

**Generic Automotive Engine Intake Port Optimisation**

---

**PROJECT ENGINEER(S):** Matthew Cross, Daniel Smith

**DATE:** 14 February, 2005

**SIGNATURE:** \_\_\_\_\_

**APPROVED BY:** \_\_\_\_\_

**REQUESTED BY:**

---

**SUMMARY**

This report aims to describe the steps taken to optimise a generic automotive intake port geometry for mass flow rate under a given pressure drop using CFD.

A series of 11 unique parameters have been defined using Sculptor™ together another 4 parameters which have been defined as functions of these. The cross-sectional area of the intake port was also constrained to within +/- 15% of the baseline geometry during the optimisation.

An in-house code has been used to define an initial set of experiments based on a Latin Hypercube sampling technique. A response surface approach has then been used to identify areas of the design space that require refinement and subsequently has predicted an optimum design that is situated within the bounds of the cross-sectional area constraint.

A total of 124 runs were solved as part of the study and an overall gain of 1.94% was found within the prescribed limits.



# CONTENTS

<b>SUMMARY</b>	<b>1</b>
<b>1. INTRODUCTION</b>	<b>3</b>
<b>2. SCULPTOR METHODOLOGY</b>	<b>6</b>
2.1. ASD Volume generation	7
2.2. Parameter creation	10
2.3. Making smooth deformations and finding parameter limits	11
2.4. Creating the other ASD volumes	12
2.5. Time summary	13
<b>3. DESIGN OF EXPERIMENTS AND AUTOMATION</b>	<b>14</b>
3.1. Design of experiment	14
3.2. Automatic generation of experiments	14
3.3. Performance function and the area constraint	15
<b>4. RESULTS</b>	<b>16</b>
4.1. Results of optimisation	16
4.2. Design variable sensitivity	17
4.3. Comparison of baseline and optimised geometries	18
4.4. Comparison of flow structure	19
4.5. Time summary	22
<b>5. CONCLUSIONS</b>	<b>23</b>
<b>6. APPENDIX A</b>	<b>24</b>

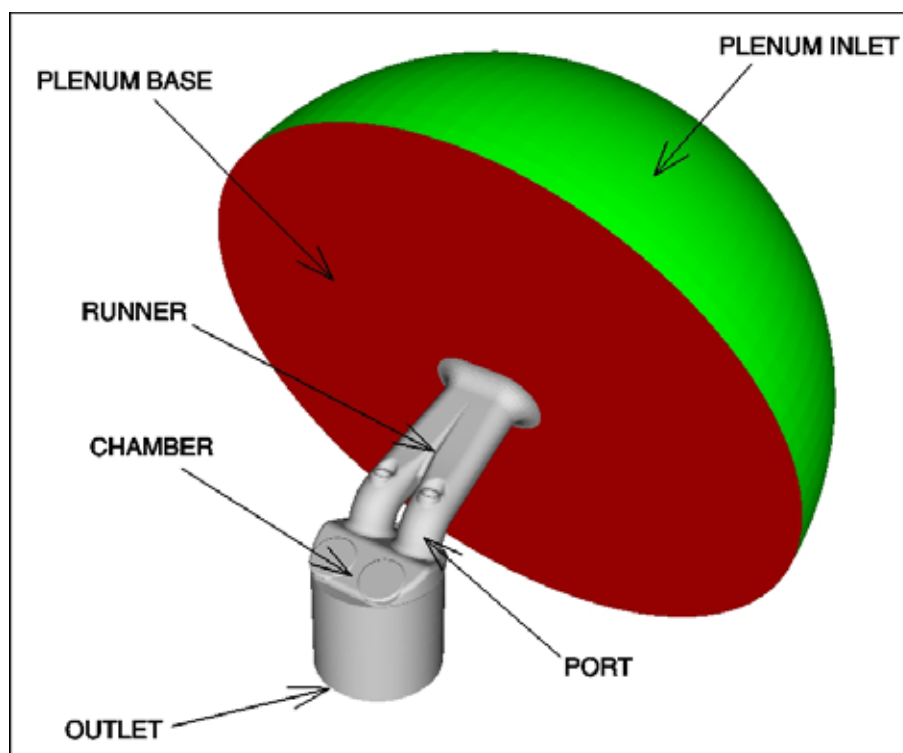
## 1. INTRODUCTION

This report presents the results of a study to optimise the geometry of a generic automotive engine intake port using CFD. The goal of the optimisation process was to increase the mass flow entering the chamber for a fixed pressure drop.

A total pressure inlet of 0 Pa was used on the plenum inlet and a pressure outlet of  $-16.9\text{kPa}$  was applied to the exit. All analyses were carried out at a single valve lift of 10.0mm and used the standard  $k-\epsilon$  turbulence model with non-equilibrium wall functions. The volume mesh was fully tetrahedral and contained 2.7 million cells

The CFD model was supplied by a third party to demonstrate the use of Sculptor™ to this type of application and is shown in Figure 1.

Figure 1 – Intake Port Geometry



A series of 11 individual parameters were defined to modify the geometry of the intake port. These variables, DV1 to DV11 are identified in Figure 2 to Figure 4 together with 4 additional variables, DVTEMP1 to DVTEMP4, that are defined as functions of DV1 to DV11.

The cross-sectional area of the port is measured in the two locations identified in Figure 5. The optimisation was constrained so that the areas of the final optimum were within  $\pm 15\%$  of the baseline.

Figure 2 – Variable DV1

Junction Shape Variable – DV1

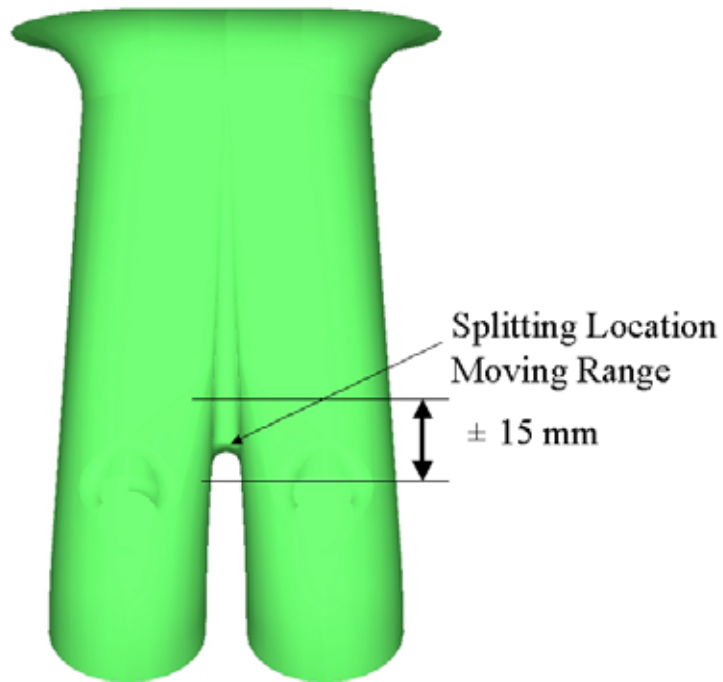


Figure 3 – Variable DV2 – DV6

Runner shape variables – (DV2 – DV6)

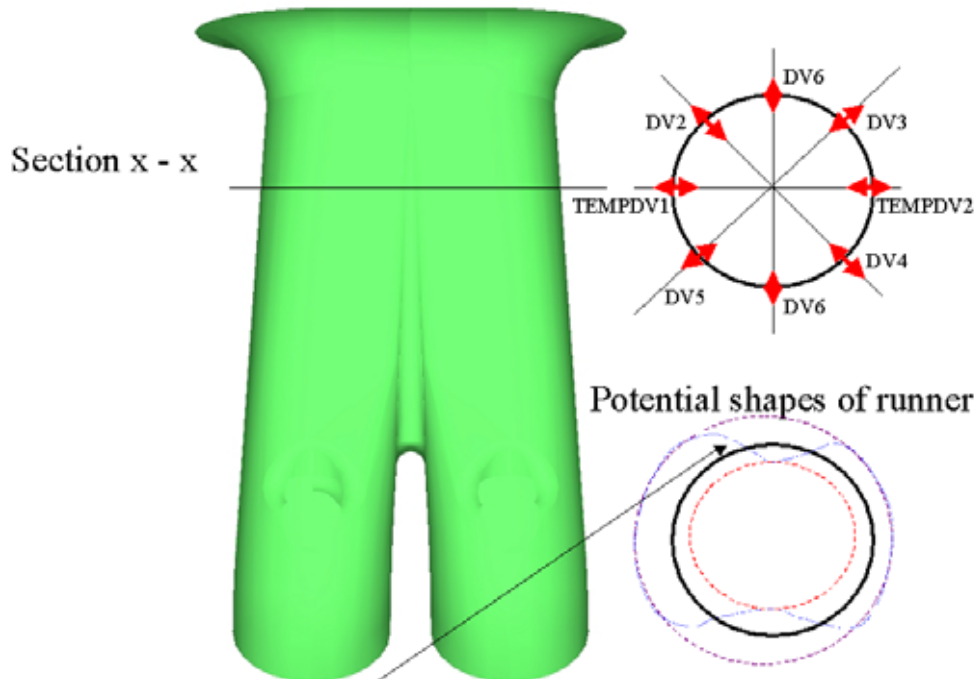


Figure 4 – Variable DV7 – DV11

Port shape variables – (DV7 – DV11)

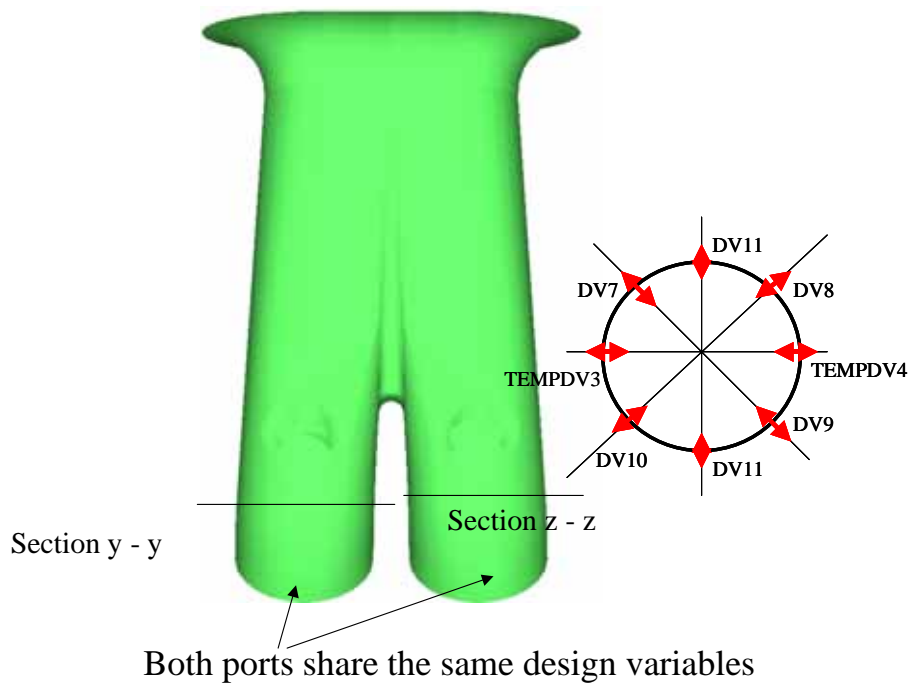
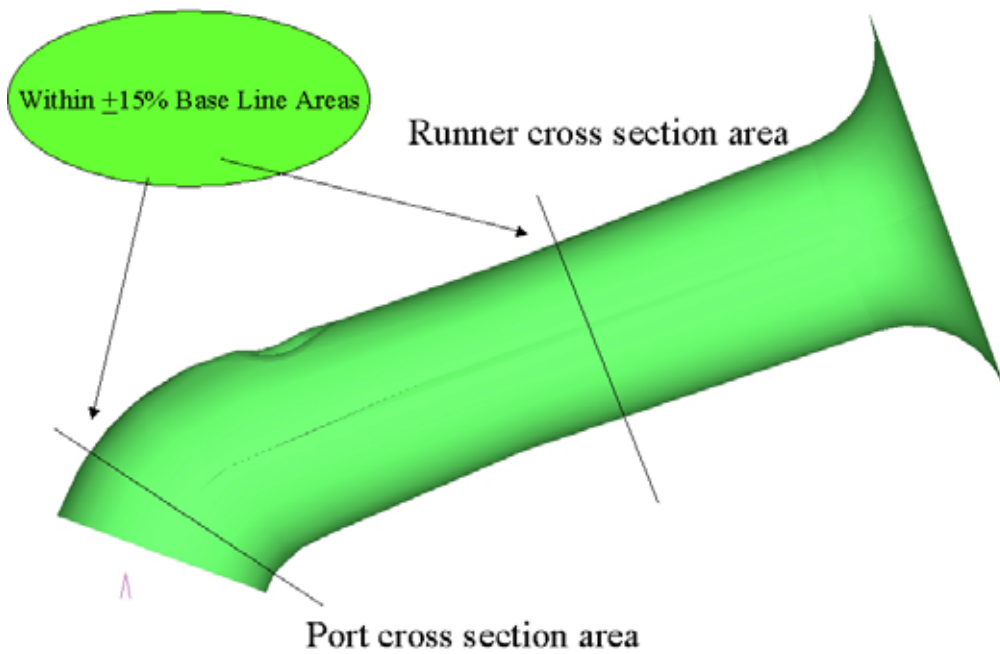


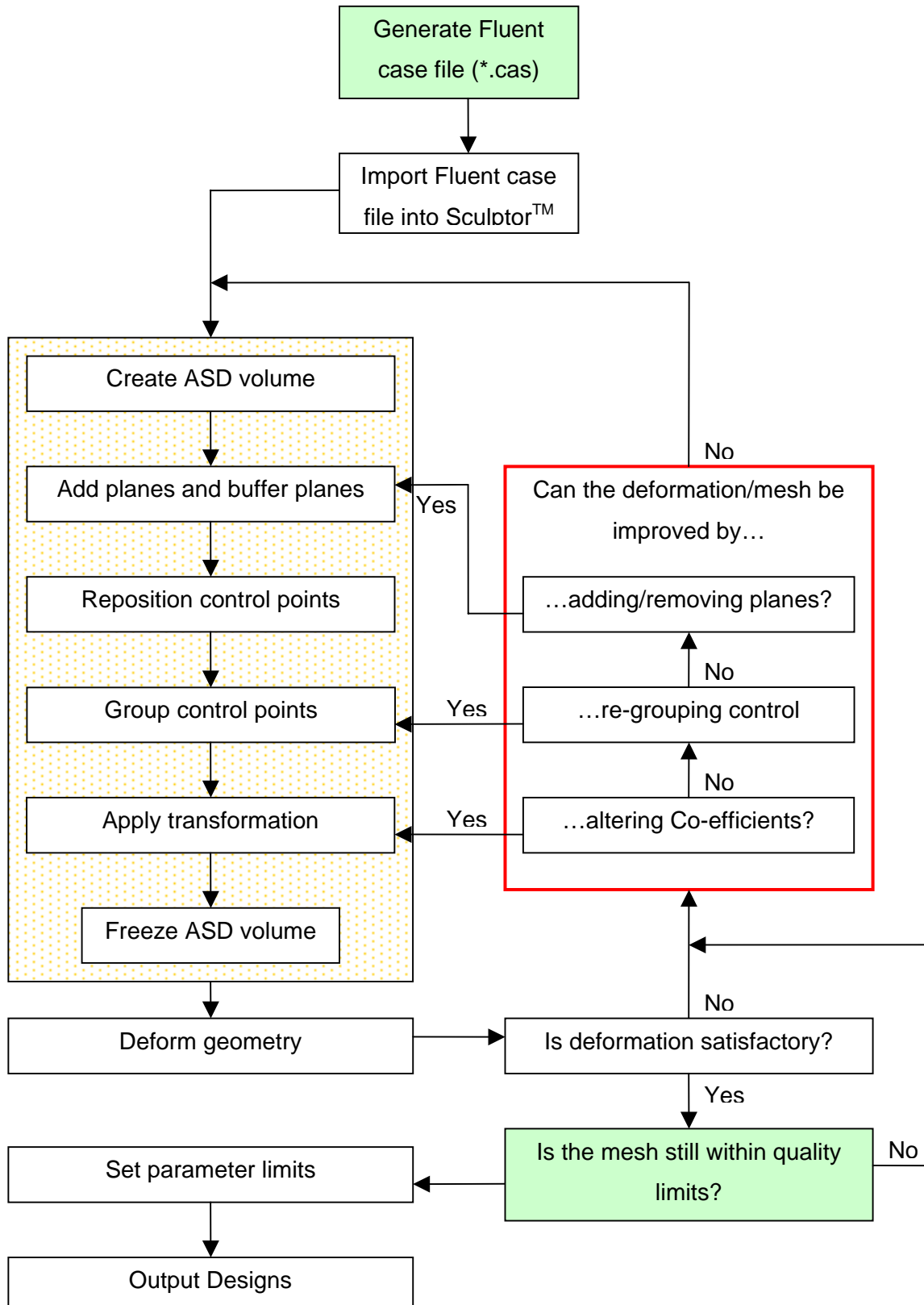
Figure 5- Cross section constraints

Cross section area constraints



## 2. SCULPTOR METHODOLOGY

The parametric deformations to the baseline geometry have been made using Sculptor™. This section of the report presents the method used to define these deformations. An overview of the process is shown below.



## 2.1. ASD Volume generation

Once the CFD volume mesh has been imported into Sculptor™ (in this instance a FLUENT case file – see Figure 6) it is possible to begin constructing the ASD (Arbitrary Shape Deformation) volumes around the geometry. These ASD volumes provide the basis for the deformations.

Figure 6 – FLUENT case file read directly into Sculptor™

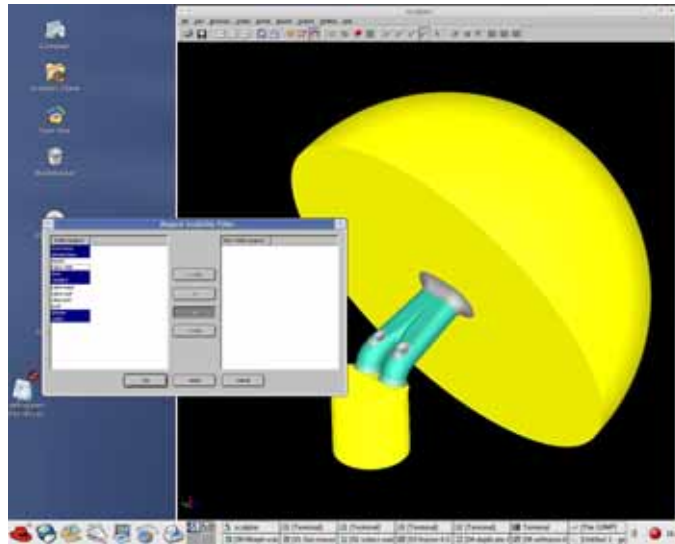
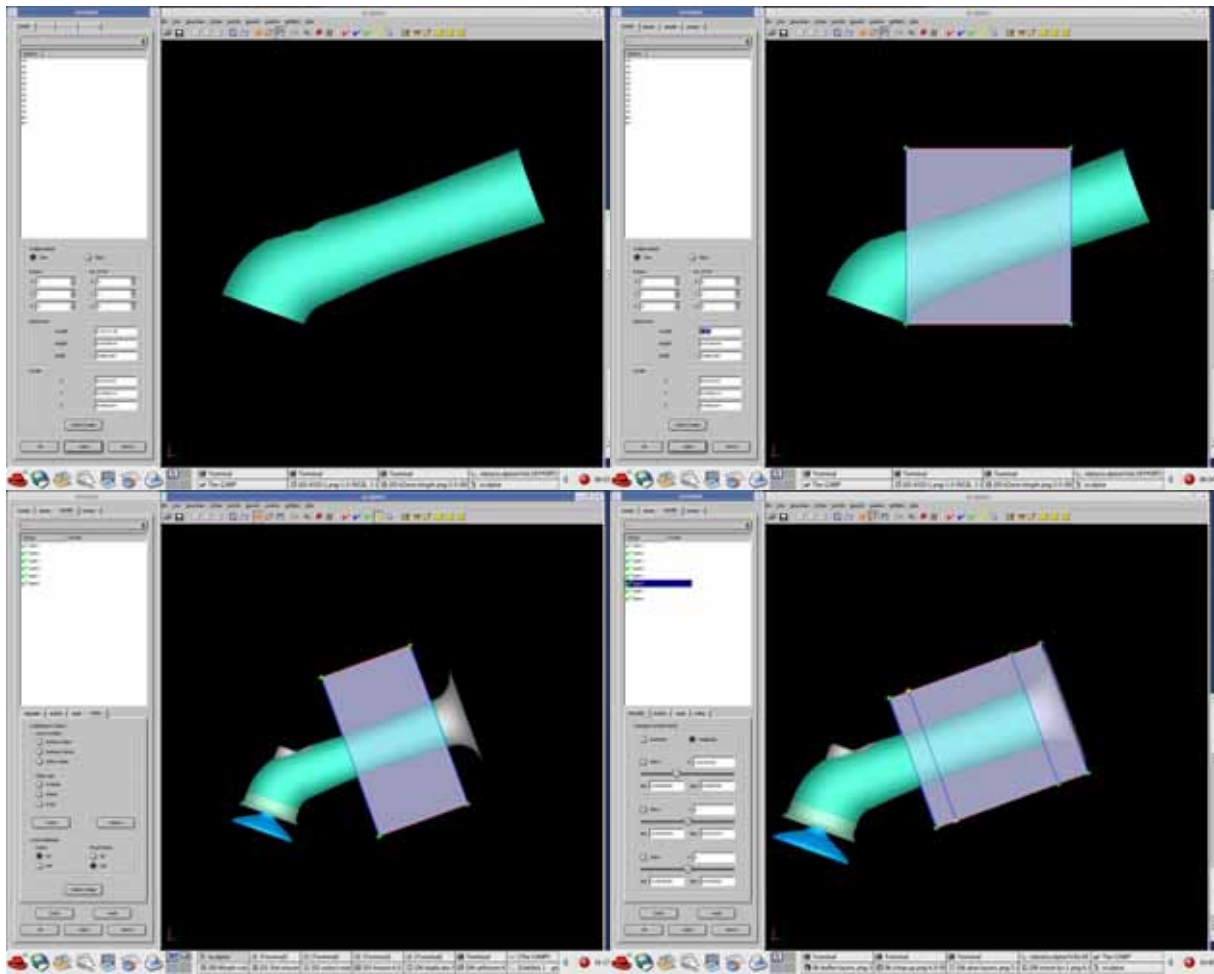


Figure 7 shows the initial stage of defining the first ASD volume around the runner. This volume is intended to make the deformations specified in Figure 3.

Essentially an initial box is described around the boundary zone of interest that is then positioned using tools within Sculptor™. This ASD volume is then subdivided (or extended) with a series of planes. The nodes at intersections of the planes are later used to control the deformation of the geometry.

Any geometry/volume mesh that sits outside of the ASD volume is not modified so it is possible to isolate changes to an accurately defined volume.

Figure 7 – Initial generation of ASD



The further subdivision of the initial ASD is shown in Figure 8. The positioning of the ASD planes close to the FLUENT boundary zone can also be identified.

The final ASD volume around the runner is displayed in

Figure 9.

Figure 8 – Further subdivision of ASD

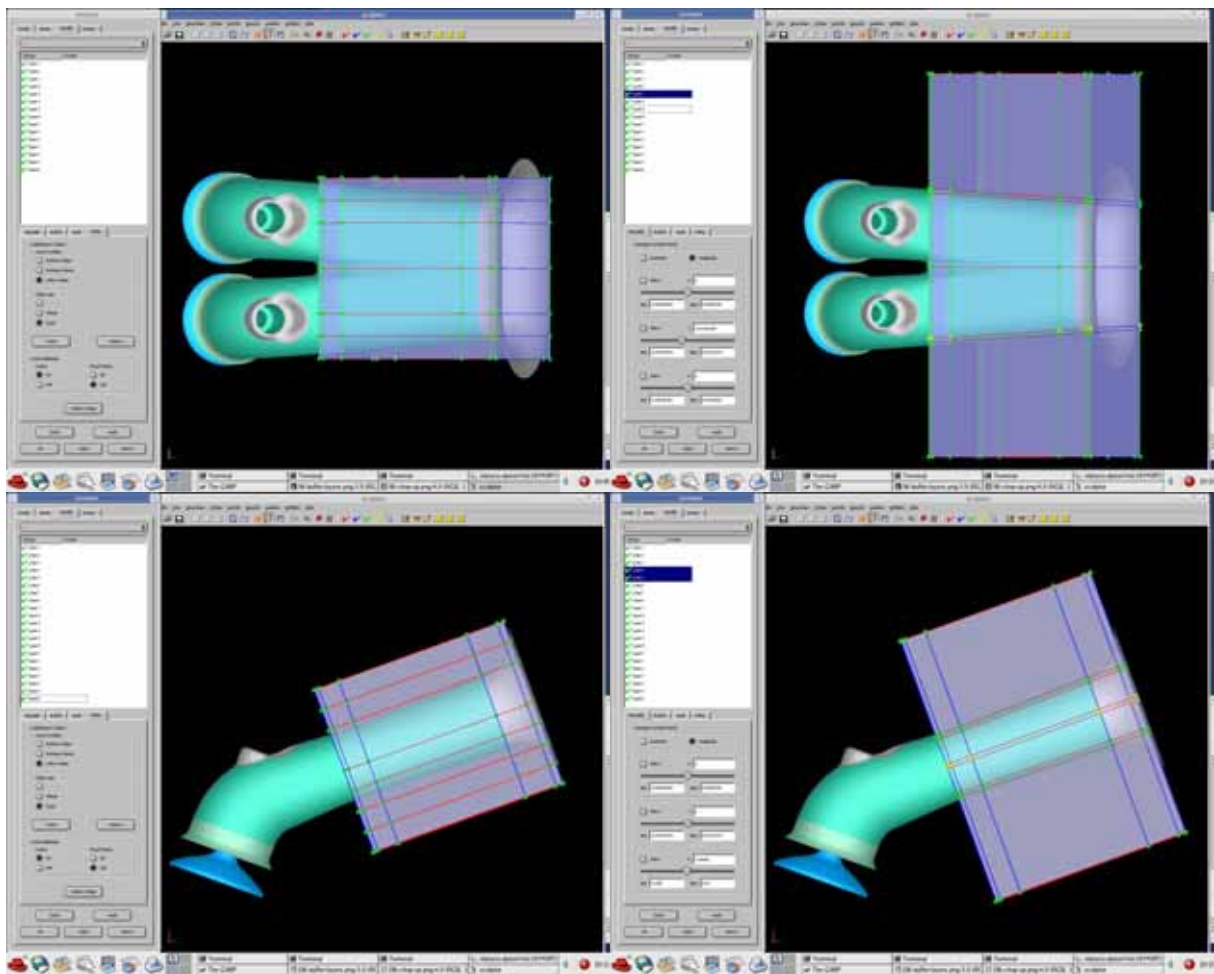
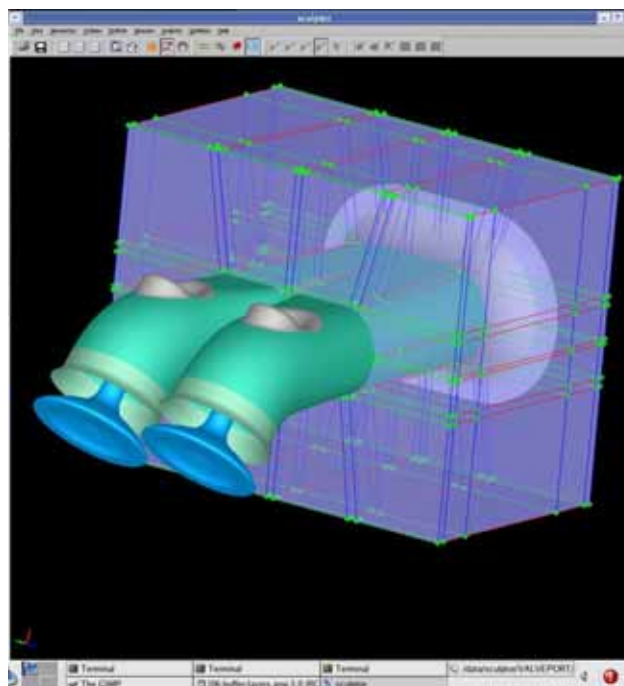


Figure 9 – Completed ASD volume around runner

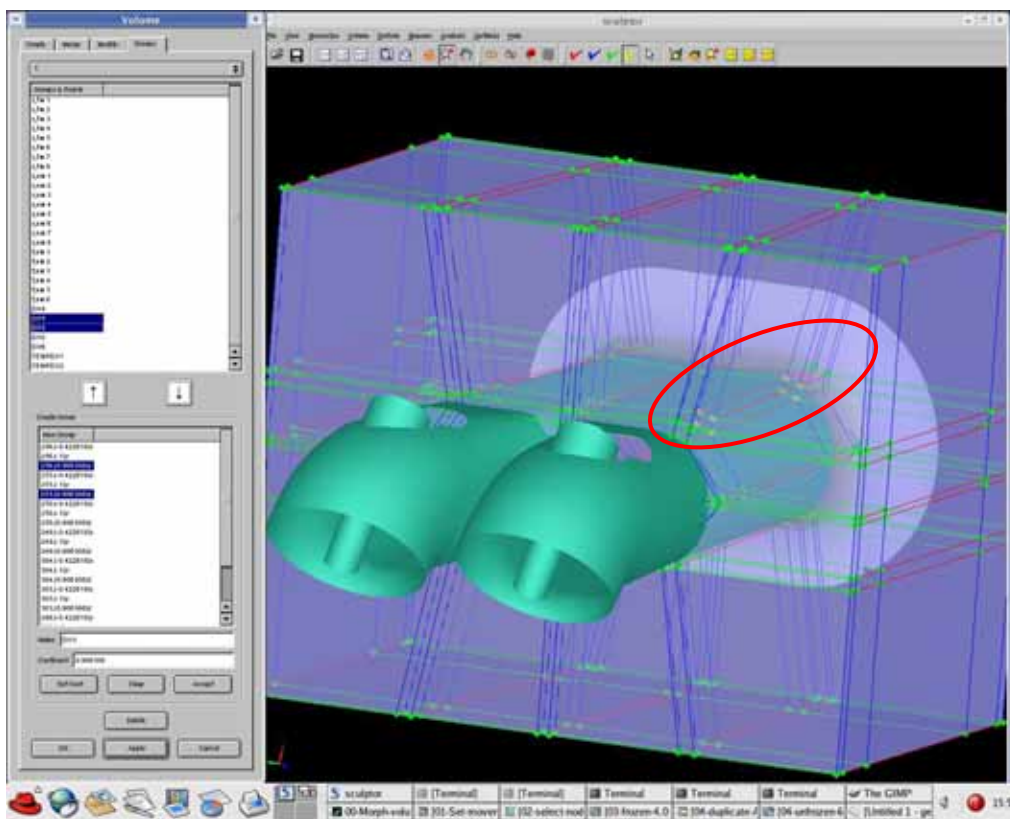


## 2.2. Parameter creation

The next stage in the Sculptor™ process is to create relationships between the nodes within the ASD volume to define parameters that deform the geometry as required.

These nodes are initially selected by the user (shown in yellow and highlighted in Figure 10) and grouped together. Each group of points can later be translated, scaled or rotated in either cartesian or parametric space.

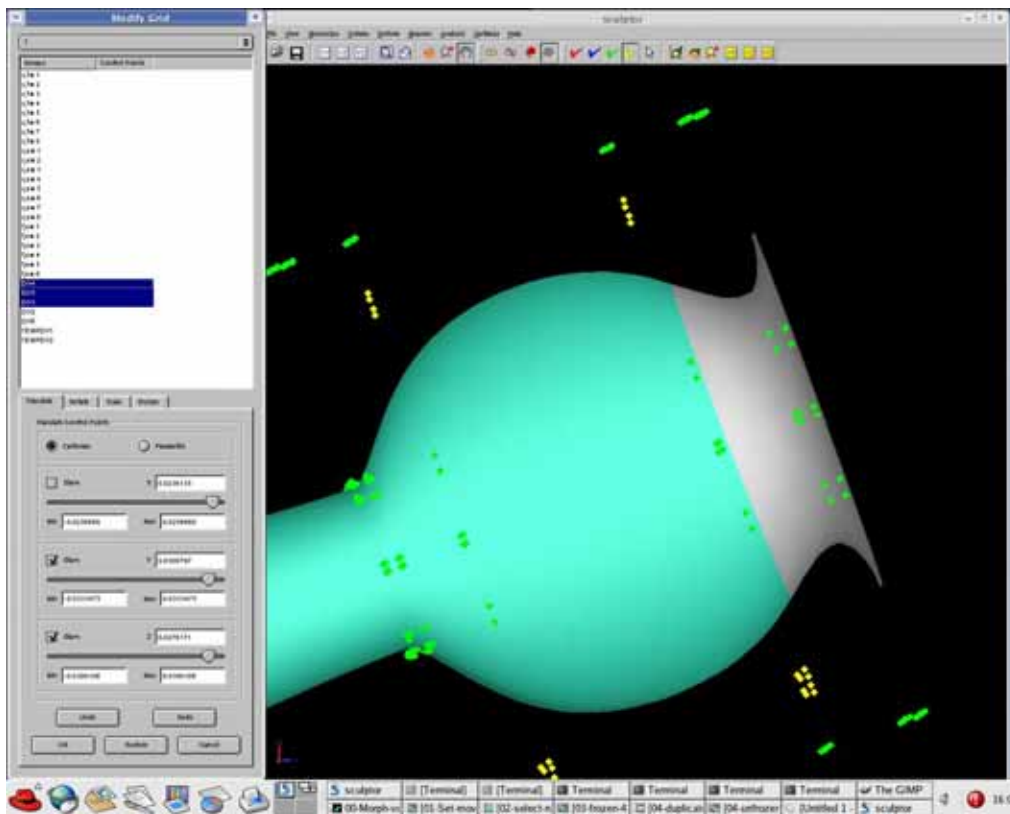
Figure 10 – Points selected for grouping



In addition to this it is possible to add coefficients to each individual point within a group. This allows points to be moved in diagonal directions or to be moved in different directions to the other points in a group just using a single parameter. An example of this is shown in Figure 11 where a single parameter has been used to define a powerful change to the geometry.

There was some iteration of these groups and the position of the planes before suitable parameters were defined.

Figure 11 – Example of using coefficients in a group



Before these deformations can be made however the ASD volume must be 'frozen'. This process maps the nodes within the FLUENT case file to the parametric co-ordinates of the ASD volume allowing distortion of the ASD to deform the FLUENT volume mesh smoothly and interactively.

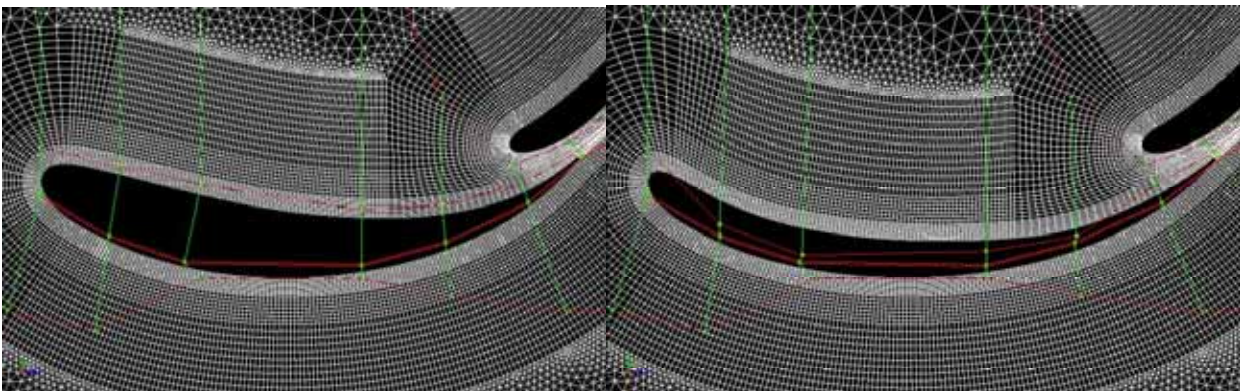
### 2.3. Making smooth deformations and finding parameter limits

Once the volume has been frozen it is possible to deform the geometry with each of the parameters. Before an optimisation can begin bounds need to be defined for each parameter. These bounds are generally a function of geometrical constraints (such as restrictions to the design space) or cell volume/skewness limits.

Currently, the best way to find these limits is to make a range of changes to a parameter and then check the case in FLUENT or TGrid to ascertain the acceptable range of movement. An in-built skewness and cell volume checker is in development for Sculptror™ as is collision detection with constraint surface which will improve the speed of this process.

The deformations made to the FLUENT case file are distributed extremely smoothly to the volume mesh. Figure 12 shows an example of a deformation made to the thickness of a wing section.

Figure 12 – Example of volume mesh deformation



## 2.4. Creating the other ASD volumes

The other ASD volumes were generated in a similar way to that specified for the runner. Figure 13 and Figure 14 show the ASD volumes around the port junction and the port itself respectively. Of note is the way in which the planes have been positioned around the valve to ensure that the stem is only moved by a negligible amount.

Figure 13 – ASD volume around port junction

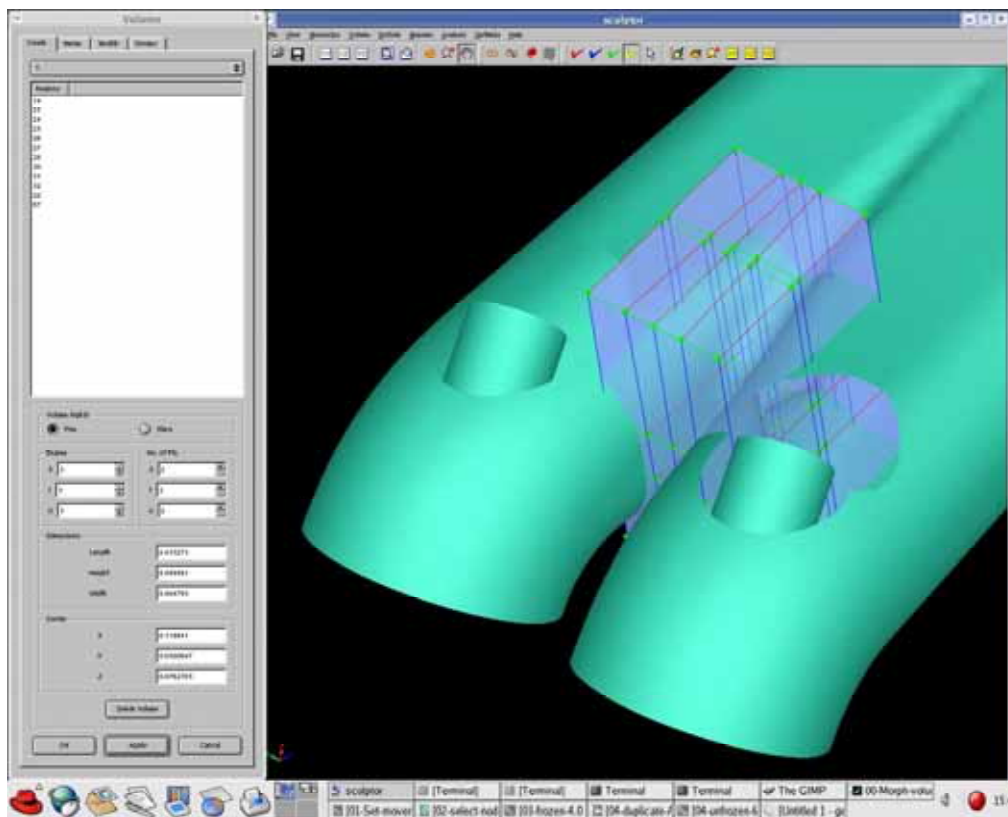
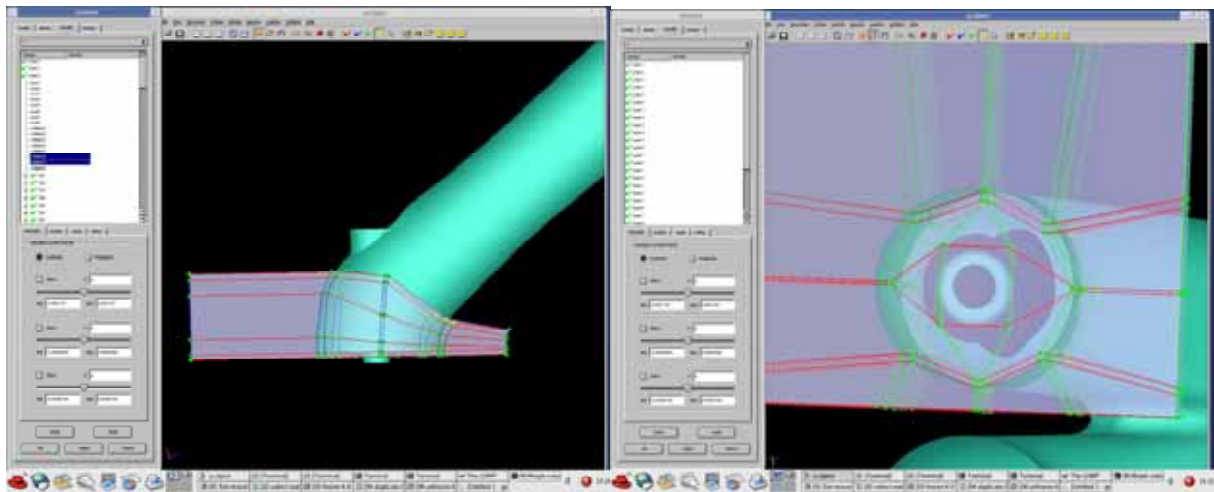


Figure 14 – ASD volume around port



## 2.5. Time summary

A breakdown of the time taken for each of the stages involved in generating the ASD volume around the various components and defining the parameters is shown in Table 1.

Table 1 – Time breakdown

Time Summary	Man time	CPU time
Import fluent case file	1 minute	-
Select visible regions	1 minute	-
ASD volume 1	1 minute	-
Setting the design space	1 minute	-
Positioning the ASD volume	1 minute	-
Adding layers to the ASD volume*	100 mins	-
Grouping points**	60 mins	-
Assigning movement co-efficient**	75 mins	-
Creating the other ASD volumes***	540 mins	-
Freezing the ASD volumes****	-	205 mins
Finding parameter limits	120 mins	-
<b>Total</b>	<b>15hrs</b>	<b>3hrs 25mins</b>
* Includes time added for second attempt		
** Includes time added for refinements		
*** Includes time added for refinements and second attempt on ASD volume 2		
**** Includes time added for re-freezing volumes after 2nd creation iteration		

### 3. DESIGN OF EXPERIMENTS AND AUTOMATION

#### 3.1. Design of experiment

Once the 11 parameters had been defined and tested it was possible to design a series of experiments that would allow a relationship between these parameters and the performance of the intake port to be calculated.

It was calculated that at least 75 initial experiments would be required in order to solve the coefficients of a second order response surface. An in-house code was used to define 77 experiments by means of a Latin hypercube sampling method.

#### 3.2. Automatic generation of experiments

Using a piece of in-house code the combination of the 11 parameters for the initial 77 experiments were converted into a series of journal files for Sculptor. From these it was possible to automate the generation of these 77 cases from Sculptor™ in batch mode thus removing the need for any human input.

In-house scripts were also used to automate the checking of each case for negative volume cells and high skewness, the generation of a FLUENT journal for each case and the processing of results.

To create, check and process all of these cases took less than 3 man-hours in total.

Significant computational time reductions were possible as each experiment could be started from the converged solution of the baseline. As only the node co-ordinates in the volume mesh are modified and not the connectivity or CFD setup then reconverging the solution is possible.

Reconverging the solution reduced the number of iterations required to get to a steady mass flow from approximately 4000 to 500 – a saving of nearly 90% of the time taken to solve each experiment.

### 3.3. Performance function and the area constraint

Throughout the optimisation process the 11 parameters were optimised for maximum mass flow rate through the system. During the majority of this process the variation in the cross-sectional area was ignored. This allowed the design space to be populated in both ‘illegal’ and ‘legal’ designs, which improved the accuracy of the response surface near the limits of the design space.

The cross-sectional area constraints were only applied in the final stage of the optimisation.

## 4. RESULTS

### 4.1. Results of optimisation

The mass flow rate for each of the different runs in the optimisation process is shown in Figure 15 with the 4 different stages are identified with different colours. After the initial experiments defined by the latin hypercube (shown in blue) two refinement stages were used to improve the resolution of the response surface in the regions of predicted maxima and minima. Then a final search stage (shown in red) was used to assess the predicted global maxima.

Note that it was only in this final search stage that the cross-sectional area constraints were applied.

Figure 15 – Mass flow rate results over optimisation process

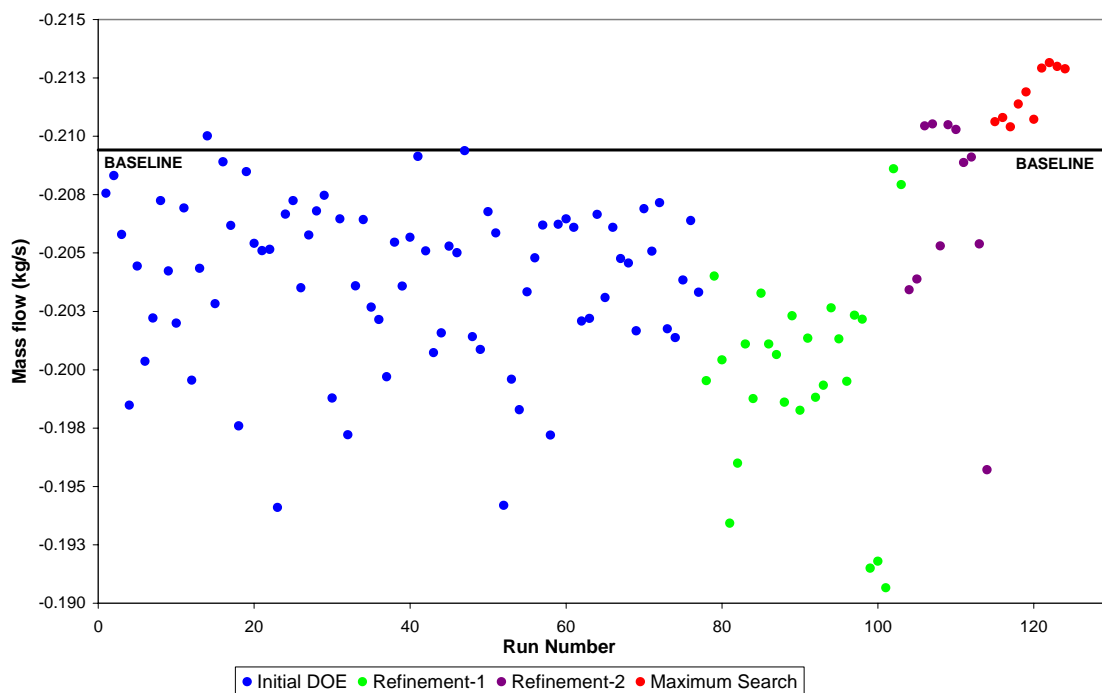


Figure 16 compares the mass flow for the baseline and the optimum geometry found within the cross-sectional area constraints (Run 123). These results are for both geometries run from an initialised solution in Fluent for 4000 iterations with the same setup.

The cross-sectional areas are compared in Figure 17.

Figure 16 – Comparison of baseline and optimum mass flow rate

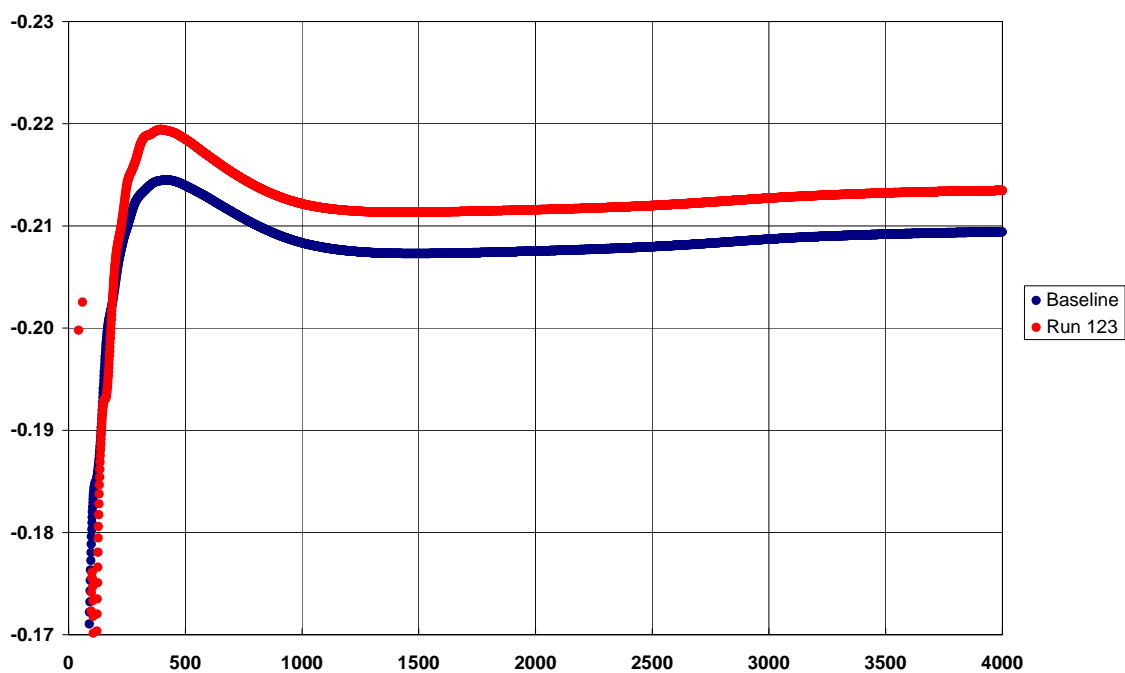
	Mass Flow (kg/s)
Baseline	0.2094
RUN123	0.2135
<b>% change</b>	<b>1.94%</b>

Figure 17 – Comparison of cross-sectional areas for baseline and optimum

	Runner	Port (lh)	Port (rh)
Baseline Area	0.002104	0.001000	0.000999
RUN123 Area	0.002248	0.000850	0.000850
<b>% change</b>	<b>6.8%</b>	<b>-15.0%</b>	<b>-14.9%</b>

A comparison of the mass flow rate convergence history for the two cases is shown in Figure 18.

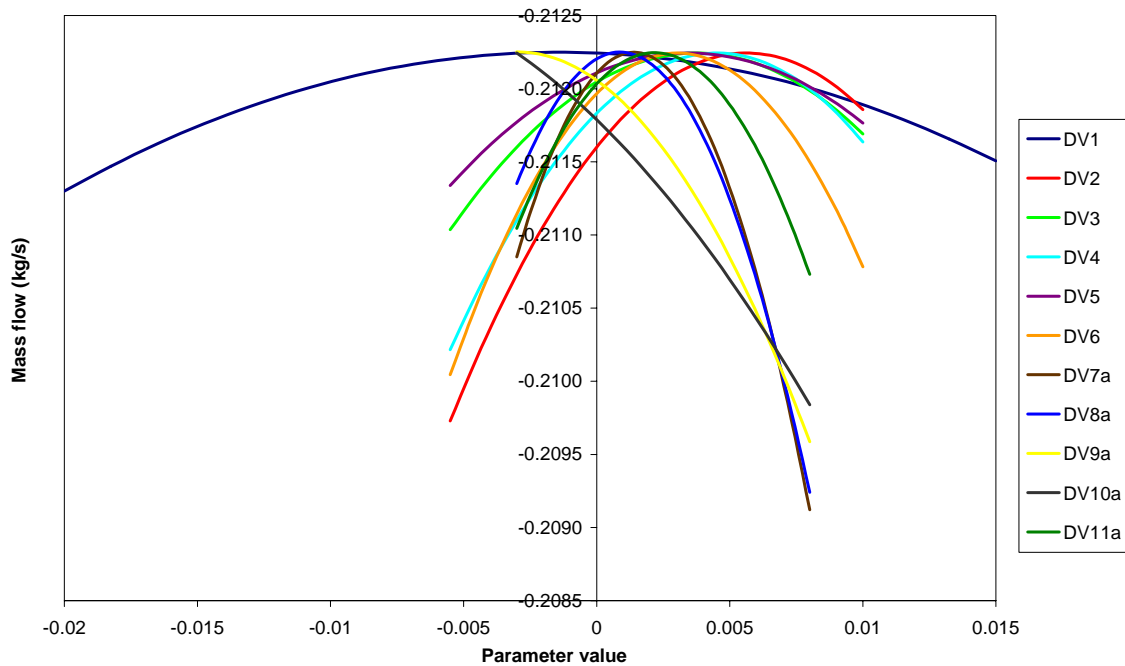
Figure 18 – Comparison of mass flow rate convergence history



#### 4.2. Design variable sensitivity

Figure 19 shows the variation in performance for each of the 11 parameters. For each curve the parameter is varied between its maximum and minimum value whilst all other variables are kept in the optimum position.

Figure 19 – Variation of performance with parameter value



Details of the variation in performance for 2 interacting parameters can be found in APPENDIX A.

### 4.3. Comparison of baseline and optimised geometries

A comparison of the original and optimised geometries is shown in Figure 20 and Figure 21. The cross-section of the runner has been increased in all directions but well within the limits of the design space. Around the port the cross-section has been dramatically reduced in the region near the centreline of the bore (Figure 21) and increased slightly on the opposite side. Figure 17 shows that the cross-sectional area here is reduced to the minimum possible within the constraints.

Figure 20 – Comparison of baseline and optimised runners

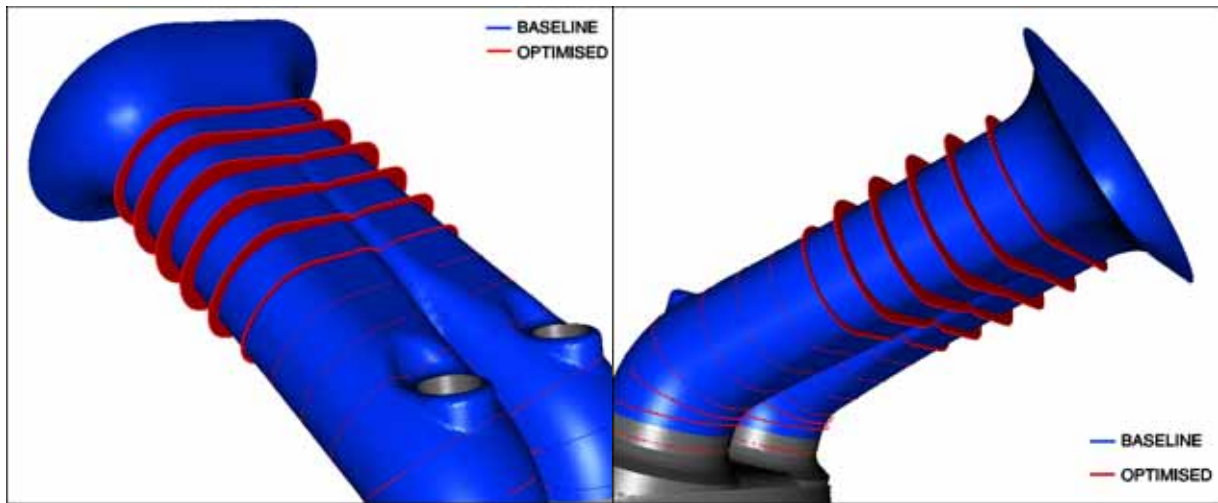
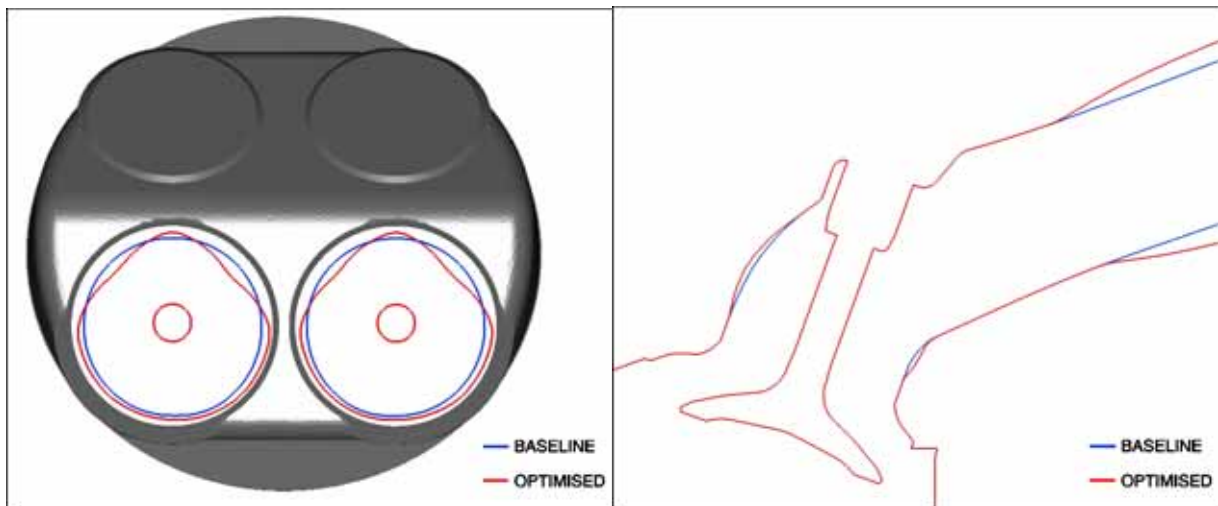


Figure 21 – Comparison of baseline and optimised geometries around the port



#### 4.4. Comparison of flow structure

Figure 22 to Figure 25 identify different aspects of the flow structure for the baseline and optimum. Figure 22 compares the velocity profile between the valve and the seat for the two cases

Isosurfaces of velocity cutaway through the centre of a valve are shown in Figure 23 and surface contours of static pressure are shown in Figure 24. Figure 25 shows the flow direction local to the intake port surface using oilflow and total pressure contours.

Figure 22 – Comparison of flow between the valve and the seat

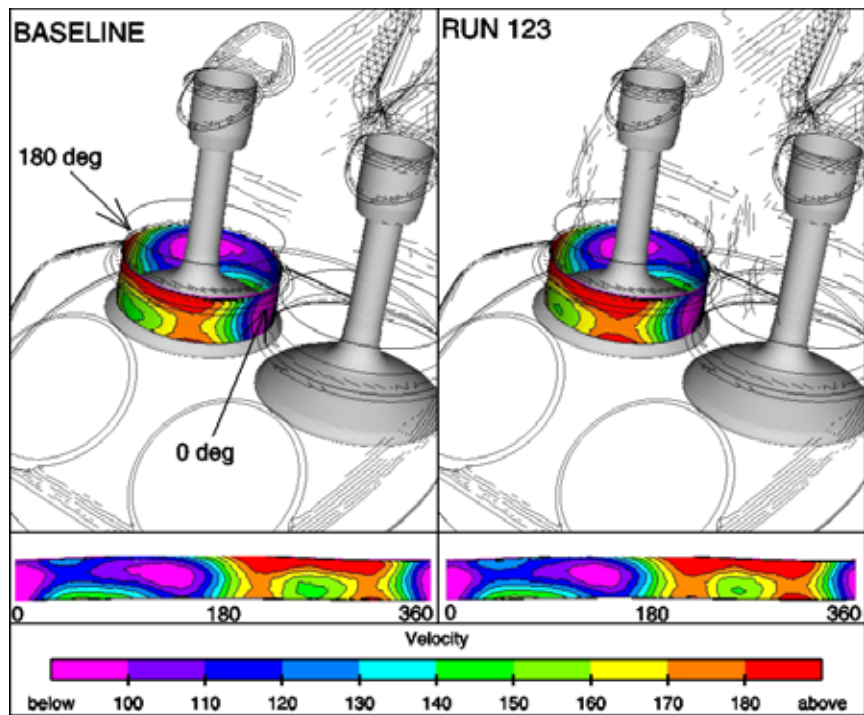


Figure 23 – Comparison of isosurfaces of velocity through the valve centreline

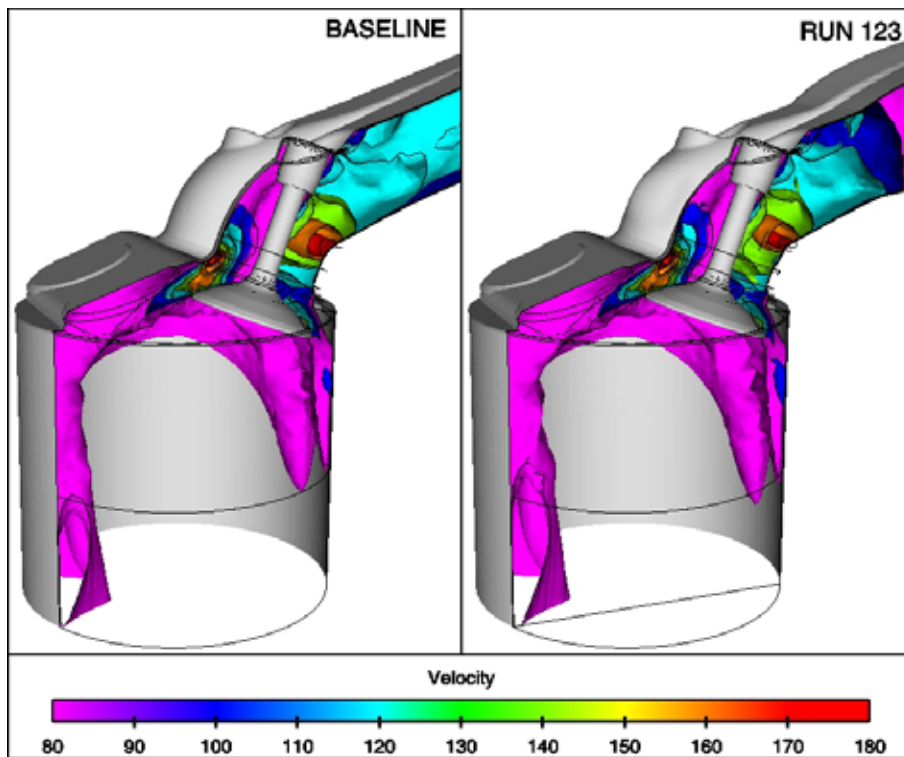


Figure 24 – Comparison of surface pressure for the two intake ports

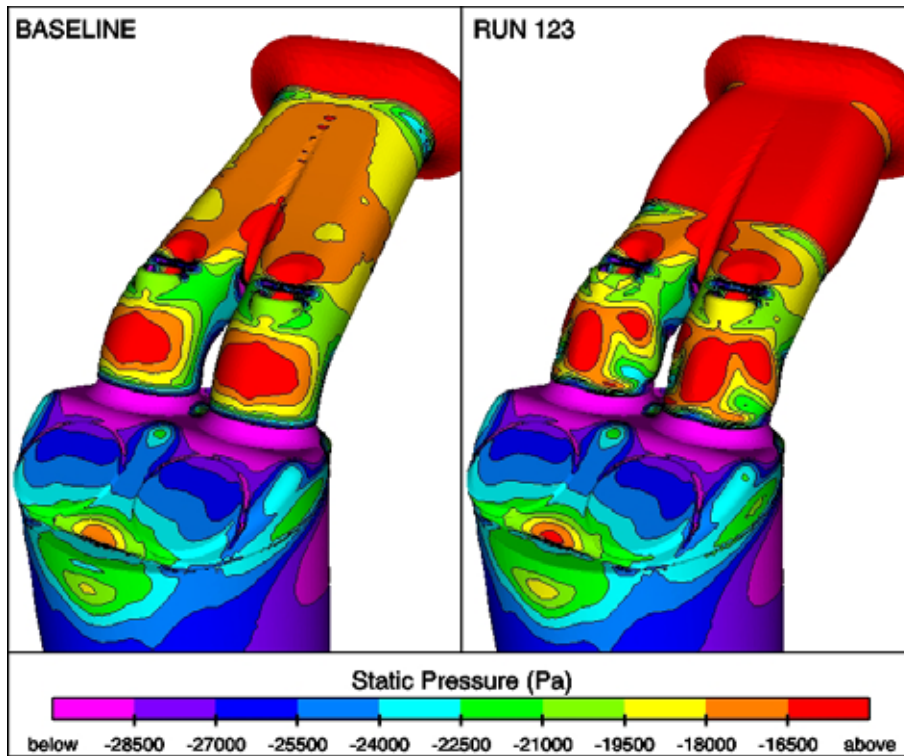
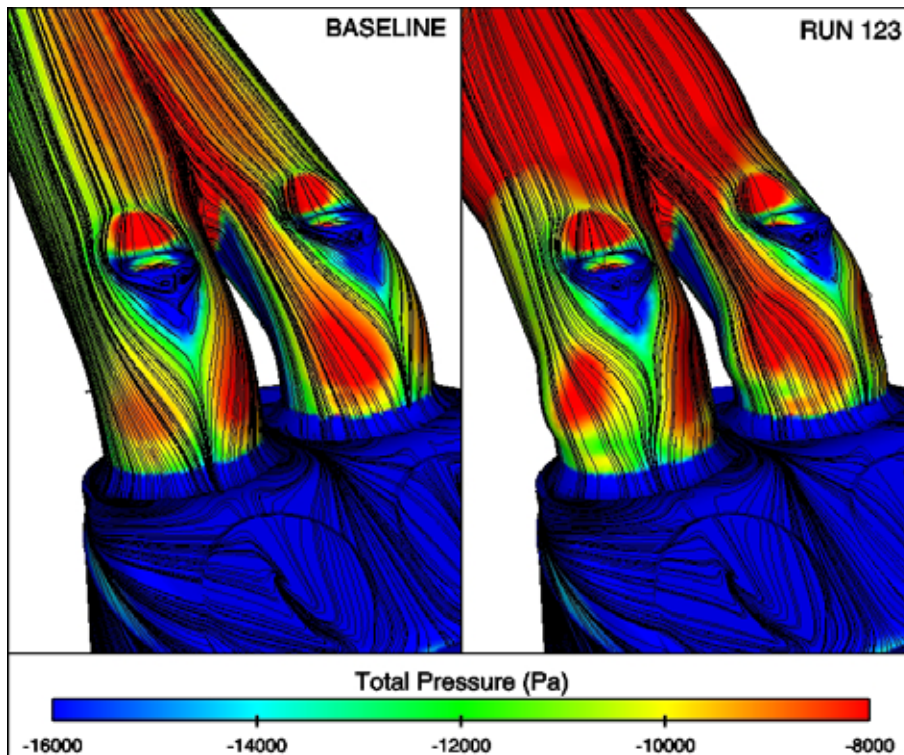


Figure 25 - Comparison of oilflow and total pressure on the surface of the two intake ports



#### 4.5. Time summary

Table 2 shows both the man time and CPU time involved in this phase of the optimisation. The total duration of these stages was less than one week, including solve time (more than one 12xCPU PC-array was used at some stages).

*Table 2 – Time summary*

Time Summary (for all 124 cases)	Man time	CPU time
Working with in-house Latin hypercube code	30 mins	50 mins
Modifying existing scripts to automate case generation	20 mins	-
Export time	-	370 mins
Modifying existing scripts to automate case checking	20 mins	-
Case checking	-	180 mins
Modifying existing scripts to automate journal creation	10 mins	-
Journal creation time	-	2 mins
Modifying existing scripts to automate processing results	10 mins	-
Processing results	-	30 mins
Working with in-house response surface code	60 mins	20mins
<b>Sub-Total</b>	<b>1hr 30 mins</b>	<b>10hrs 52 mins</b>
<b>Sub-Total from Table 1</b>	<b>15hrs</b>	<b>3hrs 25mins</b>
Total solve time (12xCPU PC-array)	-	9300 mins
<b>Total</b>	<b>16hr 30 mins</b>	<b>169hrs 17 mins</b>
Please note that these figures are approximate		

## 5. CONCLUSIONS

The following conclusions can be made about the process used in this study:

- The geometry of a generic automotive intake port has been successfully modified to increase the mass flow for a given pressure drop by 1.94% whilst staying within a +/- 15% cross-sectional area constraint in the runner and port. The entire process was possible in just over a week.
- Sculptor™ has enabled the definition of 11 parameters to deform the case file directly so no re-meshing has been necessary. The process of defining and refining the ASD volume and parameters for this complex problem has taken approximately 2 man days.
- The generation of these cases has been automated using Sculptor™ in batch mode and a series of scripts reducing the man-time taken to produce each subsequent design to a matter of seconds.
- Latin hypercube sampling has been used to define a set of experiments based on the 11 parameters defined. A response surface method was then used to predict a global maxima.
- A total of 124 cases were run to optimise the 11 parameters.
- All experiments were reconverged from the baseline data file reducing the number of iterations required from 4000 to 500 thus cutting the computational requirement by 90%

The results of the optimisation show the following:

- An overall increase in mass flow rate of 1.94% has been possible based on the 11 parameters used in this study. The cross-sectional areas of the runner and port are constrained within the bounds identified at the start of the project (Figure 17)
- The cross-sectional area of the runner has been increased in all directions. The area of the port has been reduced near the centreline of the bore and increased on the opposite side (Figure 21)
- The changes to the runner have reduced the local velocities thus reducing some of the losses.
- The decrease in cross-sectional area around the port has altered the flow local to the surface downstream of the valve stem. With the optimised geometry the surface flow is injected behind the valve stem slightly more rapidly leading to a small reduction in the size of the stem wake.
- It is a complex combination of these 11 parameters, which has led to the performance increase. To make such a gain without using parametric optimisation techniques would have been difficult.

## 6. APPENDIX A

The variation in performance with two interacting parameters is plotted in Figure 26 and Figure 35. As the two parameters may have different limits they are varied between their maximum and minimum values at the same rate. For this reason there is no scale shown on the x-axis of the graph.

Figure 26 to Figure 30 show where the two parameters are both **increased** at the same rate between their limits. Figure 31 to Figure 35 show where one parameter is being **increased** whilst the other is **decreased**.

*Figure 26 Variation of performance with interacting parameter values*

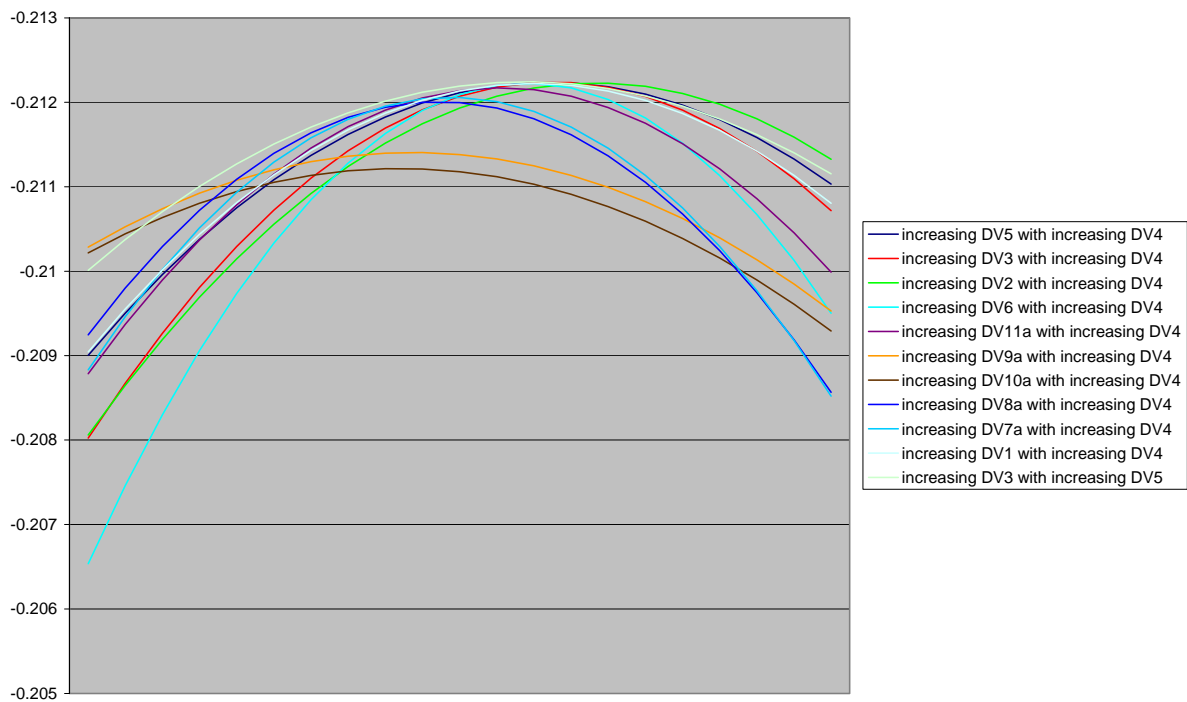


Figure 27 Variation of performance with interacting parameter values

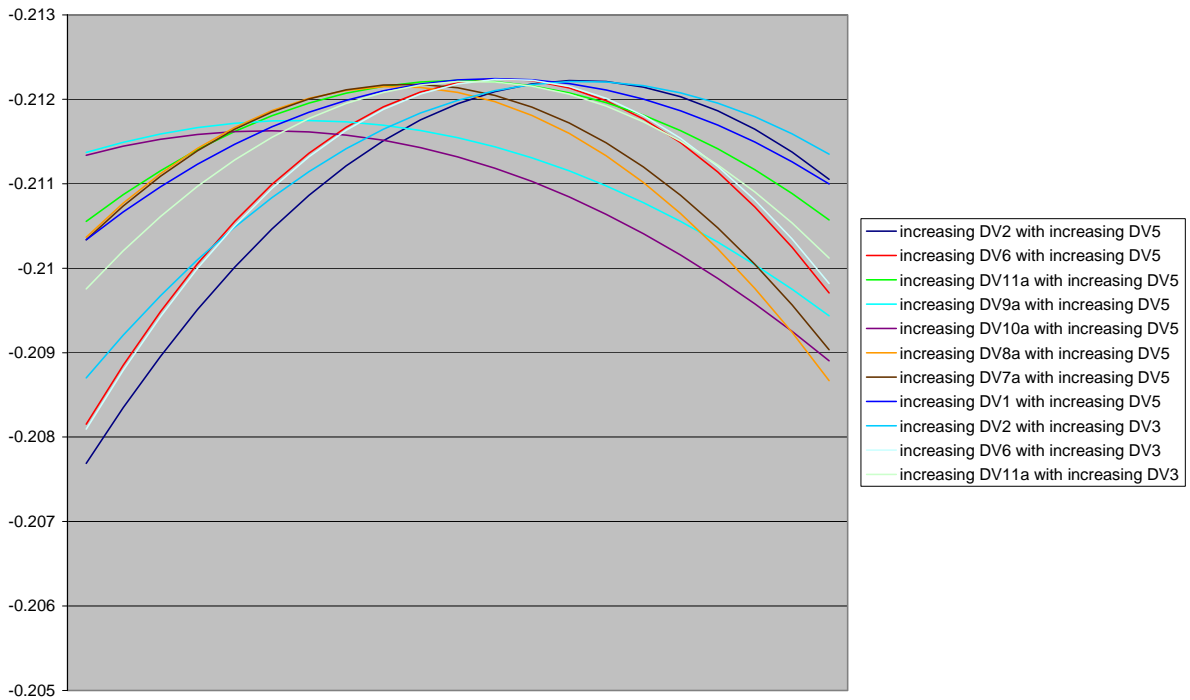


Figure 28 Variation of performance with interacting parameter values

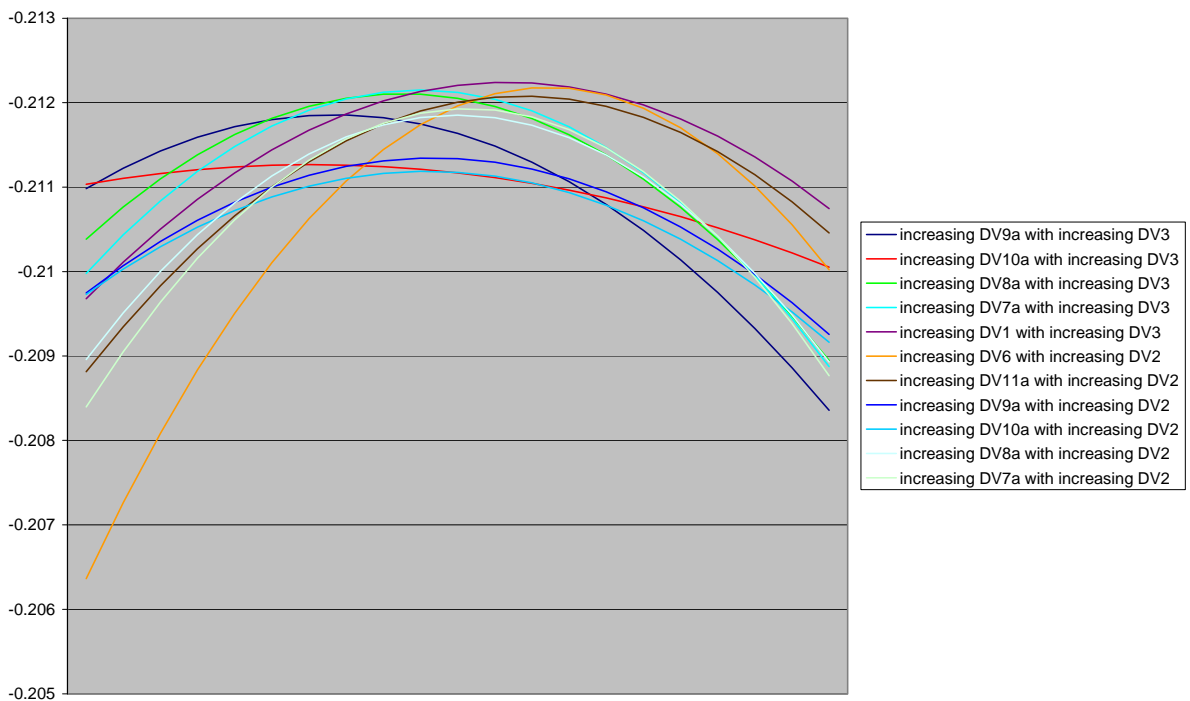


Figure 29 Variation of performance with interacting parameter values

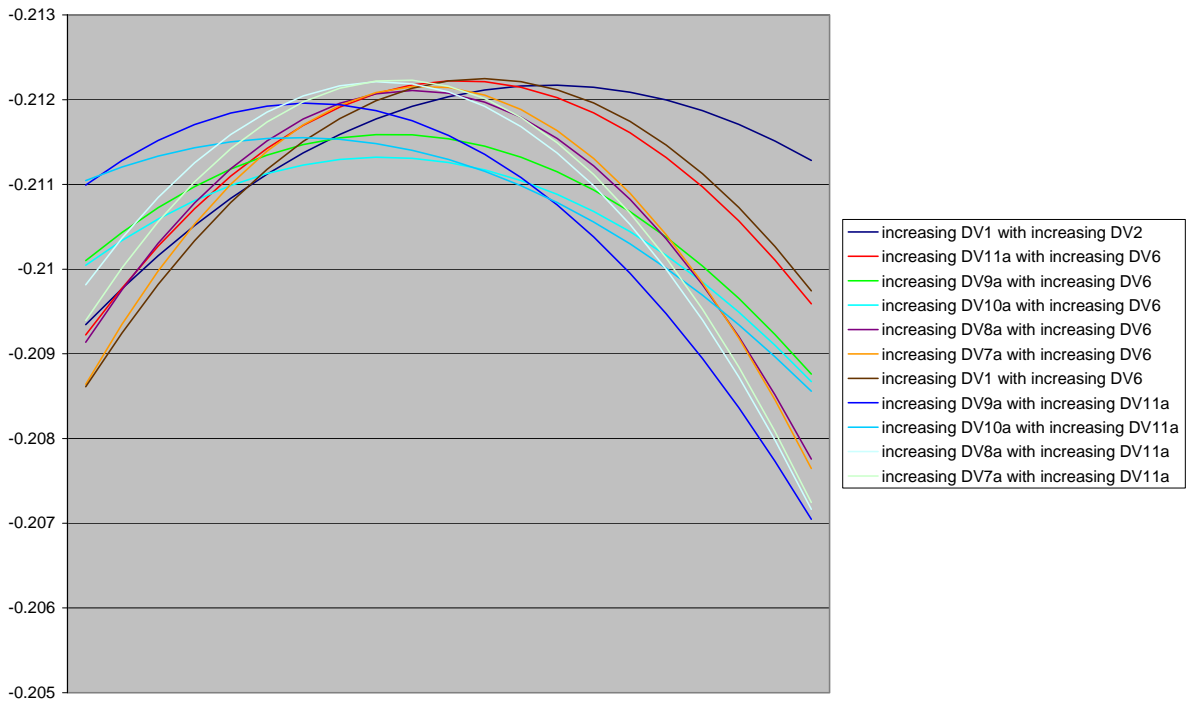


Figure 30 Variation of performance with interacting parameter values

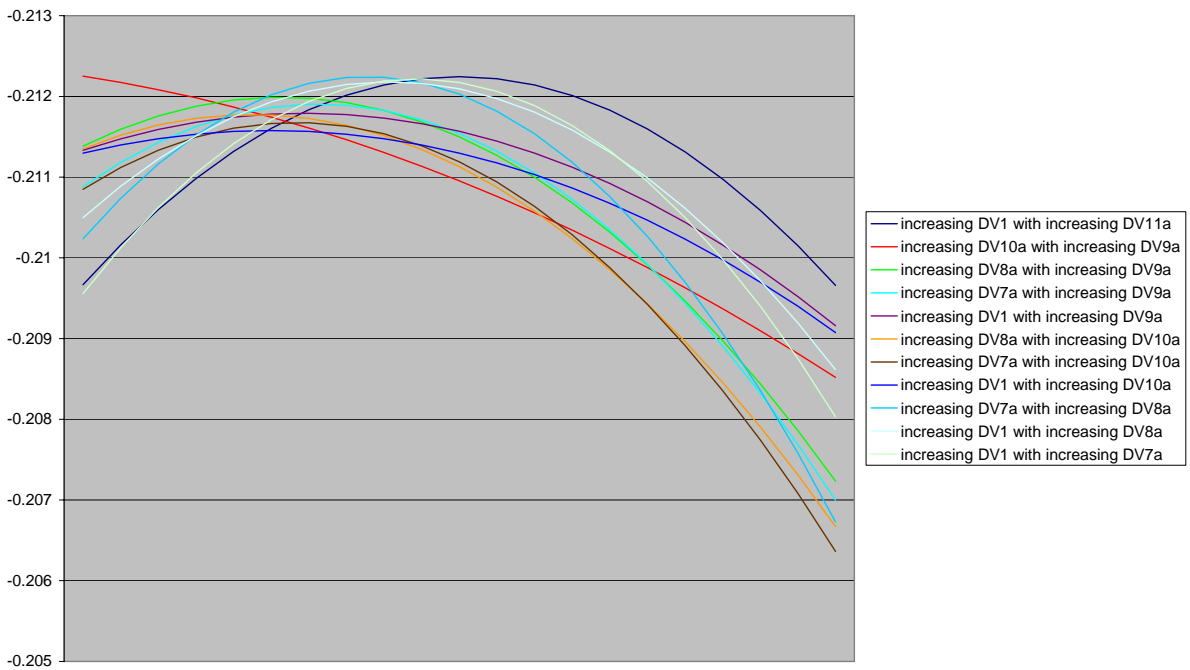


Figure 31 Variation of performance with interacting parameter values

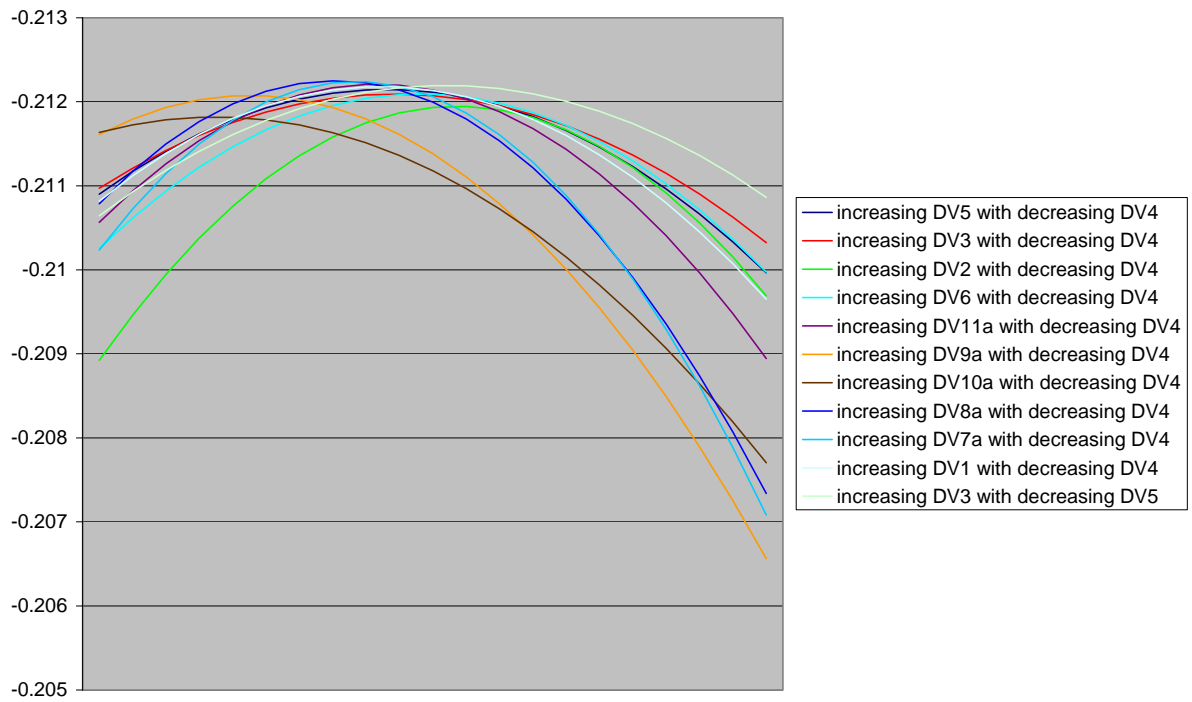


Figure 32 Variation of performance with interacting parameter values

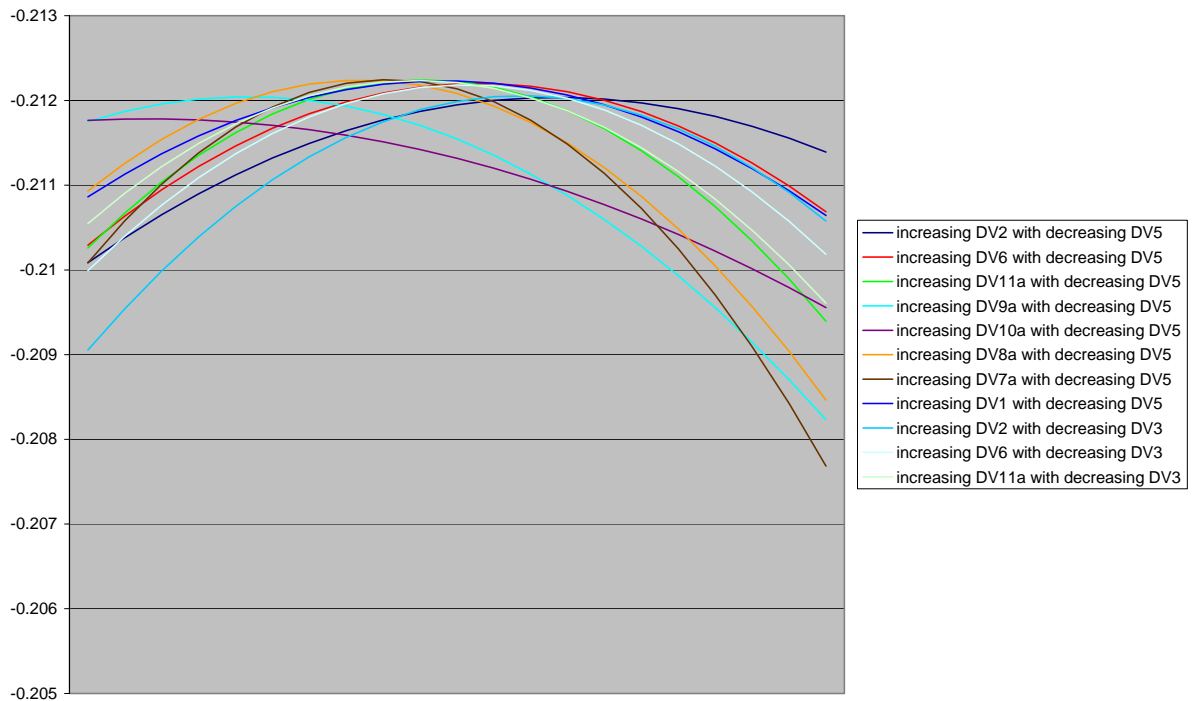


Figure 33 Variation of performance with interacting parameter values

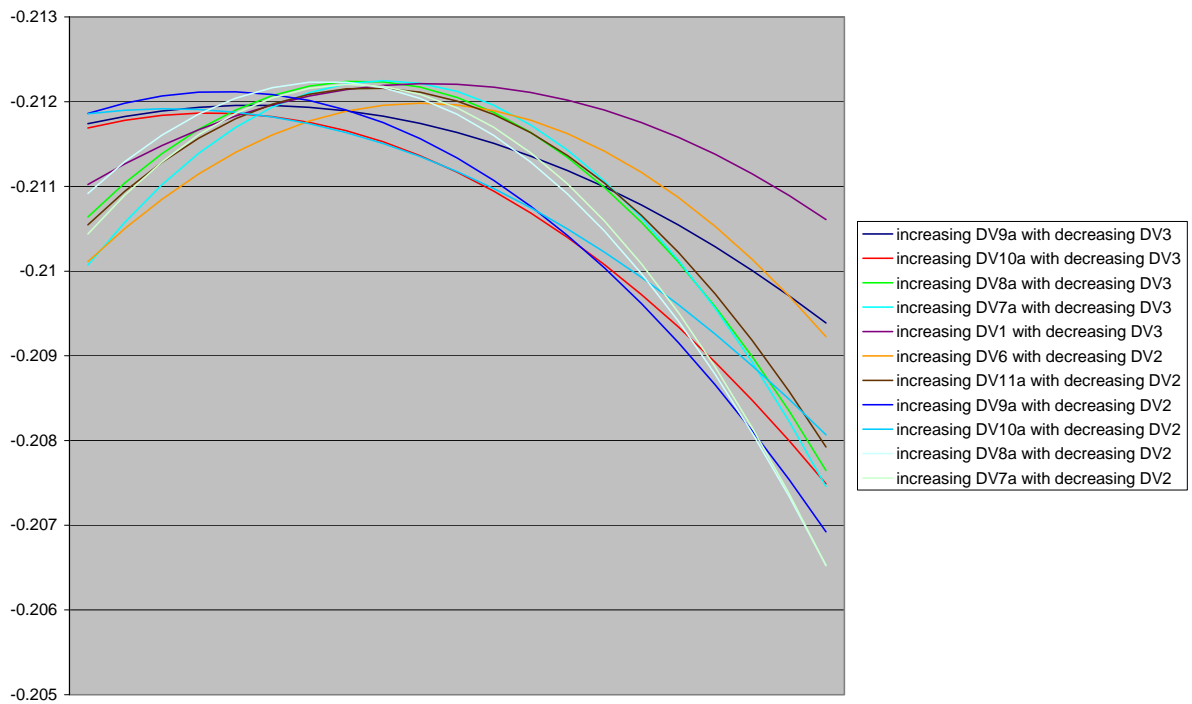


Figure 34 Variation of performance with interacting parameter values

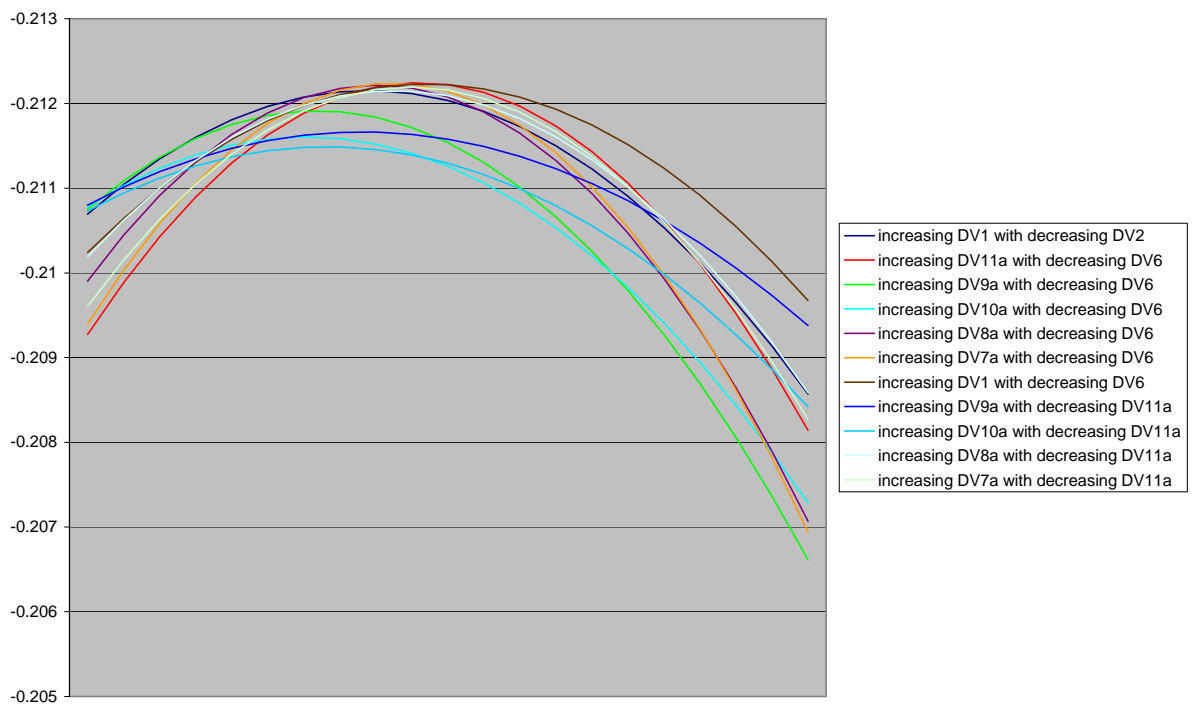


Figure 35 Variation of performance with interacting parameter values

