Concurrent Practice of Graphical Representation of Assembly Sequencing and Tolerance Analysis

(University of Birmingham):1–9

Jonathan Jones[1]

Abstract

It is well known that the manufacturability of a mechanical design has significant effects of the time and cost of bringing a product to market. Three of the most pertinent factors within product development are tolerance design, design for assembly and, with the advent of automated manufacturing, assembly sequencing. This paper seeks to investigate how the above factors can be considered simultaneously in one user-friendly and time effective process that both expedites the process of assembly sequencing and calculates probabilities of mating-failures between parts. With the advent of CAD software and CAE, there is increasing call for a grand unified package for integrated design and tolerance analysis. The procedure for analysis within the software, referred to in this paper as ”GRAS Workspace”, involves the user entering necessary features of each part, while the system checks the data is acceptable, the cost analysis and fitting and feeding ratios are given and graphics to represent to each part generated which may be placed and connected to represent the assembly graphically. The connections are used to identify functional relationships between parts and sub-assemblies and generate interference matrices to be used for tolerance analysis and as an input for heuristic sequencing algorithms. In its current state, software requires further development and optimisation but has been developed in part on the proprietary language MatLab™ and is documented within this paper.

Keywords

Design for assembly, tolerance analysis, assembly sequencing, computer aided tolerancing.

1. Introduction

This introduction seeks first to highlight the importance of each factor in isolation followed by identification of current commercial and academic projects that link tolerance analysis with graphical representation and, in so doing, verify this projects claim to originality in its expansion of concurrent engineering.

Tolerancing techniques have evolved from being simply a means to reliably achieve functional assemblies to a process that enables mass production to determine its defects per million opportunities (DPMO) governed by consumer expectation, cost and the chosen statistical standard. most commonly 3σ or 6σ[1]. Genichi Taguchi developed principles of robust design throughout the 1950s using descriptive statistics which defined quality as the loss to society as a result of functional variation[2]. Today functional variation is commonly measured using the Taguichi loss function and controlled by adopting Motorola 6σ techniques[3] with which the DPMO is reduced to as little as 0.002. GD&T, General Dimensioning and tolerancing, is frequently mentioned to in literature regarding non-specific dimensions on individual parts whereas in modern computational tolerance, functional dimensioning and tolerancing or FD&T is of greater interest because most dimensions on each part are of no consequence overall assembly functionality and incur unnecessary computational cost to process[4].

Graphical Representation of assembly sequencing (GRAS) has been practiced in several forms and is a useful visualisation tool in designing for automated assembly so is therefore a significant area for cost saving given that as much as 50% of the total manufacturing cost is consumed by the process of assembly[?]. GRAS should be fast to implement and extract information from, capable of displaying fastening methods as well as other attributes of parts. An example of benefits of GRAS usage is the potential for use in heuristic sequencing optimisation algorithms. Most heuristic optimisation algorithms require a numerical representation of part interference, often in the form of a marix, to explain all possible sequences that can be acquired by computational interpretation and is discussed in greater detail in thw work conducted by Xiaowen Song et al.[5]. The AND/OR method was developed by Homem de Mello and Sanderson, it involves a root node which represents the whole assembly, each subsequent line represents a possible part to be removed[6]. This method suitable for small assemblies however drawing all permutations of disassembly for an assembly with more than ten parts is unfeasibly time consuming. The AND/OR method is also losing potency given the growing use of heuristic algorithms and other methods for assembly sequencing optimisation which is far more useful for both assembly and disassembly applications even for large assemblies[6]. Conversely research at the University of Birmingham began representing assemblies graphically as a tree representing the hierarchy between parts within the assembly the aim of identifying dimensional

[1]School of Engineering, University of Birmingham, B15 2TT
Email: JAJ515@student.bham.ac.uk

interferences within sub assemblies. This method proves a useful starting point for the implementation of tolerance analysis and visually requires little more information to build than is provided by existing methods of DFA. GRAS has already been successfully combined with geometric variation data from real-life assembly processes to analyse the variability in the assembly of compliant components within the wing box assembly at Boeing[7].

DFA is one of two key parts of DFMA (Design for manufacture and assembly), the other being DFM, the principal philosophy is to reduce complexity of the assembly and reduce the overall production time and cost. This is achieved by objectively ranking geometric, topological and mating features of parts to determine not only their assemblability but whether they are required at all in an assembly[8]. Three notable DFA methodologies are Lucas, Boothroyd-Dewhurst and Hitachi AEM (alternative assembly methodology). The Boothryod Dewhurst method requires the timing of each of the insertion motions which, for effective use, requires data from manufacturing laboratories[9] (Boothroyd) whereas the Hitachi AEM Method and Lucas methods require only part features and motions of assembly[10].

Design for assembly and graphical representation - or as it is described in the Ullman Product Evaluation process graphical modelling - are intrinsically linked according to Ulmans model which affirms the value of concurrent practice[11].

A Sweedish engineer, Carlson found that reasons for under-use of DFA in industry was because engineers were not aware of the method, did not consider it time effective to learn and the economic benefits were unproven[12]. If linked with tolerance analysis the economic benefits become obvious and well-researched and summarised no better than by the following Taguchi loss function Shown in equation 1[13]. Where product loss of N products, L is measured as a function of some measurable physical parameter y against time t.

$$L(y) = \frac{1}{N} \sum_{i=1}^{\infty} \int_0^T L_i(t, y) \tag{1}$$

Tolerance analysis has been effectively combined with graphical assembly representation in vector chain representation. In closed-loop vector chain (or vector loop) a vector represents a functional, kinematic linkage between two parts which can be expressed as an equation and either shortened or lengthened to conduct tolerance stack-up. While this method has merit, it required comprehensive dimensioning therefore integration of such a system is unsuitable for this project[14].

The Monte Carlo simulation is a statistical process that simulates statistical distributions and is often applied to tolerance analysis amongst other scientific and engineering applications. In this case it can be used to virtualise the geometric variation of manufactured parts to resemble that of a manufacturing process. This simulation involves generating random variables to which descriptive statistics can be applied to indicate whether a design, material or process should be altered. The output from such a process can provide accurate results and consider long term manufacturing responses brought about by factors such as changing atmospheric conditions and tool wear[1].

Crystal Ball is a spreadsheet-based statistical analysis tool owned by Oracle which has been widely used to compute the Monte Carlo simulation to analyse tolerance accumulation in assemblies. Statistical distributions including normal, triangular and Poisson as well as the mean and standard deviation can be selected to calculate the probabilities of yielding the mean assembly gap, from which other instances can be predicted.

There is extensive literature describing varying methods of assembly representation, methods of tolerance analysis as well as that of DFA in its various forms. Many commercial and academic entities have pursued a seamless linkage between tolerance analysis with graphical representation of assemblies in a process that is often described as computer aided tolerancing (CAT). Brigham Young University are developing software to interpret CAD models to perform vector chain analysis by applying statistical tolerancing techniques in a package now known as CATS-1D XL.[15]. Similarly, the University of Twente developed a prototype for another example of computer aided tolerancing known as FROOM (Feature and Relation based Object Oriented Modelling)[16] which is based on the TTRS (Technologically and Topologically Related Surfaces) tolerance model[17]. In the 1980s, Oyvind Bjorke had developed TOLTECH at the university of Norway[18].

Commercial packages for CAD integrated tolerance analysis include 3DCS developed by DCS, CETOL by Sigmetrix and with VSA as well as Technolmatrix now owned by Siemens PLM Software. Others include Analytix by Saltire Systems. An example of DFMA automation software is Boothroyd Dewhurst inc. who developed commercial software that provides a question and answer based interface to rank the assembleability of a part but provides no relationship between that and sequencing nor tolerance stack up[4].

As customer expectation heightens and methods within the product evaluation process become increasingly sophisticated, engineers are forced to concurrently practice a wider range of tasks along the product engineering process increasing thus the demand for a more complete concurrent model, the possibility and implementation of which will be evaluated in this report.

2. Tolerance Analysis

In the interest of leveraging GRAS for computer aided tolerance analysis, it is important to consider a range of tolerancing techniques. The relationship between design tolerance and process capability is a determining factor of the manufacturability of a product. Within tolerance optimisation there is a fundamental trade-off, the optimisation of manufacturing process performance and that of product design performance. If the manufacturing tolerance range falls outside of the capabilities of the design tolerance the greater the likelihood of quality loss. The manufacturing process capability index is widely used for statistical tolerance analysis and is given by equation 2[1].

$$C_p = \frac{USL - LSL}{6\sigma} \tag{2}$$

Where USL is the upper specification limit and LSL the lower. 6σ refers to 6 times the standard deviation of the part variability distribution as defined by manufacturing process which equates to a loss of 0.002 defects per million opportunities in otherwise ideal conditions. C_p is a measure of the ability of a process to deliver a part to the required quality level and will be referred to later in this study, 3σ quality is said to be achieved when $C_p = 1$ and, by extension, 6σ quality is achieved when $C_p = 2$.

2.1 Stack-Up

Tolerance stack-up analysis is best used to perform whats known as Worst-case analysis whereby the tolerance (assuming symmetrical tolerances) is added or subtracted from/to the nominal dimension of each part and summed as shown in equation 3[1]. Tolerance stack-up is a more primitive method and is, in most cases, limited in its assumption of linearity, however this can be a necessary simplification given the marked increase in complexity for non-linear cases. The typical procedure for analysis in what is known as worst-case tolerance analysis shown in equation 3.

$$G_m in = N_e + T_e - \sum_{i}^{m}(N_p i - T_p i) \tag{3}$$

For non linear analysis, the functional relationships between parts is harder to identify and describe mathematically. Equation 4 shows how Taylor expansion is used to approximate the sensitivity of each functional vector. Where x_i x_j are dimensional independent variables.

$$\Delta y = \sum \frac{\partial f}{\partial x_i}\Delta X_i + \frac{1}{2}\sum_{j=1}^{n}\frac{\partial^2}{\partial x_i \partial x_j}\Delta x_i \Delta x_j \tag{4}$$

2.2 Probabilistic Tolerancing

The probability of an assembly occurring in its worst-case state is minimal so it will therefore be inaccurate in the vast majority of cases. The root sum of squares method offers a statistical approach to the problem by producing the probability of an assembly mating failure. Equation 5 forms the basis for assembly tolerancing where T_n is the tolerance of the the part and T is the overall assembly tolerance.

$$T_{Assembly} = \sqrt{T_1^2 + T_2^2 + T_3^2 + ... + T_n^2} \tag{5}$$

By extension, equation 6 is the RSS method's answer to equation 5 as it adjusts the process variance for each part by applying equation 5 to form a standard deviation for the gap. Which can be related to a Z transform to calculate the probability of exceeding an assembly gap as seen in equation 7.

$$\sigma_{Gap} = \sqrt{(\frac{T_e}{3C_p})^2 + \sum_{i=1}^{m}(\frac{T_{p_i}}{3C_{p_i}})^2} \tag{6}$$

$$Z_{G_{min}} = \frac{G_{min} - G_{nom}}{\sigma_G ap} \tag{7}$$

Like the worst case method, this technique can be adjusted to suit 2D and 3D cases using partial derivatives of functional displacements and can also be adjusted to suit process variability over long periods of time using C_{pk} instead of C_p.

2.3 Sensitivity Analysis

In the context of tolerance analysis, sensitivity analysis quantifies the effect that the dimensional variation of any one part has on the whole assembly. It is a fitting continuation from statistical tolerancing methods in that it is underpinned, in this case, by the root sum of squares of the individual variances within parts as shown in equation 8 where s_n is the variance of the n^{th} part. The proportional contribution of any tolerance towards global tolerance can be found by squaring the component tolerance to and dividing it by the square of the assembly tolerance.

$$s = \sqrt{s_1^2 + s_2^2 + s_3^2 + ... + s_n^2 +} \tag{8}$$

The limitations of this method include its inability to deal with long term variances caused by too wear etc. however it saves time and cost by preventing the necessity for practical results from manufacturing processes.

From the above study, it is reasonable to conclude that no one method for tolerance analysis is sufficient to solve all geometric cases within assemblies therefore the method of graphical representation should be adapted to visually and computationally identify and analyse a range of common mating cases. The Monte Carlo simulation could prove to be a useful supplementation to to GRAS in its ability to simulate process capability dynamically. Given the unpredictable nature of different part interactions, without highly sophisticated CAD interpretation, the user will be required to determine functional relationships/displacements in order for the relevant statistical analysis to be conducted.

3. GRAS

The use of GRAS was tested for analysis of geometric variations in wing-box assemblies as large, compliant panels are fastened to the rib feet of the wing. Simple symbols represent parts, fasteners and Jigs and are connected according to their hierarchy as parent-child relationships are shown in a family tree. While offering an intuitive visual layout for engineers and managers to understand, this method offers potential for numerical analysis which can be enhanced with computation. Geometric variations occur as a result of tolerances which are specified by engineers yet ultimately imposed by the manufacturing methods used. With a parent-child layout, the effect of these variations upon sub assemblies can be analysed. Figure 1 shows the layout for GRAS, each stage represents an individual sub-assembly

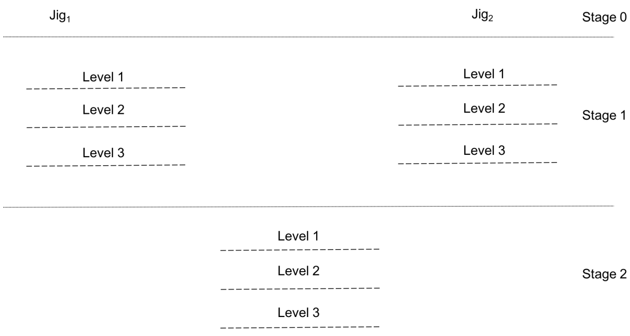

Figure 1. Heirarchical structure of GRAS

and each level refers to a part or selection of parts which may be placed in any order of parts on the same level.

4. Design for Assembly

DFA is one of two key parts of DFMA (Design for manufacture and assembly), the other being DFM, the principal philosophy is to reduce the complexity of the assembly and thus reduce the overall product time and cost. This is achieved by objectively ranking geometric, topological and mating features of parts and to not only their assemblablity but whether they are required at all in an assembly[8]. The Lucas method can be executed without specific knowledge of the assembly order making it an accessible tool for design engineers. Lucas DFA is best used in an iterative process as designs can be altered and parts merged to reduce combined the fitting, feeding and cost index of the parts.

Lucas DFA has the field, 'Shape-Factor' dedicated to describing the geometric features of a part designed to span the full range of geometric complexity and is separated into Prismatic, cylindrical and flat/thin walled geometry. This field was used to dictate the graphics which represent each part in GRAS Workspace and is shown in figure 2 along with some common joining methods.

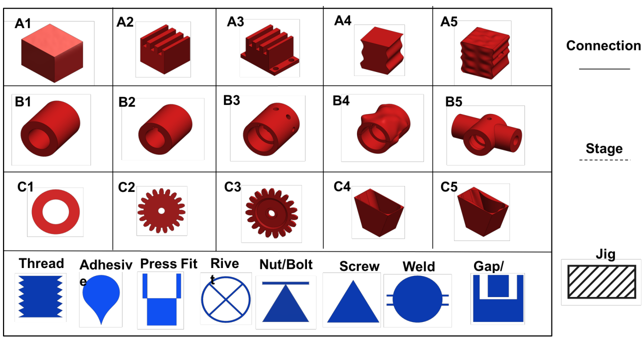

Figure 2. Icons to represnt parts in Gras Work space generated by shape factor input

5. Sequence Optimisation

DFA is more akin to sequencing optimisation than it is given credit for as many of the attributes considered by DFA are the same or closely related to those entered into heuristic sequencing optimisation processes. In the interest of concurrent engineering, it follows that combining these two processes is worthwhile and time-saving. For example, genetic algorithm is a widely practiced method of sequencing optimisation. While models vary between academics, the principals remain consistent. A sequencing optimisation study conducted by Maroua Khede et al. will be referred to for the purpose of this comparison[19]. For every manufacturing process, a fitness function must be produced which can be used to objectively measure the performance of a particular sequence. A sequence is known as a gene and an initial population of genes is produced which are randomised but conform to mating relationships given in an an interference matrix which will be mentioned later in this report. In the case of M. Khede et al, parameters of interest within the fitness function include maintainability, tool, volume and direction of feeding. direction of access is described in Lucas DFA more simply into "Straight line from above", "Straight line not from above" and so forth. Volume is present in both methodologies and tools are described within Lucas DFA with regards to the handling difficulties and could easily be adapted to include a more detailed tool description to suit the GA process. Parents are selected based on fitness function performance and randomly generated 'mask' is applied to both which determines which genes will be expressed in the child chromosome. If the child out-performs the parents then it will be carried forward for future testing and masking with other 'child' chromosomes until the algorithm converges.

From the above comparison it is clear that there is scope to combine DFA and sequencing optimisation with the use of GRAS to produce interference/functional matrices.

6. Graphical User Interface Development

Based upon research outlined above, the purpose of GRAS Workspace (GRASW) is to create an environment which asks only functional requirements of the assembly so that the user need only enter dimensional, kinematic and attribute data that will have a significant and measurable effect upon the overall assembleability. GRASW is still a work-in-progress so completed work and ideological discrepancies will be made clear in architecture and process flow charts.

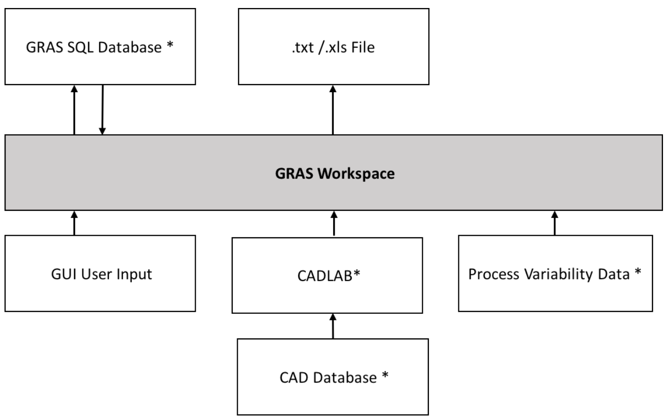

Figure 3. GRAS Architecture

Figure 2 shows a schematic diagram of the architecture for GRAS workspace. Items marked with * show areas for future

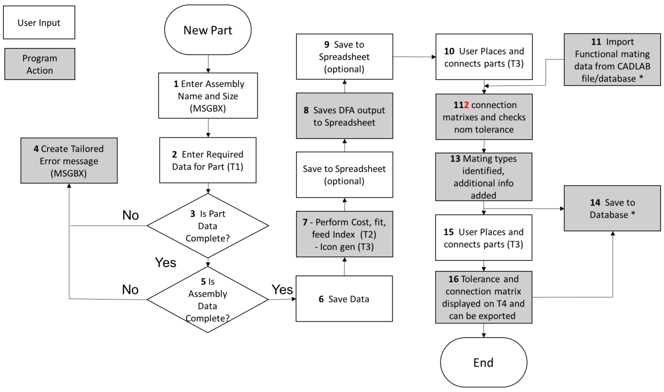

Figure 4. Flowchart showing the system and user work-flow

developement which will be discussed later.

Figure 3 shows the process flowchart for GRAS workspace. There are various other functions that allow parts to be reset and the user to return to alter data in order to change the materials, processes and dimensions in response to the DFA scores and tolerance analysis. The following processes describes the stages in figure 4 in greater detail.

1. The assembly name is added and the size of the assembly is inputted into a pop-up box.

2. Part data is added via the user interface panel shown in figure 5a. The information from the list-boxes is added to 3 data structures: n, p and cellp, n (matrix) contains the list-box item selected for the part and p (matrix) shows the corresponding DFA score while cellp (cell array) contains the corresponding string to each list-box.

3. Once the user deems the part complete the Next button is clicked and GRASW uses a for loop to identify and locate any zero elements in the n matrix. If there are any empty fields, the program will not proceed to the next row/part and a string is displayed in a message box notifying the user which items are un-filled as seen infigure 5b.

4. PCount is a variable (double) which dictates which part is being edited, once the program finds that the number PCount equals the number of parts specified for the assembly a message box appears instructing the user to save the parts and proceed to DFA out.

5. When the save button is selected, a function CostCalc is initiated. CostCalc first checks (again) if there are any erroneous zeros in the matrices using the following code.

   ```
   n(any(n==0,2),:) = [];
   ```

   It then removes the same row on other matrices as it did from n and the user is notified that "something went wrong". CostCalc loads Lucas DFA data from a program file, it then preforms the simple calculations to compute Lucas DFA and concatenates vertical matrices equal to the cost index elements such as waste factor etc.

6. The Update Data button on Tab 2 can then be pressed and the output from CostCalc appears in tables on Tab 2 as shown in figure 5c. The data can be saved to a .xls (Excel) file by clicking the 'Save to .xls' Button.

7. During the Update Data process, graphical icons (figure 2) are loaded onto Tab 3. These icons can be dragged and dropped along with the joining methods into the configuration that represents the assembly hierarchy. Connections are made by the user simply pressed and releasing the right mouse button on a part while holding the mouse still then doing the same on the part or joining method to be connected, a line is drawn between the two icons. An example of a visually complete GRAS is shown in figure 6. Drag and drop functionality requires a deceivingly large amount of code and is governed by 3 key functions required to successfully implement drag and drop functionality, one for pressing the left mouse button 'ButtonDownFcn', one for releasing it 'ButtonUpFcn', and one for moving the mouse 'WindowMotionFcn'.

   ButtonDownFcn obtains the name of the handle of object calling the function (hObject) Assigns the hObject to the variable drag Assigns value of mouse position to variable mousepos Assigns 'mousepos' to variable downPos.
   WindowMotionFcn assigns current mouse position to variable newPos. If dragging contains the handles to a calling object (as in ButonDownFcn), the difference between newPos and mousePos is assigned to dPos,'dPos' is added to the position of the the calling object (in this case an icon).

   Within the ButtonUpFcn, if dragging contains the handles to a calling object (as in ButondownFcn) the variable Drag is set to void again.

   Visual connection occurs when the user depresses and releases the left mouse button upon a graphics icon without moving the mouse and does so on the subsequent part. A line is drawn between the two icons.

8. The back-end connectivity creates matrices as shown in section 7.2. The rows and columns each equate to the numbered order of the parts the upper right half of the matrix represents the fastening type, as illustrated in matrix A_{fun} in section 7.2. To represent gaps or extrusions a negative number must be placed if the mating part is to be inserted into the part. GRASW then carries out tolerance stack up analysis as well as RSS analysis. For simplicity, the system assumes linearity as non-linearity is only feasible with a great deal more information from the part which may only be possible through CAD interpretation using software such as CADLab. Other

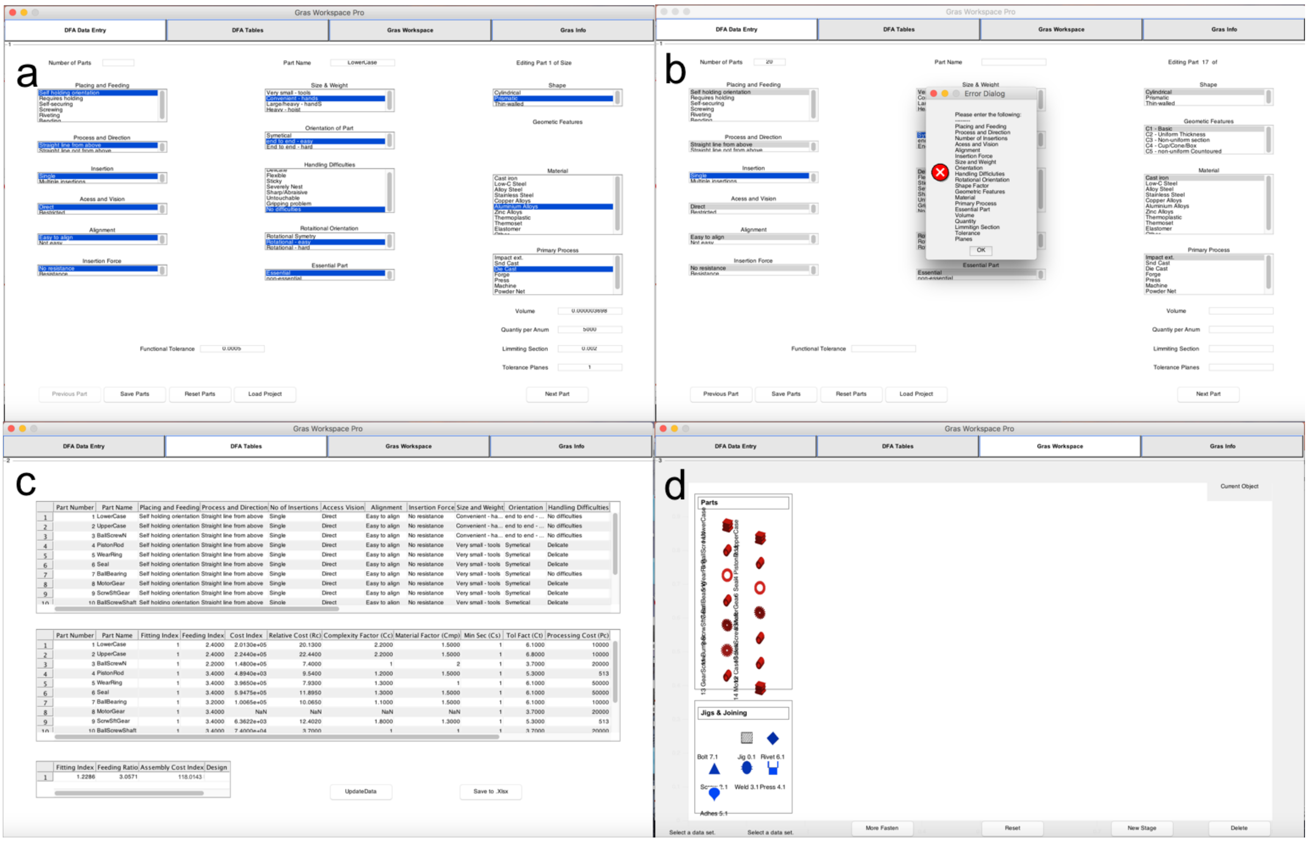

Figure 5. a: DATA Entry GUI b: Tailorised error warning c: DFA output stage d: Generated figures

features are added to aid the front end (visual) or back end (computational) outcome. Such as the ability to remove unwanted connections or fasteners, add a stage to neatly separate sub assemblies with a dashed line. Other error prevention measures include a short piece of code that removes any whitespace from the beginning or end of a part name added which will case the system to crash.

7. Case Study

This section follows the GRAS Workspace process from project initialisation to statistical tolerance analysis using a re-designed SMC servo actuator, shown in figure 4, as an example.

7.1 DFA Output

DFA metrics are the first numerical output to be produced. Data entered on tab 1 is passed to a separate function which calculates the fitting and feeding and cost index for each part and generates the respective ratios as well as the design efficiency consistent with Lucas DFA methodology. The DFA Output is shown in figure 5c.

7.2 Sequencing

GRASW uses the position of parts as well as connection process to determine connection between parts and interface types between sub assemblies and number is assigned to each joining method as seen in matrix A_{fun}. GRASW generates a relational matrix in which rows 1-n and columns 1-n relate to parts 1-n and a connection is denoted by placing the number assigned to the joining method in element. For example if element (8,12) contains 7.1 this means part 8 and 12 are connected with fastener 7.1. The 7 refers to the fastening type (screw) and the decimal is the number of that fastener. If a 1 is found, this is a fit of small (functional) clearance but no specific joining method. This output is for use in optimization of assembly and disassembly sequencing. This matrix saved to an Excel or .txt file. The GRAS configuration and corresponding functional connection matrix A_{fun} is shown as below.

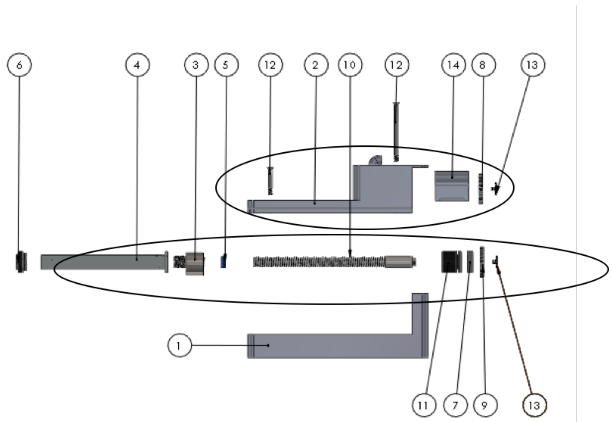

Figure 6. Re-Designed Servo Actuator - based on SMC Model

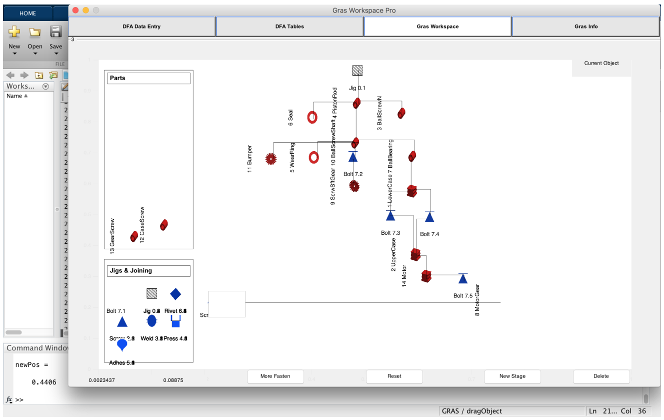

Figure 7. Screen Shot from GRAS Workspace

$$A_{fun} = \begin{pmatrix} 0 & 7.3,7.4 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ & & & & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ & & & & & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & 0 & 0 & 0 & 1 & 0 & 0 \\ & & & & & & & 0 & 0 & 0 & 0 & 7.1 \\ & & & & & & & & 0 & 7.2 & 0 & 0 \\ & & & & & & & & & 0 & 1 & 0 \\ & & & & & & & & & & 0 & 0 \\ & & & & & & & & & & & 0 \end{pmatrix}$$

7.3 Tolerance

Initially, a comparison between the nominal part tolerance stated on tab 1 is compared with typical process values unique to the material-process combination. These are compared and flagged if there is a potential incompatibility. The manufacturing capability tolerance is calculated using equation 3. However, nominal process capability alone is not a reliable test of tolerance accumulation.

The matrix is generated computationally based on the functional relationships between the parts. In this case, a functional relationship represents a coupling between parts that is pertinent to assembly planning AND is dependent on tolerance. For example, the diameter of the ball screw shaft relative to the case is not important as the clearance is large and far exceeds the tolerance range of either of the process capabilities. However, the outer diameter of the bearing is used to locate the lower circled sub assembly in figure 5 concentrically to the groove on the inside of the case (Not pictured) a location which is also a crucial step to the assembly process.

In later versions of GRAS Workspace, the functional dimension for each part in each functional connection will be requested from the user so that all desired connections can be analysed using the correct methodology for the joining type. As an example of the potential for GRAS Workspace to compute worst-case stack-up analysis, the joining of part 1 and 2 (lower case and upper case) by bolts 7.3 and 7.4 is shown below. For simplicity the case is assumed to be linear (which conveniently it is). The two M4 bolts must pass through the hole in the top case and into a threaded hole in the bottom case which for the purpose of this analysis can be of diameter $4.1mm$. This configuration can be assumed to be a mating pin and hole assembly problem illustrated in figure 8. The results shown in table 1.

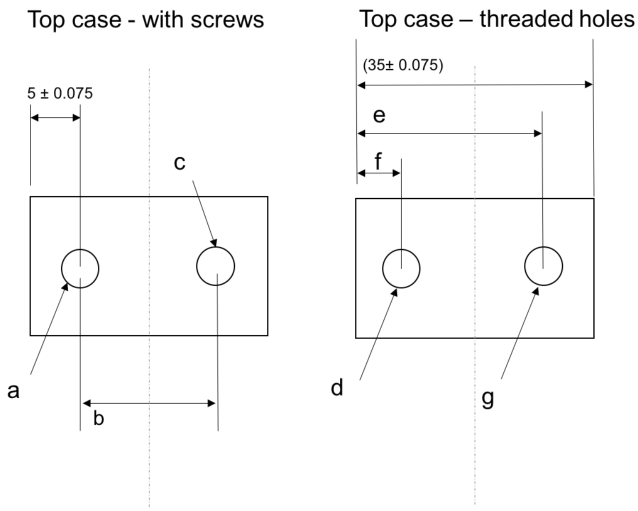

Figure 8. Pin-hole assembly case

Table 1. The choice of options.

Dimension	Nominal Value	Tolerance Squared (mm)
a	$-4/2$	$\pm 0.005/2)^2$
b	-25	$(\pm 0.075)^2$
c	$-4/2$	$\pm 0.005/2)^2$
d	$-4.1/2$	$\pm 0.005/2)^2$
e	$+30$	$(\pm 0.075)^2$
f	-5	$(\pm 0.075)^2$
g	$+4.1/2$	$\pm 0.005/2)^2$
sum	$+0.1$ (nominal gap)	± 0.13 (gap Growth)

Application of equation 5 to the third column yields a root sum of squares of $\pm 0.13mm$ which equates to a maximum gap growth of $0.13mm$ however if this is subtracted from the nominal gap ($0.1mm$) this indicated a $0.03mm$ interference. Without a geometric variability distribution, it is impossible to predict the likely hood of any case arising. The next section will determine such a probability using fictitious yet realistic process distributions. The geometric variablitiy brought about by the manufacturing process can be estimated for the above case. Table 2 shows the approximations made.

Table 2. Statistical Tolerance analysis

USL	LSL	σ	C_p	σ_{Gap}	Z_{Gap}
2.25	-2.25	0.75	1.125	0.0282	3.53

A Z transform of 3.5 equates to a DPMO of 0.004 which equates to 4000 mating failures in 1 million. As manufacturing methods are altered and expected process capabilities change, with GRAS Workspace the user can evaluate their decisions based on the statistical tolerance output in order to ensure the specified tolerance is in-line with the process capability to ensure the correct quality is delivered for the desired price.

8. Further Development

As eluded to in section 6.0, the development of GRAS Workspace is ongoing as producing user-friendly software of this level of sophistication is a substantially difficult process.

The following section describes measures which have already been taken to make it scalable and ways in which this can be improved. In addition, further developments are also discussed to reduce required user input and increase the accuracy of the results.

Grass workspace was designed to be as scalable as possible in the given time however further improvements can be made to increase this. Control of variables is important when developing software, when a new function is called from the main function, it needs to have the necessary variables passed to it so it can use them or change them then pass those variables back to the main function (maybe with some new ones) GRASW so far has all of the callback functions nested in the main function due to speed of development for this project however for scalability these callbacks should be placed in separate files.

Integration with an SQL database will increase the scalability of this project as it will allow users to save and upload part information from a cloud database and for multiple users to contribute to the addition of other joining methods.

The use of heuristics to suggest the best material-process combination for the chosen geometry would be a useful tool as well as locally applying similar algorithms to carry out sequencing optimisation mentioned in section 5. Additionally, drawing inspiration from the Monte Carlo simulation, the ability to simulate the geometric variability of a manufacturing processes taking into account long term time responses will provide more accurate tolerance data. Furthermore, as was mentioned in section 5, given the similar input demands between DFA and sequencing optimisation, GRAS Workspace could have native sequencing optimisation tool to add additional value to the process with little extra time.

As is already practiced in a number of software packages, the integration of CAD software is hugely powerful. Resources such as SolidWorks™ API or CADLab could be used to extract mating information and crucially identify the functional relationships and dimensions of parts so that the graphical representation may be completed autonomously.

9. Conclusion

This project has seen significant progress in the design and development of a computer aided tolerancing tool that allows for concurrent practice of several product development methods and has indubitably proven the concept of such a process. With further refinement, GRAS Workspace could prove to be a commercially viable tool for businesses to use to reduce the overall cost in the product development process bought about by time saving, energy saving and reduced material wastage.

Acknowledgements

I have been fortunate enough to receive counsel from various members of the School of Engineering at the University of Birmingham and therefore thank the institution for allowing me to use their intellectual resources and staff. My thanks in particular go to Soran Parsa for his support with sequencing optimisation and to Carolina Lavecchia for aiding the application developement process. Finally I would like to extend my gratitude to Dr Mozafar Saadat for conceptualising and guiding this project and for his steadfast mentorship throughout.

References

1. Creveling CM. *Tolerance Design: A Handbook for Developing Optimal Specifications*. Addison-Wesley. ISBN 978-0-201-63473-0.
2. Phadke MS. *Quality Engineering Using Robust Design*. Prentice Hall. ISBN 978-0-13-745167-8.
3. Ganeshan R, Kulkarni S and Boone T. Production economics and process quality: A Taguchi perspective ; 71(1-3): 343–350. DOI:10.1016/S0925-5273(00)00130-4. URL http://linkinghub.elsevier.com/retrieve/pii/S0925527300001304.
4. Islam M. Functional dimensioning and tolerancing software for concurrent engineering applications ; 54(2): 169–190. DOI:10.1016/j.compind.2003.09.006. URL http://linkinghub.elsevier.com/retrieve/pii/S016636150300215X.
5. Song X, Zhou W, Pan X et al. Disassembly sequence planning for electro-mechanical products under a partial destructive mode ; 34(1): 106–114. DOI:10.1108/AA-01-2013-006. URL http://www.emeraldinsight.com/doi/10.1108/AA-01-2013-006.
6. Momcm LS. AND/OR GRAPH REPRESENTATION OF ASSEMBLY PI.ANS ; : 7.
7. Saadat M, Sim R and Najafi F. Prediction of geometrical variations in Airbus wingbox assembly ; 27(4): 324–332. DOI:10.1108/01445150710827104. URL https://www.emeraldinsight.com/doi/10.1108/01445150710827104.
8. M Nazri Ahmad, Maidin NA, Mhd Hairizal Osman et al. REDUCING PRODUCT COST BY IMPLEMENTING DFMA METHODOLOGY LUCAS HULL: A CASE STUDY. DOI:10.13140/rg.2.2.32881.17763.
9. Boothroyd G, Dewhurst P and Knight WA. *Product Design for Manufacture and Assembly*. 2nd ed., rev. and expanded ed. Number 58 in Manufacturing engineering and materials processing, M. Dekker. ISBN 978-0-585-40709-8 978-0-8247-0584-8.
10. Eastman CM. *Design for X: Concurrent Engineering Imperatives*. Springer Science & Business Media. ISBN 978-94-011-3985-4.
11. Ullman DG. *The Mechanical Design Process*. 4 edition ed. McGraw-Hill Education. ISBN 978-0-07-297574-1.
12. Eskilander S. A Method For Product Design: DFA2 ; : 190.
13. Taguchi G. *Taguchi on Robust Technology Development*. ASME. ISBN 978-0-7918-0028-7. DOI:10.1115/1.800288. URL http://ebooks.asmedigitalcollection.asme.org/book.aspx?bookid=228.
14. Polini W. Geometric Tolerance Analysis. In Colosimo BM and Senin N (eds.) *Geometric Tolerances*. Springer London. ISBN 978-1-84996-310-7 978-1-84996-311-4. pp. 39–68. DOI:10.1007/978-1-84996-311-4_2. URL http://link.

springer.com/10.1007/978-1-84996-311-4_2.
15. Chase KW and Magleby SP. A Comprehensive System for Computer-Aided Tolerance Analysis of 2-D and 3-D Mechanical Assemblies ; : 16.
16. Salomons OW, Poerink HJJ, Slooten FV et al. *A Computer Aided Tolerancing Tool Based on Kinematic Analogies*.
17. The TTRS: 13 oriented constraints for dimensioning, tolerancing and inspection. URL https://www.researchgate.net/publication/284831984_The_TTRS_13_oriented_constraints_for_dimensioning_tolerancing_and_inspection.
18. Bjrke . *Computer-Aided Tolerancing*. ASME Press. ISBN 978-0-7918-0010-2.
19. Kheder M, Trigui M and Aifaoui N. Disassembly sequence planning based on a genetic algorithm. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 2015; 229(12): 2281–2290. DOI:10.1177/0954406214557340.

Appendix 1. Abbreviations

CAE
Computer Aided Engineering

CAD
Computer Aided Design

CAT
Computer Aided Tolerancing

DPMO
Defects Per Million Oppertunities

FD&T
Functional Dimensioning and Tolerancing

FROOM
Feature and Relation based Object Oriented Modelling

GA
Genetic Algorithm

GD&T
General Dimensioning and Tolerancing

GRAS
Graphical Representation of Assembly Sequencing

GRASW
Graphical Representation of Assembly Sequencing Workspace

PPM
Parts Per Million

TTRS
Technologically and Topologically Related Surfaces