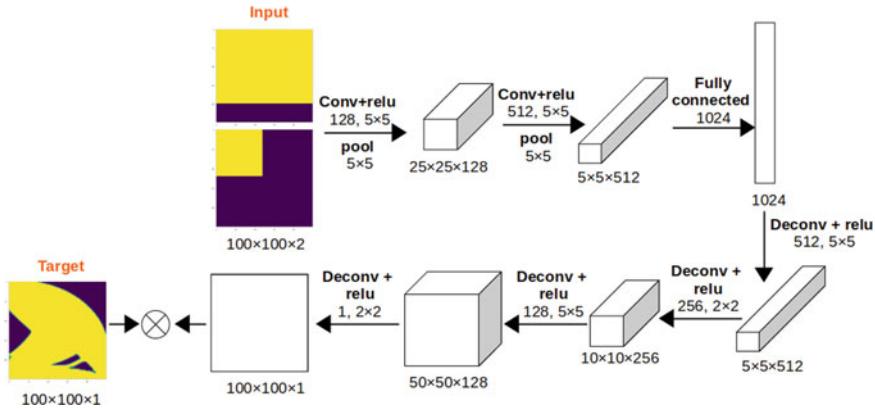


Fig. 1 1st encoder-decoder architecture

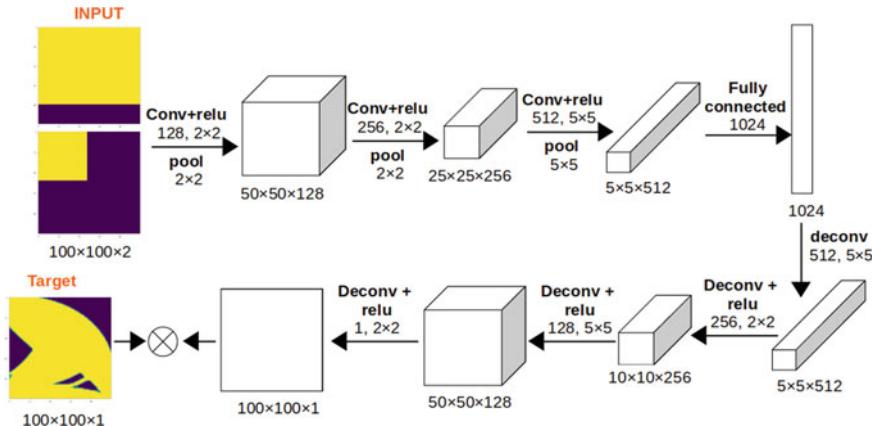


Fig. 2 2nd encoder-decoder architecture

2.4 Modeling the Input Images

We have modeled four types of input images by combining volume percentage and Poisson's ratio in different combinations.

Input type 1 images consist of two layers. First layer is (100×100) size image, has volume percentage information, modeled by filling volume percentage number of rows with ones completely and rest by zeros. Second layer is (100×100) size image, has Poisson's ratio information, modeled by filling $100 \times$ Poisson's ratio of rows and columns with ones and rest by zeros. Refer to Fig. 3a.

Input type 2 images consist of two layers. First layer has volume percentage information, modeled similar to Input type 1. Second layer is (100×100) size image,

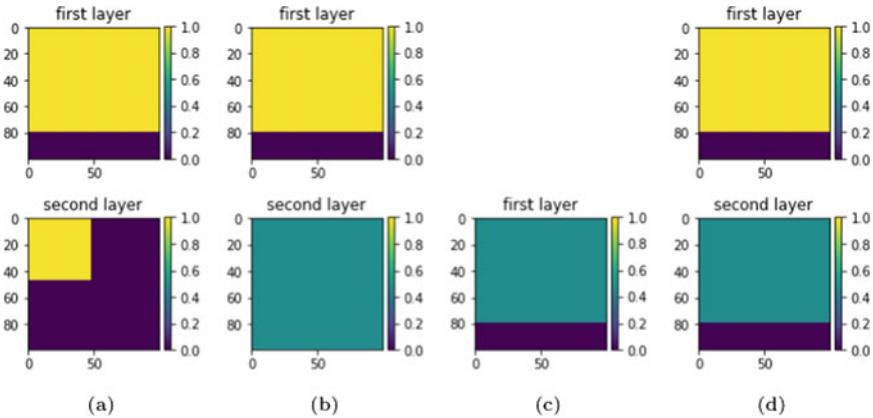


Fig. 3 Features represented as inputs for volume percentage of 80% and Poisson's ratio of 0.49

has Poisson's ratio information, modeled by filling entire layer with Poisson's ratio. Refer to Fig. 3b.

Input type 3 images consist of one layer, which has both volume percentage information and Poisson's ratio information, modeled by filling volume percentage of rows with Poisson's ratio completely and rest by zeros. Refer to Fig. 3c.

Input type 4 images consist of two layers. First layer has volume percentage information, modeled similar to Input type 1. Second layer is (100×100) size image, has Poisson's ratio information, modeled by filling volume percentage of rows with Poissons's ratio completely and rest by zeros. Refer to Fig. 3d.

2.5 Training the Network

The input and target datasets have total 931 images, in which 70% of data is taken as training set, 20% of data is taken as cross-validation set, and 10% of data is taken as testing set. Training the first network took 6 h for 2000 epochs, whereas second network took 12 h approximately.

3 Results and Discussion

To demonstrate the results after training the encoder-decoder network with input features given as input images, we have taken input and target images for volume of 45, 55, 65%, and Poisson's ration of 0.2. Figure 4 shows the results of first encoder-decoder network, and Fig. 5 shows the results of second encoder-decoder network for all 4 types of inputs. From the figures, it is clear that the second encoder-decoder

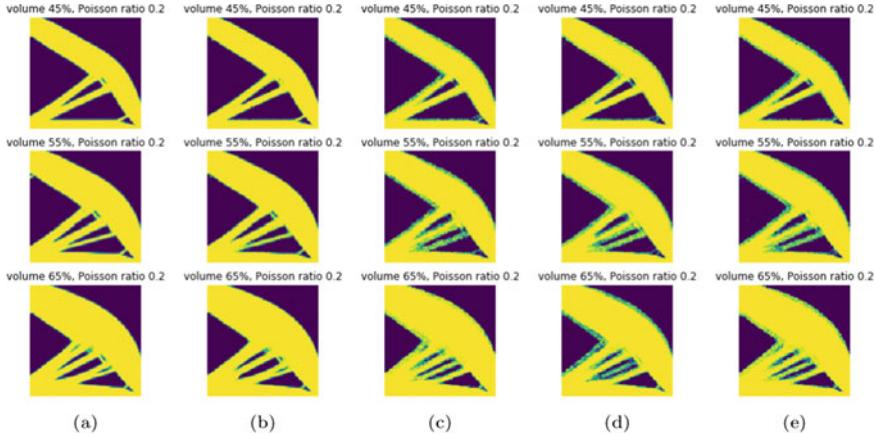


Fig. 4 Results of 1st encoder–decoder network for volume of 45, 55, 65%, and Poisson’s ration of 0.2

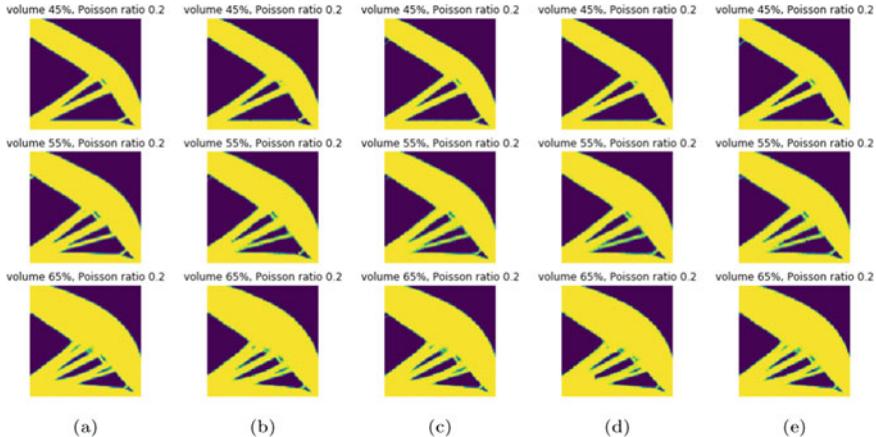


Fig. 5 Results of 2nd encoder–decoder network for volume of 45, 55, 65%, and Poisson’s ration of 0.2

network combination with input type 4 captures very fine details, which are necessary to consider for the continuity in volume of the original beam. The MSE of 100 input type 4 images tested on the second encoder–decoder network reaches 0.025, which is far below than rest of the input types.

4 Conclusion

In the present work, we have developed a encoder–decoder network, which takes input conditions, i.e., volume fraction and Poisson’s ratio as input images and optimized structures correspond to the same input conditions generated by MATLAB code as target images. While training takes 12 h, which is considerably huge, testing only takes seconds. Using MATLAB to generate one image takes 5 min, whereas using this proposed network takes only 10 s, which is 30 times faster than MATLAB code.

References

1. O. Sigmund, A 99 line topology optimization code written in matlab. *Struct. Multidisc. Optim.* **21**(2), 120–127 (2001)
2. T.E. Bruns, A reevaluation of the SIMP method with filtering and an alternative formulation for solidvoid topology optimization. *Struct. Multidisc. Optim.* **30**(6), 428–436 (2005)
3. I. Sosnovik, I. Oseledets, *Neural Networks for Topology Optimization*. arXiv preprint [arXiv:1709.09578](https://arxiv.org/abs/1709.09578) (2017)
4. S. Banga et al., *3D Topology Optimization using Convolutional Neural Networks*. arXiv preprint [arXiv:1808.07440](https://arxiv.org/abs/1808.07440) (2018)

Rank Aggregation Approach to Feature Selection for Improved Model Performance



Mohan Sangli and Anish Ravishankar

Abstract Feature selection is an integral part of machine learning. The focus on reducing complexity and time in processing data is more now than ever before. A direct and simple way to boost performance is to identify and remove the noise parameters, thereby keeping the features that positively contribute to model performance. The existence of different kinds of datasets to be worked on demands a method that is adaptive and efficient, and less prone to noise. One such solution is identifying parameters from multiple algorithms and then applying the robust rank aggregation method to filter out the parameters that are not important.

Keywords Importance scores · Feature selection · Model building · Random forest · Aggregation

1 Introduction

The advent of modern-day electronics, such as smaller transistors, sensors and memory devices, has enabled the collection and storage of data like never before. One need not be too concerned with evaluating the necessity of accounting for certain factors while collecting information; it seems like more is always better. However, when the data must be processed through machine learning algorithms or studied in statistics, having more features is not necessarily good. The existence of features that are degenerative, redundant, or even noisy could significantly hamper the performance and interpretability of models built for data containing such features. More so for situations where the number of examples or data points is only a few hundred whereas the number of variables could be a few thousand [1]. Dimensionality reduction is a regular practice in machine learning and rightfully so lesser

M. Sangli (✉) · A. Ravishankar
ICUBE Consulting Services India Pvt. Ltd, Bangalore, India
e-mail: msangli@icubecsi.com

A. Ravishankar
e-mail: aravishankar@icubecsi.com

dimensions(features) means that the models are likely to be trained faster, while lesser noise-inducing features in the dataset could mean lesser error in model performance. The industry has regularly used Random Forest (RF), eXtreme Gradient (XG) boosting and other such algorithms to extract variable(feature) importance. These have been proven to be quite successful as evident in [2] and [3]. Some practices also involve the use of linear regression (LR) model coefficients, as mentioned in [4]. The coefficients are obtained by iteratively solving the system of linear equations. By extension, the recursive feature elimination method is used to select features by recursively considering smaller sets of features. First, the estimator is trained on the initial set of features. Then, the least important features are pruned from the current set of features. That procedure is recursively repeated on the pruned set until the selection of the desired number of features is eventually reached.

The overarching issue with all these methods is that they are not one sure stop for all kinds of datasets. Some may perform better with datasets that are more horizontal and others with those that are vertical. Even within these categories of datasets, there are many factors, such as linearity of the relationship between independent variables and the response that factor into the unpredictable nature of the performance of these models. The more intuitive solution to this problem is to use multiple methods to find the importance of each feature and find the average of the importance across all methods or aggregate it using statistical methods like described in [5], [6] and [7]. While methods such as principal component analysis (PCA) have been regularly used in efforts of dimensionality reduction, these methods do not yield importance metrics or scores. Indicating the extent of influence independent variables have on a response variable would be useful in reevaluating data collection efforts.

Rank aggregation approach to feature selection consists of two steps: First is to select methods to score the importance of features and to generate rank lists, and the second is to aggregate these rank lists to generate the final rank for the features. To get importance scores, we make use of surrogate models. Each model generates a score for each feature, and when sorted in descending order of these scores gives the rank. In choosing the scoring methods, it is important to consider methods that use statistically different approaches, which encompass a wide variety of datasets, to calculate feature importance scores. It is also worthy to note that most ranking methods are likely to have some degree of consensus as most of them are based on variance measures in one form or the other.

The rest of the paper is organized as follows: Sect. 2 introduces the selected surrogate models used to generate rank lists, and Sect. 3 contains a discussion on the aggregation algorithm used. Section 4 illustrates the experimental setup, and the results are shown in Sect. 5.

2 Surrogate Models

Given a dataset with independent variables and response variable, a surrogate model builds a mathematical relationship between the two. From this relationship, we can derive feature importance scores. Models described in Sects. 2.1–2.3 are regularization-based models, 2.4 and 2.6 are decision tree-based models, and 2.7 is gradient boosting-based model. Let \mathbf{x} be the input vector such that $\mathbf{x}^{(i)} = (x_{i1}, x_{i2}, \dots, x_{ip})$ and p is the number of input dimensions.

2.1 Ridge Regression

Ridge regression is a type of regularization on the linear regression model where, in addition to the residual sum of squares, L_2 norm of the coefficients is also minimized. Here, β is the coefficient from the linear regression component. Larger the value of α , the more the coefficients get penalized [8].

$$\min_{\beta} \|\mathbf{x}^T \beta - y\|_2^2 + \alpha \|\beta\|_2^2 \quad (1)$$

2.2 Least Absolute Shrinkage and Selection Operator (LASSO)

Lasso uses L_1 norm of the coefficients to regularize [9].

$$\min_{\beta} \frac{1}{2n} \|\mathbf{x}^T \beta - y\|_2^2 + \alpha \|\beta\|_1 \quad (2)$$

2.3 Elastic Net

In Elastic Net, the regularization is performed with both L_1 and L_2 norm of the coefficients [10].

$$\min_{\beta} \frac{1}{2n} \|\mathbf{x}^T \beta - y\|_2^2 + \alpha \rho \|\beta\|_1 + \frac{\alpha(1-\rho)}{2} \|\beta\|_2^2 \quad (3)$$

2.4 Decision Tree

Decision tree algorithm is often referred to as classification and regression trees (CART). It makes use of recursive binary splitting and pruning to make the model performance better. The decision trees are used to fit a sine curve with addition noisy observation. As a result, it learns local linear regressions approximating the sine curve [11].

2.5 Random Forest (RF)

Random forest is an example of bagging-type ensemble technique. To reduce the variation of the prediction function, many independent sub-models are built and later combined by taking a weighted average. This technique is called bagging or bootstrap aggregation [12].

2.6 Extra Tree

It essentially consists of randomizing strongly both attribute and cut-point choice while splitting a tree node. In the extreme case, it builds totally randomized trees whose structures are independent of the output values of the learning sample. The strength of the randomization can be tuned to problem specifics by the appropriate choice of a parameter [13].

2.7 XG Boost

Tree boosting is an effective and widely used machine learning method. In this approach, the models are built sequentially instead of independently. XG Boost is a scalable end-to-end tree booting system [14].

3 Robust Rank Aggregation Method

From each of the models, we generate n ($n = 7$ in this case) distinct rank or preference lists. For the purpose of finding the rank aggregate, we use the robust rank aggregation method proposed by Kolde et al. in [7]. In most cases, we are interested in finding features that are ranked higher in many preference lists while ignoring non-informative list. These non-informative lists could be considered as noise in

the process. Given n preference (rank) lists, for each feature, the algorithm looks at where the feature is positioned in each of preference lists and compares it to the preference list that is generated by randomly shuffling the features. This is our null hypothesis rank list. For each feature, let the corresponding rank vector be $r = (r_1, r_2, r_3, \dots, r_n)$ where r_j is the rank of the feature in j th preference list. We find the binomial probability that $r_k < \hat{r}_k$ where \hat{r} is the rank vector of the feature generated by the null model. From this binomial probability, we find the minimum of p values and assign ranks accordingly. Minimum p value suggests a stronger case to reject the null hypothesis in this case.

To demonstrate that the ranking methods widely used in the industry and academia alike give different importance scores when used independently, we compare them against each other in Figs. 1 and 2. The values for scores are normalized to fit the same scale on Y-axis. This is allowed as the scores are relative to other features within the same ranking method. The X-axis is arranged in order of the result provided after applying the robust rank aggregation method. We can notice the behavior previously mentioned, i.e., each of the scoring algorithms gives different scores for the same features, but there seems to be a consensus among them. We can observe in Fig. 1 that the “Month” feature would have failed to qualify as its Random Forest score is lesser than many of the other features, but it falls within the cutoff when using

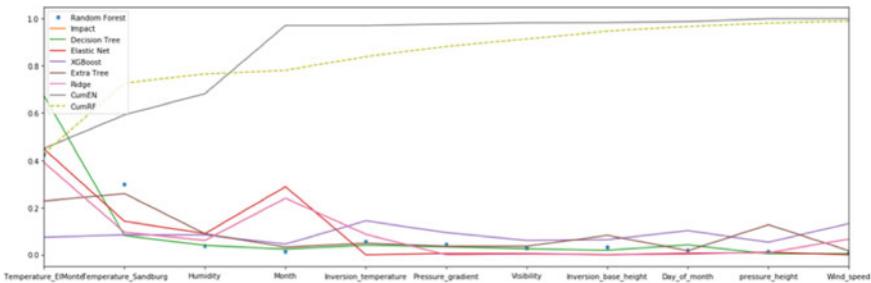


Fig. 1 Importance scores given by various models with cumulative RF and EN scores for Ozone dataset

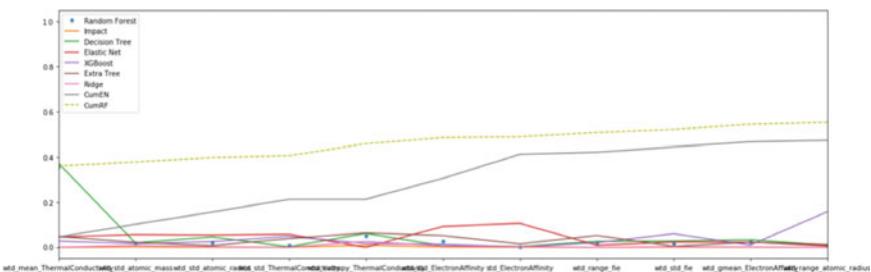


Fig. 2 Importance scores given by various models with cumulative RF and EN scores for Superconductivity dataset

rank aggregation method as three other models have found it to be important, by assigning higher score. Figure 2 depicts a similar case with another dataset. The feature “wtd_std_ThermalConductivity” that is ranked third by the rank aggregation method is scored low by Random Forest but has high scores from three other models. We use this graph to find the cutoff for the number of features to be used in model building later in Sect. 4.

4 Experiments

To corroborate the validity of the proposed method, we run tests with two popularly available datasets. The experiment is designed to check the selection of features by various methods—rank aggregation, random forest importance scores and coefficients of the linear regression model and compare the results.

First, “Ozone” data available in `m1bench` package in R is considered. After removing missing values, the dataset contains 12 features, 1 target (“ozone reading”) and 203 instances (of 366) out of which 150 are used for training the model. The dataset contains instruments’ reading for various factors that might influence the ozone levels (ozone readings) such has humidity, inversion temperature and pressure gradient. The second dataset considered is “Superconductivity,” which records at what temperature a conductor becomes a superconductor based on features such as entropy and electron affinity. The first 3000 records are considered with all 81 features and 1 target (critical temperature). The method used here is to build (train) a model that is frequently used—Elastic Net with all the features first and then compare it with the reduced set of features from rank aggregation, cumulative random forest scores and cumulative linear regression coefficient values. To sub-select the features for dimensionality reduction, we can consider taking least root mean squared error (RMSE) or out-of-bag (OOB) error [15] or taking 90% of cumulative importance score. To get a conservative evaluation, we use the 90% cumulative score methods as it gives fewer features. However, any sub-selection method can be used as long the number of features selected remains consistent when comparing. As the model that is used for training and evaluating is Elastic Net, we will use the cumulative of Elastic Net’s importance scores. For the Ozone dataset, it can be seen in Fig. 1 that the first four features make a 90% importance score. Similarly, for Superconductivity dataset, the cutoff is 40 features. Now, the first 40 features in the rank aggregation list, Random Forest importance scores list and linear regression coefficients list are compared.

The result parameters to compare are maximum train error (Train Max Err), maximum test error (Max Test Err), mean test error (Test Mean Err), root mean squared error (RMSE) and maximum absolute percentage error (MAPE). The parameters for the Elastic Net model are allotted at the end of 100 iterations of hyper-parameterization. Implementation is done using the scikit-learn Python package and an analytics cool ‘*Intuceo*’ [16].

Table 1 Error metrics for Elastic Net for Ozone dataset trained on first 150 samples and last 53 as test

Importance methods	Model	Train Max Err	Test Max Err	Test Mean Err	RMSE	MAPE
All features	EN	28.51	29.09	7.75	4.45	78.52
Rank aggregation	EN	27.42	27.48	7.18	4.13	71.9
RF	EN	30.56	31.99	9.44	5.38	105.41
LR	EN	29.95	27.91	12.59	6.46	139.21

Table 2 Error metrics for Elastic Net for Superconductivity dataset trained on first 2100 samples and last 900 as test

Importance methods	Model	Train Max Err	Test Max Err	Test Mean Err	RMSE	MAPE
All features	EN	32.54	31.54	11.99	30.16	7444.59
Rank aggregation	EN	36.46	33.24	7.88	22.06	2284.64
RF	EN	30.84	30.4	11.49	28.84	7200.79
LR	EN	33.21	32.09	12.14	30.53	7660.95

5 Results

5.1 Ozone Dataset

See Table 1.

5.2 Superconductivity Dataset

See Table 2.

6 Conclusion

As the results indicate, neither the random forest importance scores list nor the coefficients of the linear regression model perform as well as the rank aggregation method. The test mean errors and root mean squared errors are lesser in rank aggregation method than either of the other methods. The results obtained when all features are

used indicate higher errors. Comparatively, we are able to remove noisy features and reduce error. Notice that the rank aggregation methods outperform the model built on all features as well, which indicates the presence of noisy features in the dataset.

References

1. I. Guyon, A. Elisseeff, An introduction to variable and feature selection. *J. Mach. Learn. Res.* **3**(Mar), 1157–1182 (2003)
2. R. Díaz-Uriarte, S.A. De Andres, Gene selection and classification of microarray data using random forest. *BMC Bioinform.* **7**(1), 3 (2006)
3. B.H. Menze, et al., A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC Bioinform.* **10**(1), 213 (2009)
4. E.P. Xing, M.I. Jordan, R.M. Karp, Feature selection for high-dimensional genomic microarray data, in *ICML*, vol. 1 (2001)
5. O.L.M.A. Griffith, S.J.M. Wiseman, S.M. Jones, S.M. Wiseman, Meta-analysis and meta-review of thyroid cancer gene expression profiling studies identifies important diagnostic biomarkers. *J. Clin. Oncol.* **24**(31), 5043–5051 (2006)
6. S. Aerts, et al., Gene prioritization through genomic data fusion. *Nat. Biotechnol.* **24**(5), 537 (2006)
7. R. Kolde, et al., Robust rank aggregation for gene list integration and meta-analysis. *Bioinformatics* **28**(4), 573–580 (2012)
8. A.E. Hoerl, R.W. Kennard, Ridge regression: biased estimation for nonorthogonal problems. *Technometrics* **12**(1), 55–67 (1970)
9. C. Hans, Bayesian lasso regression. *Biometrika* **96**(4), 835–845 (2009)
10. J. Friedman, T. Hastie, R. Tibshirani, Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **33**(1), 1 (2010)
11. L. Breiman, et al., *Classification and Regression Trees* (Wadsworth, Belmont, CA, 1984), p. 432
12. P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees. *Mach. Learn.* **63**(1), 3–42 (2006)
13. A. Liaw, M. Wiener, Classification and regression by randomForest. *R News* **2**(3), 18–22 (2002)
14. T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, 2016)
15. N. Chehata, L. Guo, C. Mallet, Airborne lidar feature selection for urban classification using random forests. *Int. Arch. Photogram. Remote Sens. Spat. Inf. Sci.* **38**(Part 3), W8 (2009)
16. www.intuceo.com