EEE5109
MULTI-TARGET MULTI-SENSOR TRACKING AND DATA FUSION

1   PREREQUISITES

There are no formal prerequisites for this subject. The following undergraduate level topics are required to comprehend the contents of this course. It is expected of course attendees to perform all necessary self study in elements which are not well understood with regards to the topics listed below.

- ⚟ Signals and linear systems

- ⚟ Engineering probability, statistics and stochastic processes

- ⚟ Linear algebra

- ⚟ Programming proficiency (either C++ and/or Matlab).

2   COURSE FORMAT AND DATES

The course is given in a five day, intensive format, with lab sessions in the afternoons. The afternoon lab sessions will entail the completion of a *Class Assignment* throughout the week. The lecturer will be available for assistance and questions during these lab sessions. The class assignment will form the basic simulation platform for a further *one take-home assignments* which the course attendee has to complete in the weeks following the course. Both assignments will be combined in a semester mark, which will contribute 60% towards a final mark. The final exam will contribute the remaining 40% towards the final mark. Students may book appointments with the Course Convener and the Tutor.

The course Calendar is the governing document for planning: please monitor it frequently.

https://sites.google.com/site/radarmasters/schedule

Course interaction is via the UCT Vula System. You will have access to this information once you have registered for the course. It is important that you provide your preferred email address (one that it checked frequently) for your Vula registration.

3   STAFF

Convener:	Prof. M.R. Inggs, UCT
Telephone:	+27 (0) 21 650 2799
Fax:	+27 (0) 21 650 3465
Email:	mikings@gmail.com
Lecturer:	Dr Pieter de Villiers, B.Eng (UP), M.Eng (UP), PhD (Cantab), Pr. Eng (ECSA)
Telephone:	+27 (0) 12 420 4637 (UP) +27 (0) 12 814 2590 (CSIR)
Fax:	+27 (0) 12 362 5000
Email:	pieter.devilliers@up.ac.za (UP), jdvilliers1@csir.co.za (CSIR)
Contact time:	Only during the lecturing block week and during the pre-determined Skype session schedule. The preferred contact method is email.
Tutor:	Roaldje Nasjiasnagar, UCT, neddje@gmail.com

4   COURSE DESCRIPTION AND OVERVIEW

The course provides an overview of the most popular methods and techniques in the field of multiple target tracking and multi-sensor data fusion. It introduces engineers, scientists and military technicians to a toolbox of readily used key technologies that can be implemented with success in the field. The course is designed to cover fundamental topics in detail, whereas more advanced topics and extensions are dealt with in such a way that they remain easily accessible to attendees.

4.1   OVERVIEW

Data fusion in a multiple sensor environment has become of paramount importance in a world where information from several sources are readily combined to make inferences and decisions. In the engineering and computing contexts, the application areas of mining, manufacturing, robotics and defence have become reliant on data fusion methods. This course on multiple target tracking and data fusion methods aims to provide a foundation for research into more advanced topics in the field and to provide the course attendee with practical knowledge on how understand and apply the basic methods to a variety of challenging problems.

Data fusion methods combine the aspects of multi-target and multi-sensor detection, estimation, target tracking, target classification, situation assessment, impact assessment and resource management in order form a refined and coherent picture of the environment being sensed both in terms of object (target) states and their properties. The field of sensor fusion drawn from statistical and mathematical techniques such as detection, decision theory, association theory, sequential estimation and filtering, Bayesian inference, Dempster-Shafer evidential theory, machine learning, fuzzy logic, voting methods and random set theory to name but a few.

4.2   REQUIREMENTS

Even though for this module, an applied approach is taken, it is done so with the aim to lay the foundation for novel future research and development in both theory and applications. The course attendee will be required to grasp theoretical fundamentals from first principles, apply the theory to relevant problems as well as propose and research

possible improvements to the relevant methods and algorithms. In the case where such improvements are implemented, they should be evaluated and compared against classical solutions commonly found in the literature.

The course attendee will not be expected to know the material off by heart but must have a working understanding of the material and must be able to apply the knowledge to practical situations and problems. It must however be stressed that the course attendee is expected to have complete understanding of the theory and must be able to derive all mathematical aspects studied unless otherwise stated.

Course attendees will be evaluated on assignments and a written exam. Assignments will take the form of small projects in which algorithms and methods could be implemented and investigated to solve a particular problem. It must be stressed that tasks and projects assigned during the course of this module should be properly documented. Implementation notes, results and findings should be documented in a lab book during the implementation phase of the relevant project. These writings should be compiled and edited for recording in a proper report, a template of which will be supplied in an Appendix at the end of this document. Both the lab book and the report may be referred to for evaluation.

Course attendees are strongly encouraged to purchase the text prescribed textbook. The attendee will find this course very difficult, if not impossible without the textbook as a reference source. In the case where course material is sourced from journal papers, students should take necessary steps at the UCT library to gain access paper or electronic versions of the journal papers.

The study of substantial amounts of additional material is something that is particularly different from undergraduate courses and if one is not used to this it can appear daunting. Generally as you progress beyond a first degree the amount of background reading required increases. Due to the volume of material one clearly cannot study everything in detail. Learn to skim/speed read material to detect essential concepts. The ability to do so comes with practice. The goal is to expand understanding and to be aware of the contents of the material so that one can locate relevant information for a particular assignment/problem. Similarly, it is expected that for each assignment the additional references be studied to extract material relevant to the assignment. In summary, course attendees are expected to do their own filtering of material.

4.3   ASSIGNMENT METHODOLOGY AND RECOMMENDED APPROACH

Care must be taken to avoid spending time on less important aspects or trying to fix badly developed implementations. Most course attendees who reach postgraduate level have established an effective study and research methodology for themselves. There are however course attendees who have not yet developed such a methodology and it is recommended that they approach the study themes and assignments as follows:

1. Begin by creating a new report file using the assignment report template. Each course attendee should create such a template at the beginning of the course based on the report guidelines provided in Appendix A.

2. Scan through all of the designated chapters in the textbook associated with the study theme to obtain a global view of the theme.

3. Read through the entire assignment description to determine what aspects are considered important and what theory will be needed to complete the assignment. In many cases the assignments also include hints that can save considerable time and effort.

4. Study all of the material in the relevant textbooks and/or papers relating to the study theme in more detail. Particular emphasis should be placed on the aspects that are the focus of the assignment.

5. Whilst studying the material, begin writing the literature review section in the report that summarises the current body of knowledge on the subject. Start by consulting textbooks for the "classical" approaches to the problem and then proceed to the journal articles for the current state-of-the-art. Be sure to properly cite all of the relevant literature in your report.

6. Proceed by then writing the sections which entail the theoretical analysis (e.g., background and fundamental theory, possible approaches, motivations of assumptions and approximations, derivation of all required equations, etc.) and the algorithmic development (i.e. pseudo-code with final equations inline or references to derived equations). At this stage aspects of the material in the textbook may be unclear or the details of certain aspects may be missing.

7. The above three steps should be repeated for all theoretical "concepts" which relate to the task at hand. Do this by periodically referring back to the assignment and "checking all the boxes", until the theoretical analysis and algorithmic development have been completed and a clear understanding of both has been established. Remember to keep the objectives of the assignment in mind to help remain focused. At this stage the bulk of the study material in the textbook should have been mastered.

8. Where feasible, experiment with existing implementations in order to further increase understanding of the assignment topic and the expected outcomes.

9. Based on the algorithm utilised/developed, create an implementation and perform the necessary tests on suitable test data to verify that the implementation functions correctly. If possible, compare the implementation to existing implementations explored in the previous item. In the report provide a brief description of the implementation and the basic tests done to validate the implementation.

10. If applicable, study the dataset specified to be used to provide a comprehensive evaluation of the implementation. Provide an overview of the dataset in the report.

11. Develop an experimental plan for the comprehensive evaluation of the implementation, as appropriate. Document the experimental plan in the report taking care to motivate all aspects.

    Important note: When analysing and testing an implementation, try to separate and isolate the parameters of interest. The effects of these parameters should be evaluated one at a time, thereby reducing cross influences, and aiding in explaining the observed effects. Of course, there may be situations where parameters are dependent and cognisance should be taken of such possibilities.

12. Perform the experimental evaluation and note implementation aspects, results and findings in a lab book. Concisely organise and summarise a suitable summary of the results in the report, together with a discussion of the results.

13. At this stage the implementation and the body of the report has been completed. Perform any necessary editing to finalise the report body.

14. Now write the introduction and conclusion of the report based on everything that has been learned during the execution of the assignment. Make sure that the introduction explains the purpose of the assignment within the larger subject matter and that the conclusion highlights to what degree this has been achieved. Write the report abstract, which very succinctly describes what is attempted in the task. Note that the abstract is not an introduction/motivation, but simply describes what has been done.

To reiterate, the assumption is that the course attendee will **begin writing the report from the moment work begins on the assignment**. Only once the necessary derivations and pseudo-code has been documented in the report should actual implementation begin. Attempting to create an implementation without a correct algorithm as basis will invariably lead to an incorrect implementation. Unnecessary time will then be required to fix the implementation and there will most likely not be sufficient time to complete the assignment. The lab book, together with the report should therefore reflect the systematic investigation of the assignment problem. Similarly, before any experimentation and simulations are done, an experimental plan must be developed and documented in the report. Note however that many of the experiments can take considerable time to complete and careful planning is therefore required. It is also important to perform as much as possible of the experimental evaluation and results processing in parallel.

In the report, you should describe the algorithms implemented. The "algorithmic/algorithm" environment in LaTEX is ideal for this. Wherever you make use of an equation in your algorithm, ensure that there is a corresponding derivation and explanation of the equation. Note that this aspect is considered to be a crucial aspect of the learning process. Generally, the course attendee must fully derive, from first principles, all equations used in your implementation. The derivations must be given in the report. Failure to do so will result in reduced marks unless the attendee motivates fully why such derivation is not applicable or feasible.

5   LEARNING OUTCOMES:

Outcome 1:

Description: On successful completion of the module, the course attendee will be able to know and understand the key elements of multiple target tracking and data fusion.

Range: Key elements of data fusion are defined and standardised by the Joint Directors of Laboratories (JDL). This will serve as the basis of the scope and context of this outcome.

Outcome 2:

Description: On successful completion of this module, the course attendee will have knowledge of a wide range of tools capable of solving problems in different areas of data fusion (multiple target tracking, state and attribute fusion).

Range: The extent of these tools include methods for solving problems related to the different levels of data fusion as defined by the JDL.

Outcome 3:

On successful completion of this module, the course attendee will be able to identify which methods, algorithms and techniques are relevant to specific problems encountered in the field.

Range: Basic versions of most problems encountered in the fusion domain (JDL model) will be covered

Outcome 4:

On successful completion of this module, the course attendee will be able to apply methods, algorithms and techniques to solve specific problems encountered in the field of data fusion.

Range: Application of fusion algorithms to the basic versions of most problems encountered in the fusion domain (JDL model) will be covered.

6   TEXTBOOK AND SUPPORTING TEXTS

The following textbook is prescribed for the course:

[1] Design and Analysis of Modern Tracking Systems, Samuel Blackman and Robert Popoli, Artech House, ISBN: 978-1580530064.

The following books are recommended for additional study material:

[2] Sensor and Data Fusion: A Tool for Information Assessment and Decision Making, Lawrence A. Klein, SPIE Press, ISBN: 978-0819454355.

[3] Estimation with Applications to Tracking and Navigation, Yaakov Bar-Shalom, Wiley Interscience, ISBN: 978-0471416555. Available as e-book at the university library: link to e-book.

[4] Handbook of Multisensor Data Fusion: Theory and Practice, Second Edition, CRC Press, ISBN: 978-1420053081.

The following books provide some of the basic theory upon which topics in information and data fusion are based

[5] Pattern Recognition and Machine Learning (Information Science and Statistics), Christopher M. Bishop, ISBN: 978-0387310732.

[6] Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory, Steven M. Kay, Prentice Hall, ISBN: 978-0133457117

[7] Bayesian Signal Processing: Classical, Modern and Particle Filtering Methods, James V. Candy, Wiley, ISBN: 978047018094-5.

[8] Artificial Intelligence (A Modern Approach), Stuart Russel and Peter Norvig, ISBN: 978-0136042594.

Table 1: EEE5109F Multi-target Multi-sensor Tracking and Data Fusion 2011 Programme (topics expanded below)

Time	Mon 5/9	Tues 6/9	Wed 7/9	Thurs 8/9	Fri 9/9
08h30	Overview	Overview	S2	Overview	Overview
09h00	S1	S2	S2	S3	S4
10h00	S1	S2	S2	S3	S4
11h00	Tea	Tea	Tea	Tea	Tea
11h30	S1	S2	S2	S3	S4
12h30	Lunch	Lunch	Lunch	Lunch	Lunch
13h30	S1	S2	Break	S3	LAB
14h30	LAB	LAB	Break	LAB	LAB
15h30	LAB	LAB	Break	LAB	Conclude
16h30	Tea	Tea	Break	Tea	
17h00	LAB	LAB	Break	LAB	
18h00	Close	Close	Break	Close	

Table 2: Descriptions of Topics

Code	Topics
S1	Single Target Tracking
S2	Multiple Target Tracking and Data Association
S3	Attribute Fusion
S4	JDL Model of Data Fusion
LAB	Afternoon MATLAB laboratory sessions (Class Assignment)

8 SEMINARS

Course attendees will be provided with at least 5 seminar opportunities of about an hour each with the lecturer, convener and tutors. In the case of this module, the seminars will be optional, and will be an opportunity for course attendees to ask questions regarding the take-home assignments. The seminars will be carried out with access by Skype for students off campus after the lecture session. For bandwidth reasons, the number of parallel sessions will have to be limited. For example, all students resident in the same city will be expected to attend at a common venue, and students will have to organise their own venue and projection facilities. Screen sharing will be enabled, but it is unlikely that video will be supported, again due to bandwidth limitations. Within reason, and with prior arrangement, students can approach the tutor / and / or the lecturer for help with problem sets.

9 COURSE ASSESSMENT AND COURSE LOAD

9.1 GRADING POLICY

The time-load of a 20-credit module is taken to be a nominal 200 hours (for the average course attendee), excluding revision time. The final mark for this course will consist of a semester mark (60%) and an examination mark (40%). The semester mark is based on evaluation during the semester in the form of assignments, and the examination mark is based on a written examination. The examination is closed book, i.e. no notes may be brought into the examination venue. Students may be expected to perform mathematical derivations or problems similar to those covered in the lectures. Students may write the examination in their home location, provided satisfactory supervision of the examination can be arranged in good time.

Assignments are due to be submitted before 12h00 on the specified due date (See Section 10.3).

The semester and final marks will be constituted as follows:

	Effective Hours	% of semester mark	% of final mark
Assimilation	70	-	-
Contact time	40	-	-
Assignment 1	45	50	30
Assignment 2	45	50	30
Exam	-	-	40
TOTAL:	200	100	100

9.2 LATE ASSIGNMENT POLICY

Recognizing that course attendees may face unusual circumstances, each course attendee will have a total of seven free late (calendar) days to use as he/she sees fit. Once these late days are exhausted, any assignment submitted late will be penalized 20% per day late (that is an actual 20% penalty, not 20% of mark obtained). However, no assignment will be accepted more than four days after its due date. One full "late day" is used for each 24 hours or part thereof that an assignment is late. The date after which no submissions for an assignment are accepted is referred to as the cut-off deadline.

10   ASSIGNMENTS

A number of assignments are due for this course. Assignments will be made available electronically, either on the module web page or by email. The assignment handout contains references to supplementary material which will in some cases need to be downloaded by the course attendee. The specifications for the assignment related to a specific section of work will be included at end of each assignment document.

Programming projects must be implemented in Octave/Matlab and/or C/C++ unless otherwise arranged. Java is generally not recommended as the garbage collector tends to cause significant performance issues with these data intensive applications unless steps are taken to tune the Java VM.

Some of the assignments may require considerable computational time to execute. Additionally a large number of repetitions are typically required to produce statistically valid results.

The following is a list of assignment topics. The due dates for the assignments are included in the schedule ("Schedule" section).

•   ***Class Assignment:*** Basic tracking simulation platform.

•   ***Assignment 2:*** One of two extensions to the class assignment

   • Multiple model tracking

   • Advanced data association (JPDA or MHT)

10.1   SUBMISSION OF ASSIGNMENTS

Submission of assignments occurs by placing them in the relevant course attendee's dropbox on the Vula module website. The files names for the submitted files should be of the following format:

A<X>_<Surname>_<Initials>_report.pdf

A<X>_<Surname>_<Initials>_code.zip

where <X> is the assignment number, <Surname> is your surname and <Initials> is your initials. If you have a multi-word surname, please capitalise (proper case) each word and concatenate the result (e.g., the report of assignment 1 for "Mr P van der Merwe" should be saved in a file with name similar to "A1_VanDerMerwe_P_report.pdf").

A plain text file "code/README" should also be included which contains instructions on the implementation should be built or necessary files generated. The file should also include instructions on how to reproduce the results as described in the report.

10.2   EVALUATION OF ASSIGNMENTS AND FEEDBACK

After the cut-off deadline for an assignment, the marking process of the assignment begins. Once an assignment has been marked, an evaluation of the assignment will be made available which will address any issues encountered. The goal is that the evaluation of assignments be completed approximate 1.5 weeks after the final submission has been received. Unfortunately due to the lecturer's high work-load it may not be possible to achieve this target.

The format of evaluation feedback shall probably be in the form of a mark sheet that will be emailed to course attendees.

Unfortunately, the time available for evaluations is restricted and there is a limit to the depth in which evaluation can realistic be done, and hence the extent of annotations. Therefore, see the annotations as hints at the most significant issues in the submission that could be identified in a limited evaluation time-frame.

Depending on the quality and consistency of the report, assignments may be evaluated by attempting the execution of the code submitted. Code that does not work will result in penalties in the form of reduced marks. Note that if, during the evaluation, there is uncertainty with regards to the course attendee's assignment (report or code), he/she may be asked to demonstrate/explain his/her code/report during one of the seminar sessions.

10.3   DEADLINES AND IMPORTANT DATES

Date	Description
23 September	Class Assignment due 4pm Seminar 1
26 September	Assignment 2 available
30 September	4pm Seminar 2
7 October	4pm Seminar 3
14 October	Assignment 2 due 4pm Seminar 4
21 October	4pm Seminar 5 (Final Q&A)
31 October	3pm **Written Exam**

APPENDIX A - ASSIGNMENT GUIDELINES

The guidelines in this section serve a general template for technical report writing for this and other similar courses.

A.1 GENERAL

The two main deliverables for assignments are a typed report (in PDF format) and source code (MATLAB, C, etc. and related support files). The report should be of an acceptable standard. In the case of a compiled implementation language, please do not include compiled binaries or executables. They will be build during evaluation and hence you must included all components required to build your implementation. In general, do not include any file that is generated apart from the report PDF.

For some assignments datasets are to be processed into intermediate files, typically by means of scripts. Remember to include all scripts to generate the needed files from source or original files, which are assumed to be in the current directory (symbolic links will be created as needed).

Under Linux, most tools already include the ability to generate PDF output. For Windows users you may need to install appropriate software. CuteWriter is recommended. In order to use CuteWriter, the Converter is also required and these can be obtained here:

• Home page: http://www.cutepdf.com/products/cutepdf/Writer.asp

CuteWriter and similar software act like a printer driver with the difference being that the result is written to a PDF file. You therefore simply print the document and select CuteWriter as the target printer.

A.2 LATEX

For students who are intending to continue with postgraduate studies (i.e. Masters), it is highly recommended (in fact, it is required) that the reports be written using LATEX. LATEX produces documents of a much higher standard than typical word processors, in particular with regards to mathematical typesetting. Combined with the powerful table of contents and index generation system, as well as the automated bibliography management, alternatives are downright painful in comparison. Once a nominal proficiency with LATEX has been attained, the time required to complete the reports will typically be less than if they were done in, for example, Word. Another advantage of LATEX is that, due to its text based nature, one can create scripts to automatically generate content such as tables and plots directly from an implementation.

For an overview of the advantages of LATEX, have a look at the The Beauty of LaTeX. Another aspect is the apparent non-deterministic behaviour of MicrosoftWord. LATEX documents written decades ago produce *exactly* the same output when processed with the latest version of LATEX. This level of consistency is unheard of with Microsoft Word. On complex documents Microsoft Word tends to get confused at some point, and the result is unwanted effects in the document. It usually requires the document to be reassembled in order to correct these types of problems.

The following book will greatly assist the use of LATEX and is a good investment.

• Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle and Chris Rowley, The LATEX Companion, Pearson Education Inc., 2nd edition, 2004, ISBN: 0201362996.

Various other materials relating to LATEX can be found in the "latex" subdirectory of the class AFS directory.

A.3 MARK ALLOCATION

The marks allocated for each assignment will be proportioned approximately as follows:

• Content: 90%

• Presentation 10%

By presentation, the following is implied: layout, logic flow, consistency, use of language, spelling, punctuation, referencing, etc. The composition of the mark for the report content components will be supplied with each assignment. As a general guideline, each report component will be evaluated based on a template mark out of 5 as follows:

• **0 – Severely Deficient** (essentially no attempt made)

There was nothing useful to evaluate and no marks could therefore be awarded.

• **1 – Poor** (attempt made but mostly incorrect)

Methodology, answers, results and/or implementations are mostly incorrect, superficial and have critical errors or problems. The answer shows little or no understanding and depth of the material. Presentation has fundamental problems in terms of structure, number of errors and technical quality.

• **2 – Fair** (attempt made but contains many errors, minimal level of description)

Methodology, answers, results and/or implementations are basically correct but have major errors or problems, or have limited depth. The answer shows a basic understanding of the material but limited depth. Presentation meets minimum standards but has significant problems in terms of structure, errors and technical quality.

• **3 – Average** (basic attempt with some errors, basic level of description)

Methodology, answers, results and/or implementations are mainly correct but have moderate errors and detail, or are not optimal. The answer shows moderate depth and understanding of the material. Presentation is generally good but has minor problems in terms of structure, errors and technical quality.

• **4 – Good** (complete and correct attempt together with sufficient level of description)

Methodology, answers, results and/or implementations are mainly correct and detailed but have minor errors, or are not optimal. The answer shows a very good depth and understanding of the material. Presentation is mostly well-structured and error free, and of good technical quality.

• **5 – Excellent** (comprehensive and detailed submission)

Answers and results are correct, detailed and near-optimal, as is the working and methodology used to obtain them. The answer shows an excellent, deep and synthesised understanding of the material. A degree of critical analysis is evident and alternatives have been explored. Additional research has been done (as appropriate) and references have been provided. The implementations are flexible and modular facilitating application on new problems. The material is well-structured, error free and of high technical quality.

The mark is estimated based on the same considerations applied in 4'th year project and MEng dissertation evaluations. As certain components are more complex than others, the total mark for the component may be a scaled version of the template mark. Components that contain notable errors will receive a template mark below 3 and as low as 1 depending on the severity of the errors.

A basic correct assignment, but with minimal motivation and methodology, will most likely obtain a mark of approximately 60 - 65%. If the assignment is complete and professional, and a degree of critical analysis is apparent (which indicates deeper understanding), a mark in the region of 80 - 85% can be obtained. Typically this will include a discussion of problems encountered, an analysis of potential solutions and how these problems were eventually solved. To obtain a mark higher that 85%, significant effort and/or understanding outside the exact specification will be required. This will typically also be associated with extra features and functionality in the implementation, as well as additional references indicating background research. These comments describe the estimated effect of what is a composite marking process. The evaluation process has been so designed that a mark below 70% provides an indication that the student is not ready for further postgraduate studies and should therefore reconsider attempting further studies.

A.4 EVALUATION METHODOLOGY

The first aspect that is checked is the declaration of originality as the University considers plagiarism a very serious issue. Failure to include an acceptable statement will most likely result in the submission being ignored. A zero mark will be awarded in such cases.

The assignments are evaluated at the component level in order to ensure unbiased assessment. A particular component is assessed for all students, before proceeding with a subsequent component. This approach, amongst other, allows the calibration of the evaluation procedure and allows the detection of interpretation problems with the assignment specifications. Due to the component level evaluation, students are requested to follow the structure indicated in the mark break-down in order to simplify the marking process.

The "presentation" related components are typically assessed first, after which aspects such as layout, language and spelling were ignored for the "content" related components. With regards to the standard of the assignment and evaluation: the semester assignments are representative in nature as compared to what a student can expect the exam assignment to entail. It has been stated before: in cases where inconsistency is apparent in a report, submitted implementations are evaluated by studying the code and attempting to compile/synthesize and execute/simulate. In many cases the code is modified to test aspects and to display the results of certain operations. If the code is difficult to study and / or execute, the attained mark will invariably be low.

A.5 TECHNICAL REPORT WRITING AND REQUIRED LAYOUT

Before any type of writing commences it is essential to define the target audience. The nature and level of education of the audience determines the optimal approach to be followed. For the reports, the target audience is assumed to be the module lecturer and the external examiner. The assumption is therefore that the target audience is already very well informed regarding the subject matter. The target audience therefore will not usually read the report in order to extract knowledge. Instead, the readers will look for evidence that the student has understood the material, can execute a systematic scientific / engineering study and can properly present the results. The report should not only convey information clearly and coherently, but should also explicitly describe how and why the investigation was performed. The experimental methodology is of particular importance.

The required format for the reports is approximately as follows:

1. **Title Page:** The essential information that must be present on the title page include: the university name, the faculty, the department, the course code and name, your name and student number, the number and

title of the assignment, and the date of submission. The title of a report should correspond to the designated title as given in the corresponding assignment specification. The title page of the study guide gives an example of the desired format.

2. **Declaration of Originality:** See Appendix B.

3. **Contents:** The report must include a table of contents, with up to two levels of detail.

4. **Abstract:** The report must contain an abstract which summarises what has been done in the report and what the main results and conclusions were

5. **Introduction:** The introduction of a technical report identifies the subject, the purpose/ objectives/goals, and the plan of development of the report as well as places the report in the context of the current body of knowledge. The subject is the "what", the purpose is the "why", and the plan is the "how". The aforementioned elements should be addressed in about a half a page. The introduction should include a literature review (one to one-and-a-half pages) which details relevant existing methods and approaches on how to solve the problem (both classical and state of the art). This places the work attempted in the report in relation to the current body of knowledge. Together these acquaint the reader with the context and overall execution of the assignment. The introduction is therefore evaluated based on how effectively this is achieved. The introduction/objectives should be 1 to 2 pages.

6. **Method:** This section details the approach followed in the solving of the problem addressed by the report. It should stepwise proceed through all relevant equations and derivations contained in the method/algorithm.

7. **Results:** This section should detail the experimental setup as well as how meaningful comparisons and conclusions can be made. Furthermore, the results of the relevant experiments should be presented, discussed and interpreted (what do the results mean?)

8. **Conclusions:** The conclusion should summarise the main findings and interpretations of the results.

9. **References:** A list of references in IEEE Transactions format is compulsory (see Presentation section).

10. **Appendix:** These contain material that is too detailed to include in the main report, such as detailed data of experimental analysis, if appropriate.

A.6 CONTENT

In the report you must show clearly that you understand:

• the objectives/goals of the assignment

• the relation of the work in the assignment to the current body of knowledge

• the necessary theory (with particular emphasis on the mathematical aspects)

• how to apply the theory to create an implementation

• how to perform empirical and/or simulation based evaluation of the implementation

• the interpretation of the results

If the report does not contain a sufficient level of detail, written in your own words, the items above cannot be evaluated and the result will be a lower mark. For the evaluation of the implementation, you must included sufficient, step-by-step information on how to simulate/execute your implementation. If you did not do so, or your implementation cannot be executed, it is assumed that your results cannot be

reproduced and are considered invalid. Execution of the implementation is attempted on the cluster nodes. If the execution fails, the results are also considered invalid.

A.7 PRESENTATION

The following lists items which are considered to fall within the scope of the presentation. Most of the conventions listed follow the IEEE stylistic guidelines for authors. Please adhere strictly to the language, formatting and layout specifications for the reports. Referencing is particularly important.

1. **Consistency:** For aspects not specifically mentioned in the rest of this list the guiding principle should be readability, which is greatly influenced by consistency. If a particular style is chosen for an aspect, this style must be applied consistently throughout the entire document.

2. **Language:** The report must be compiled in English using formal / technical language.

3. **Spelling:** All relevant modern document generation environments support spell checking. Spelling errors are therefore entirely unacceptable. Use British English, not American.

4. **Sentences:** Use complete and short sentences. Try to keep sentences below a line and a half long.

5. **Layout:** Use single column format for document pages. Ensure that paragraph (full) justification is applied. Do not indent the first line of a paragraph. Avoid half-page and single sentence paragraphs. When using LATEX please use the "a4wide" or "geometry" packages to reduce margin whitespace.

6. **Font:** Use 11 point font and 1.5 line spacing. The text body font should be Times New Roman or equivalent. Matrices should be typeset in bold uppercase letters and vectors in bold lowercase letters.

7. **Sections:** Do not underline section titles. Headings should be left justified. Avoid excessive whitespace, do not start each section on a new page.

8. **Page Numbering:** Include the page number in lower right corner.

9. **Symbols and Mathematical Equations:** Mathematical symbol letters (both upper case and lower case) should always be italics (this is automatically done in LATEX). Use the simplest symbology that is allowed by the number of symbols used and that is generally accepted/used in the relevant literature. Properly punctuate mathematical equations embedded in the text (presume that the mathematics are part of the text). Properly number equations that are referred to in the text (this is elegantly handled in LATEX) . Usually, bold font symbols indicated vector-matrix notation.

10. **Figures:** Should always have a caption below the image. Figures should be horizontally centred, whilst the caption should be left aligned relative to the figures. Captions should be punctuated correctly. If the figures are derived from a reference, the caption should include the appropriate citation. If at all possible, never place a figure before the first reference to it in the text (for LATEX, use the [htbp] option to the figure environment to set placement priorities). All figures must be referenced and citations to figures in text always carry the abbreviation "Fig." followed by the figure number. The abbreviation is used even when it begins a sentence. Figures should fit on one page. Do not use frames around figures. In the case of a block / flow diagram, make the borders of functional units thicker than the flow lines used to connect the blocks to improve readability. Light shading of the functional units is also recommended. Use vector formats and avoid raster images. Raster formats based on lossy compression (such as JPEG) should avoided at all costs unless the figure is a picture.

11. **Plots:** Most of the comments for figures apply. Plots should always have a title and axis labels.

12. **Tables:** Should be horizontally centred and should always have a caption. The column and/or row titles should be in bold. Large tables with detailed results should be placed in an appendix, with a corresponding summary table in the main text. Tables in the body of the report should fit on one page.

13. **Acronyms:** Define acronyms the first time they appear. The definition should be written out as part of the sentence, followed by the acronym in parentheses.

14. **Algorithms / Source Code:** Since source code is to be submitted, do not include the source code into your report. Rather give the algorithmic pseudo code in your report and indicated in which source file the algorithm is implemented.

15. **References:** A list of references in IEEE Transactions format is compulsory (however strict adherence to the IEEE publication abbreviation conventions is not required). If you use an algorithm, quote, technique or theory someone else has developed, you must included a reference and attribute the item to them. Take care to reproduce the title of the reference exactly. The capitalisation of the original title should be retained. Reference the original/authoritative publication. Web/online references should only be used as a last resort and Wikipedia should never be referenced.

16. **Footnotes:** Technical documents should avoid footnotes as they disrupt the flow, although they are sometimes appropriate as a quick explanation.

A.8 IMPLEMENTATIONS

Although the postgraduate modules are not software engineering modules, a certain level of quality and format consistency is required in implementations. If applicable, develop reusable code by creating generic functions and modular code. Ensure readability by choosing and using a source code style. Additional requirements include:

1. **Functional description:** Each functional grouping of code should have a block description which describes in words the function of the code group / block.

2. **Block comments:** Commenting each line is generally not useful, the code itself should be written to be readable. It is however advisable to comment a block of code which performs a specific operation. This comment would typically describe aspects of the algorithm implemented and highlight non-obvious assumptions, limitations, side-effects, etc. Remember that comments are not for describing what the code technically does (that is what the code is for), comments are for what the code is intended to do. Try and comment the decisions you made when developing the code, specifically why you took the approach you did and why you didn't use other options. Don't assume that, even six months from now, you're going to remember why you did things a certain way. And the corollary: Don't assume you're going to be the one modifying the code a year or two from now.

3. **Alignment:** Standard practice is to use spaces instead of tabs for alignment. Be sure to set your editor to use 2 or 4 spaces for indentation.

4. **Page width:** Aim for approximately 80 characters per line, no more than 100 per line. Be careful not to over design. Based on an estimate of the division of marks, plan the execution of the assignments as not to spend disproportionate amounts of time on an aspect that may only count a small amount of marks.