Design Optimization for Additive Manufacturing in OptiStruct with consideration of Overhang Angle in Topology Optimization

Rob Hoglund and Dr. Fabian Fuerle
Altair Engineering
June 2018

Introduction

Across multiple industries including automotive and aerospace, topology optimization continues to play a crucial role in the design of structural parts for light-weighting and performance gains. In the topology optimization process within OptiStruct, the optimal material distribution of a structure is determined for a given set of boundary conditions and constraints, within a certain design region. However, one of the traditional challenges involved with topology optimization includes manufacturability of the optimized designs using traditional processes. The results of topology optimization, given complete freedom over a set design space, often do not produce parts that can easily be cast or formed, and thus manufacturing constraints introduced into the optimization formulation are necessary to be able to realize the design for production. OptiStruct has offered manufacturing constraints for traditional forms of manufacturing such as casting and extrusion for several years now.

Since additive manufacturing (AM) brings a level of increased design freedom compared to typical casting, machining and stamping processes, it has generated increased interest as a method to manufacture near optimal structures generated by topology optimization. In addition, the consolidation of multiple parts into a single additively manufactured part can save significant tooling and other overhead costs for manufacturers interested in light weighting and reducing total production costs. However, AM brings its own set of design challenges, including the necessity of support structures in part production, thermal distortion, and amount of post-processing required. To accommodate this, overhang angle (OHA) consideration for topology optimization in the form of a full constraint and a more lenient penalty method has been implemented in OptiStruct. Overhang angle (OHA) consideration helps to determine optimal structural topology in a design space while either avoiding all overhanging members or finding a good compromise between structural performance and the need for support structure. This paper gives a technical review and guidelines for positioning the current capabilities. Note that the following uses OptiStruct version v2018. There have been some changes to the discussed algorithms compared to previous versions. Generally, version 2017.2.3 can be used to reproduce all the presented results.

Support structures and metal additive process

In the direct metal laser sintering process (DMLS), a bed of metal powder is scanned by a laser in layers of specific hatch patterns. The metal is melted and then cooled according to the scan path, adhering to first the build plate, and then successive layers of structure beneath. After a pass of the scan path is complete at each layer, the bed is lowered by the layer height (in the order of micrometers) and more metal powder is spread evenly across the bed before the next scan of the laser. As the structure is built up from the powder, the bed is lowered incrementally until the structure is formed.

Metal structures produced by the path of the laser in the powder-bed setup can only be printed up to a certain maximum OHA from the vertical. This angle depends on the machine parameters and material characteristics. This angle is typically about 45º from the normal to the powder bed. At OHA greater than this, the powder, which is melted several layers at a time to cool and form the structure, burns underneath the structure without enough reinforcement underneath to anchor the structure to the build plate. For this reason, the standard practice is to create support structures, which are generated along with the main structure. These help to anchor the part to the base during production and support those areas of the part where the overhang of the structure is greater than the allowable build angle. The generation of support structures underneath overhanging areas not only affixes the structure firmly to the build plate, but also mitigates the effects of thermal distortion due to the heating and cooling incurred by the laser during the melting and cooling process in each layer. Figure 1 shows an example of the regions where support structure is required depending on the angle of the structure with respect to build direction.

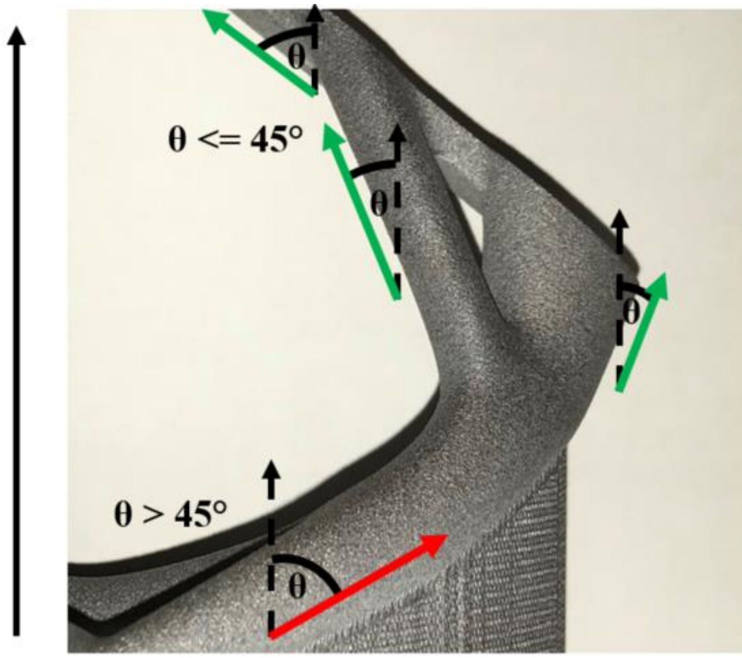

Figure 1: Example of an additive-manufactured part with build direction indicated, showing self-supporting build angles in green, and areas overhanging greater than 45º requiring support structure in red.

Although it is possible to print structures at lower OHA than the commonly acknowledged 45º with AM, the reduced surface quality and necessity of supports for those areas force an increase in material usage and build time to compensate. This is undesirable in many cases, due to the increased effort necessary to remove the support structure from interior regions in the model and reduced surface quality. Also, it is desirable in production-level environments to minimize the amount of support structure necessary not only to eliminate waste and further streamline the manufacturing process, but also to reduce the amount of post-processing time, especially as the scale of production increases.

Topology Optimization considering OHA in OptiStruct

To better guide the design of optimal material layout in structures produced by AM methods, two methods for considering the OHA in topology optimization have been implemented to address this issue in AM. Recently, researchers have used a constraint to force the material to grow from the base plate within a specific angle using a projection method [Gaynor and Guest 2016]. In this method, the design variables can form a solid phase only if they are supported by sufficient elements in a cone underneath them in build direction. The shape of the cone is determined by the chosen OHA. If there is insufficient support, the design variable is reduced to zero. In this sense, the algorithm can only produce parts which are satisfying the constraint without exception. In addition to implementing the constraint following [Gaynor and Guest 2016] in OptiStruct, Altair developed a penalty method, which is capable of allowing some violation of the specified OHA. Here, a trade-off between the structural performance and the OHA consideration is sought. This means that if a member is crucial for the performance it will not be sacrificed to satisfy the OHA. But if an overhanging member is structurally less important it will be replaced by one that satisfies the overhang.

Definition of Angle and Build Direction

Since support structures are a necessary part of the metal AM process, consideration of the build direction is an important step in designing for the AM process during the topology optimization and will have a significant effect on the resulting material distribution of the structure to meet performance requirements. It is important to evaluate the build direction as an influence in the simulation-driven design process, since arbitrarily choosing the build direction for a typical topology-optimized part will result in the necessity of at least some support structures.

In the OptiStruct formulation for topology optimization, the user can request the consideration of the OHA. This requires the definition of a vector to specify the build direction, as well as an OHA, which determines the maximum allowable angle from the build direction that the structure can grow within the design space. The option to turn on OHA consideration is specified with the continuation line OVERHANG on the DTPL card for topology optimization design variables as shown in Figure 2, similar to other manufacturing constraints such as draw or extrusion for cast or extruded parts.

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
	OVERHANG	ANGLE	GID1/X1	Y1	Z1	GID2/X2	Y2	Z2	
		METHOD	STEP/PENFAC	PENSCHE	NONDES	HOLES	ANGTOL	DISTOL	
		SUPPSET							

Figure 2: Continuation card format for the OHA consideration.

In the card, the OHA is specified in field 3, with the direction of the build vector specified in fields 4-9. The build vector can be specified by either two nodes GID1 and GID2 in fields 4 and 7, or X1, Y1, Z1 coordinates in fields 4-6 and X2, Y2, Z2 coordinates in fields 7-9. The build vector is defined in the direction from point 1 to point 2. An example OHA definition allowing the structure to grow out an allowable 45º from the build vector in the z-direction is shown below. The x-y plane is the build plane in this case, since the vector is defined by the Cartesian coordinates (0,0,0) and (0,0,1).

```
+       OVERHANG45.0    0.0     0.0     0.0     0.0     0.0     1.0
```

Likewise, an OHA of 45 º with the build vector defined in the direction from node 45 to node 60 is shown below.

```
+       OVERHANG45.0    45                      60
```

The vector from node/point 1 to point 2 always defines the build plane in the direction from point 1 to point 2 as shown in Figure 3. The OHA constraint will force the structure to grow out from this plane within the allowable angle. Specifying a larger angle here will allow more design freedom in the optimization.

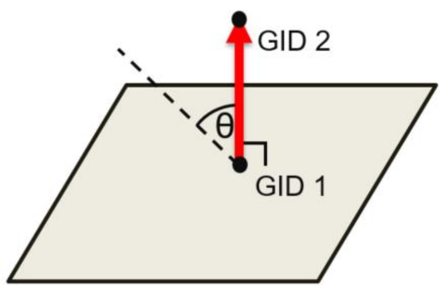

Figure 3: Definition of the build direction vector and OHA definition.

In Figure 4, the degree of overhang is reduced from 45º to 30º, and the resulting optimized structures are shown. Modifying the OHA allowed has significant effects on the material distribution and final part performance. The trade-off between performance and manufacturability should be made with full knowledge of the AM process machine parameters, reliable build angle, and desired part performance.

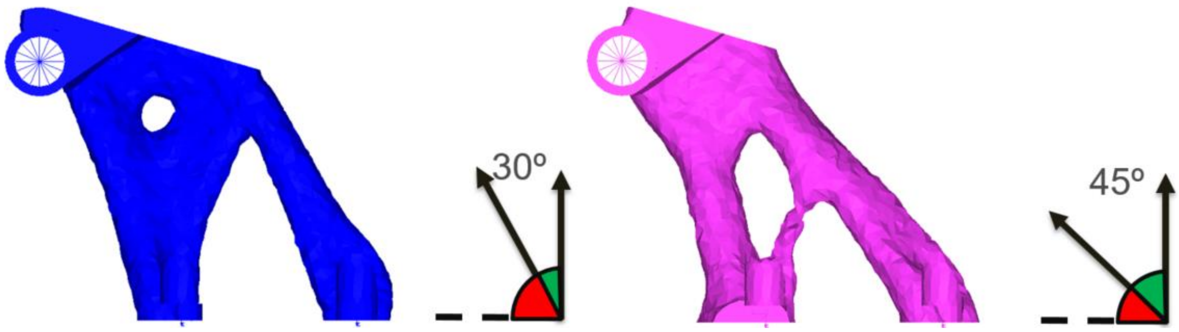

Figure 4: Example of the result obtained by varying allowable OHA parameter.

Methods and Main Options

In the second continuation line on the card, with the METHOD field in entry 2 the user makes an important decision and chooses between the OHA constraint (CONSTR) and the penalty method (PENAL). Field 3 defines the sub-method. The CONSTR option, which is the default algorithm, activates the constraint, which forces the OHA constraint to be satisfied for each element in the design space. There are two options for the STEP parameter in this field: 1 or 2. Option 1 uses more aggressive move-limits in the optimization, whereas option 2 is more conservative in this regard.

The PENAL option, activates the penalty approach, which allows some areas of the model to violate the OHA. The amount of violation can be controlled via field 4. For the penalty method this parameter is PENFAC which indicates how much the objective is penalized and thus if the final design will lean more towards structural performance or more towards a support free design. The four possible choices are LOW, MED, HIGH, and ULT. As the penalization factor on the objective function is increased from LOW to ULTRA, the penalization on the overhanging areas in the objective function is increased. Thus, a penalization method run with the LOW penalty will have lower penalization for overhanging areas, while a HIGH or ULTRA penalization would introduce a much higher penalty to the objective function for allowing those areas in the design space.

The two algorithms for the OHA consideration are intended to give the user flexibility for the particular use-case, because the full constraint can have a big impact on the objective function. Figure 5 shows an example to illustrate the difference between the full constraint and the penalty method. The constraint will remove the need for any support structure, no matter the cost in structural performance. The penalty method however, will avoid more and more support structure depending on the defined penalty factor from LOW to ULTRA. It can also be seen, that even with the highest penalty factor, it is not possible to remove all support structure. If that is desired, it is necessary to use the full constraint.

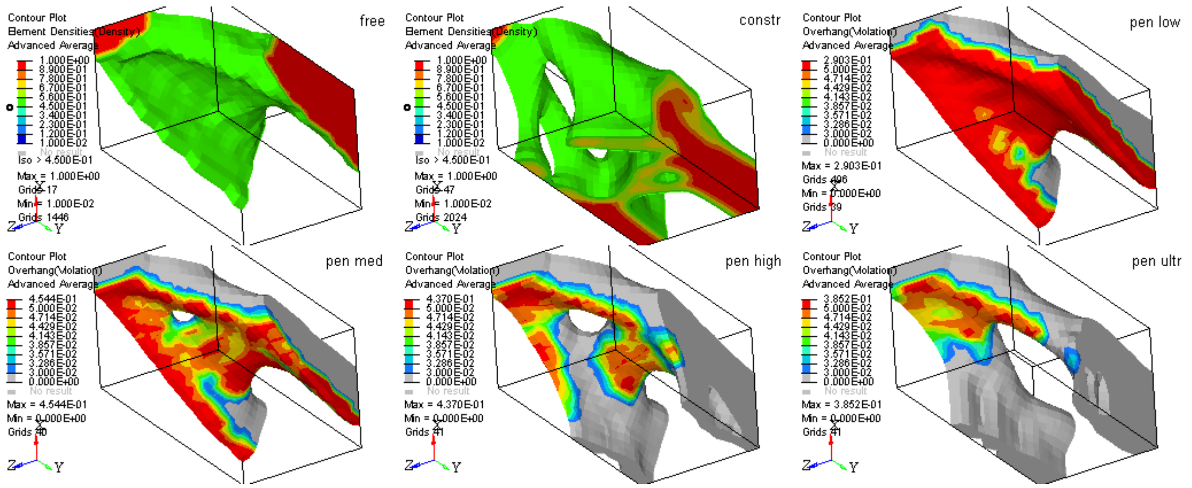

Figure 5: Optimal topologies for free run, for full constraint and for penalty method runs with increasing penalty factors (LOW, MED, HIGH, ULTRA).

As can be seen in Figure 6, the objective functions of the penalty runs lie in between the free run and the full constraint and increase with higher penalty factor. This example highlights the ideal use of the penalty method, because here the difference between constraint and free run is large, both in terms of the look of the final structure and also in the compliance. In these cases, the penalty method can interpolate between these two.

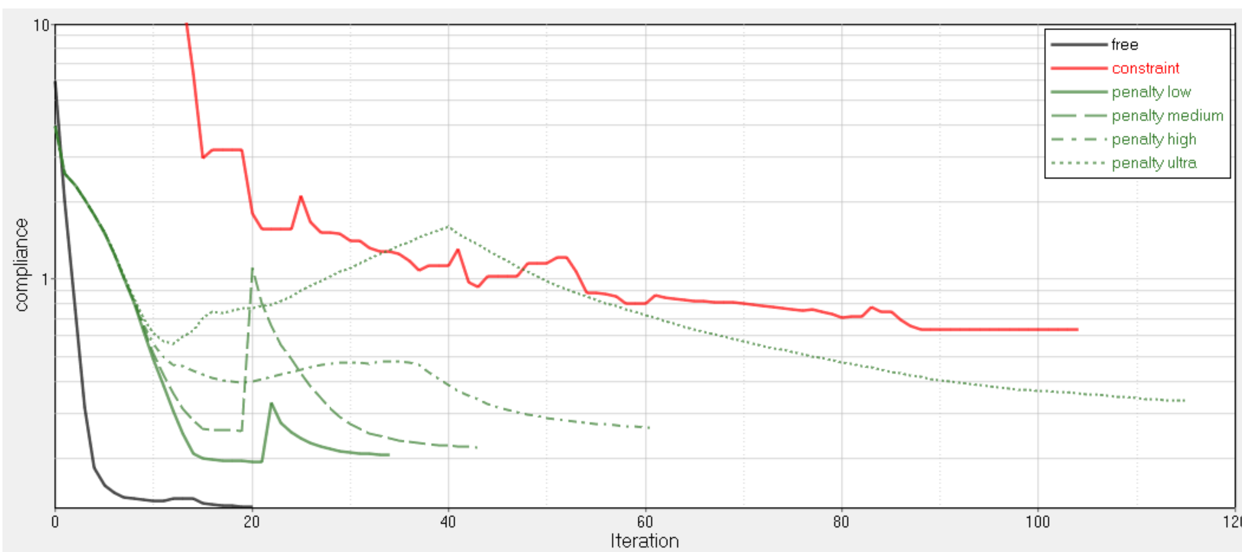

Figure 6: Development of compliance throughout optimization for free run, for full constraint and for penalty method runs with increasing penalty factors (LOW, MED, HIGH, ULTRA).

To view areas more likely to require support for the given OHA using the penalization method, the result type Overhang (Violation) is included in the output h3d file. The values shown represent the degree to which the specific areas of the design material would be requiring some amount of penalization and might not be supported. This result is 0.0, if there is either no support or no material. It is 1.0 if there is solid material and the element is not supported. This tool can be used to predict where support structures might still be required for building the part with AM. Note that this output is not available for the CONSTR method, as no violation is possible.

Penalty Scheme

The PENSCHE in field 5 indicates the penalization scheme used for computing the elemental stiffness with the overhang method. The SIMP or RAMP method can be selected. In standard topology optimization, the SIMP method is the default method used, which penalizes the stiffness based on a power-law scheme from 0 to 1 to force them towards a discrete solution. The SIMP method is used as the default method for the PENAL method. The RAMP method, an alternative approach to calculating the elemental stiffness values, is set as the default penalization scheme for the CONSTR method due to its non-zero sensitivities when the densities are zero. If the results with the default penalization scheme are not satisfactory, the user is encouraged to try the second, non-default option.

Predefined support

Field 6 and 7 indicate whether the non-design space and holes in the structure are treated as supporting by the optimizer (options SUPP or UNSUPP – see card design in Figure 2). If the non-design space or holes are supporting, then they are treated as though they are part of the build plate and structure can grow upwards from these areas, even if the non-design space is not supported underneath. Note, that this does not necessarily mean that the non-design areas will be fully supported underneath by the material in the design space. The process to achieve this is described in the next section. Figure 7 shows a comparison of the results between supporting and not supporting the non-design space in an overhang-constrained topology optimization where the allowed OHA is set to be 45º from the positive z-direction.

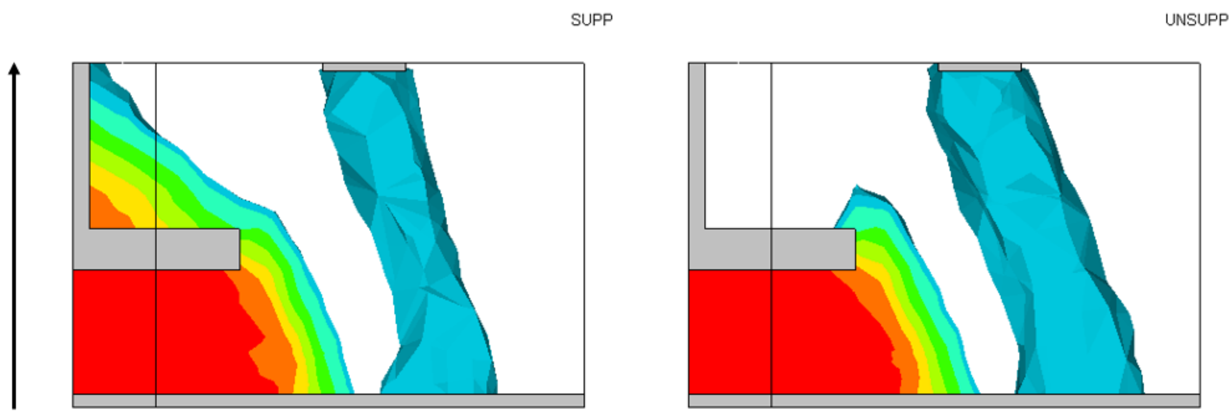

Figure 7: Example allowing the non-design space shown in grey with support (on the left) and without support (on the right), with the build direction in the z-axis as indicated.

The addition of the SUPP/UNSUPP parameter determines whether the non-design space can grow structure in the direction of the build vector and as a result, has a significant impact on the optimization result. Non-design space is supporting by default for the OHA algorithms, whereas holes are not considered supporting by default if the parameter is not specified. When support is specified (SUPP), the holes are treated as self-supporting structures from which material can grow and when they are not considered supporting (UNSUPP), the OHA must be satisfied. Figure 8 shows a simple example of an optimization with and without holes supporting the design space. Note here that physical holes are used for the purpose of illustration, but this would also hold true for any areas of the design space that are located over empty space.

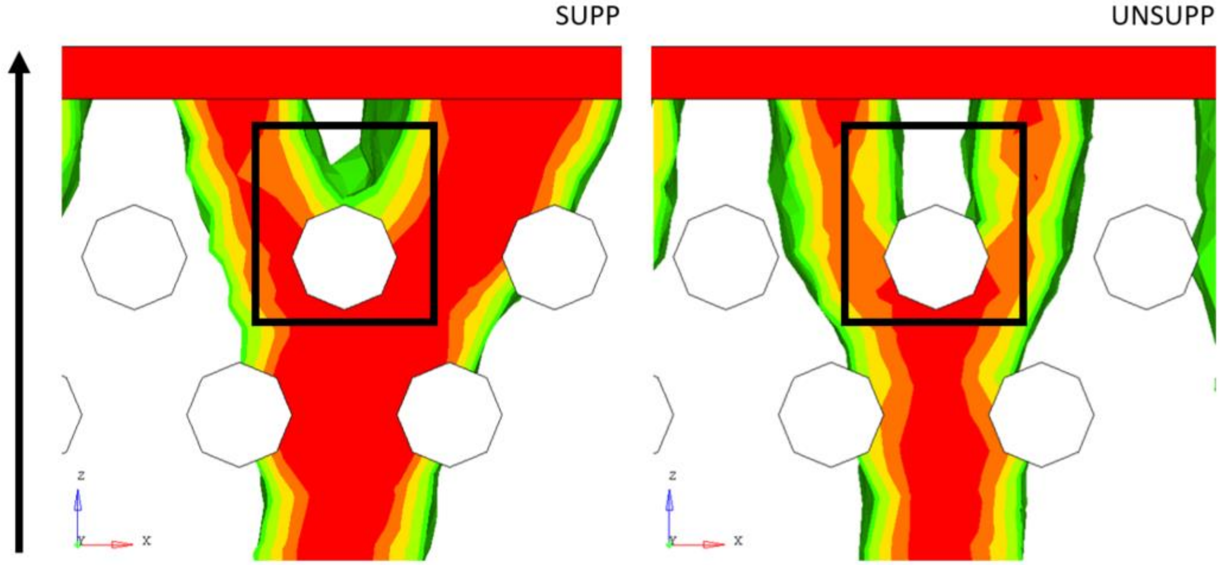

Figure 8: Comparison of OHA constraint optimization result with holes supporting (on left) and not supporting (on right), build direction vector in the position-Z direction.

Additional options with regards to predefined support include the definition of the two fields, ANGTOL and DISTOL (see card design in Figure 2), which can be used to define which elements in the first layer of the design space are considered supported in the model, i.e. what parts of the model, material can grow out of. ANGTOL can take on any value from 0 to 90º perpendicular to the build plane. By default, only the first layer of elements of the surface of the model encountered in the build direction are considered to be supported (ANGTOL = 90.0). For ANGTOL lower than 90.0, also an inclined lowest surface of the design space can be considered supported, as long as these elements are below the distance DISTOL from the build plate. If DISTOL = 0.0, as is default, only the first layer of elements is supported. Figure 9 shows a brief diagram explaining which elements are considered supported in the design optimization space.

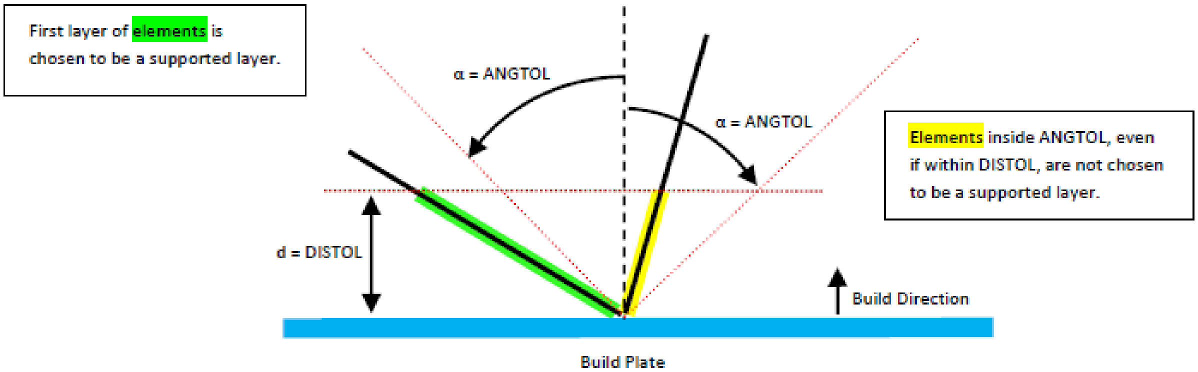

Figure 9: Determining which elements are supported in OHA optimization as they are encountered in the first layer of the structure.

An example of the ANGTOL and DISTOL parameters is shown below for a curved or inclined surface. The result type in the h3d output, Overhang (Predefined Support), can be used to visualize the supported elements as shown in Figure 10. The supported elements are shown in red and green, and the elements considered not supported and governed by the OHA method are in blue. In this case, since the non-design space was also supported, the elements greater than 45° from the build plane are supported.

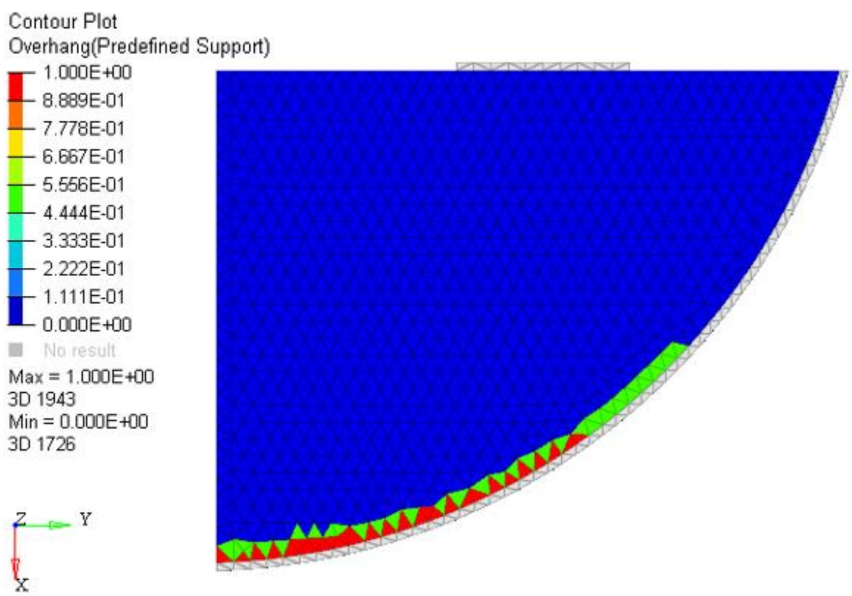

Figure 10: Output of Overhang (Predefined support). In this case, ANGTOL < 45° from the build plane are supported.

An alternative to ANGTOL and DISTOL is to define a grid set which indicates regions in the model that shall be considered as supported. This set can be referenced in the field SUPPSET on the third OVERHANG line (see card design in Figure 2). As an example of the OVERHANG line with medium penalization applied, non-design space supported, and no holes supported using the SIMP method might look like this:

```
+       OVERHANG45.0    0.0     0.0     0.0     0.0     0.0     1.0
+               PENAL   MED     SIMP    SUPP   UNSUPP
```

Ensuring support of non-design regions

To enforce the support of non-design spaces through material in design spaces the following process is recommended: the non-design space is included in the design-space and an additional volume fraction constraint is added for the region which is intended as non-design space. This volume fraction is set to 1.0 which ensures that the region is filled with material (making it a de-facto non-design region) while the OHA consideration disallows this region from being unsupported or penalizes the objective if that is not the case, depending on which algorithm is chosen. An example of this is shown in Figure 11. On the left, the design is completely supported building up from the base by including the component highlighted in blue in the design space with a volume fraction of 1.0. On the right, the model does not include a separate constraint for the highlighted region, and consequently would require support structures in AM for some of the overhanging areas.

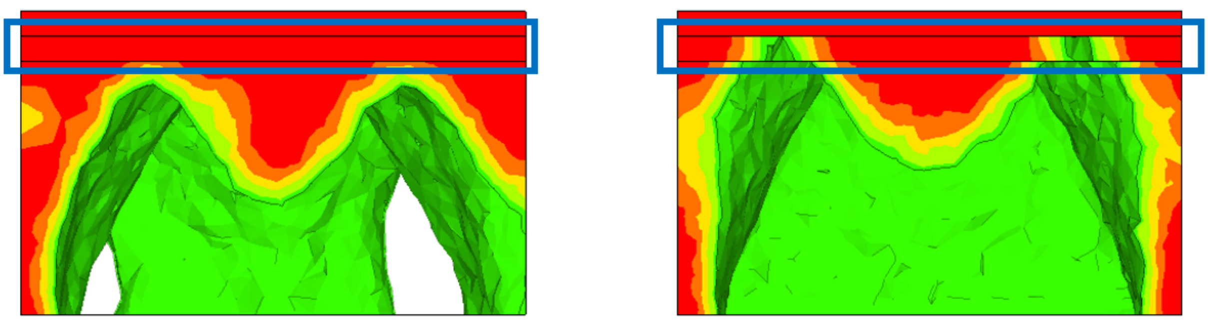

Figure 11: Example of using a "non-design" space in the topology optimization with a volume fraction constraint of 1.0 using OHA.

Example

To demonstrate the effect of considering OHA in topology optimization for AM, a simple aerospace test bracket is evaluated while applying the OHA constraint with several different build directions. This bracket consists of several forces applied at a large central pin location, and a set of six bolts where single point constraints are applied. This bracket is optimized with the six build directions shown in Figure 12 with a minimum mass formulation while constraining some displacements. The structures are optimized assuming an OHA requirement of 45º from the normal vector. Note, that predefined support has been adjusted for each orientation to allow for a reasonable area to grow material from. The results of the different optimization runs show vastly different structures particularly in the angle of the rear member. As the build direction rotates clock-wise, the rear member can become flatter.

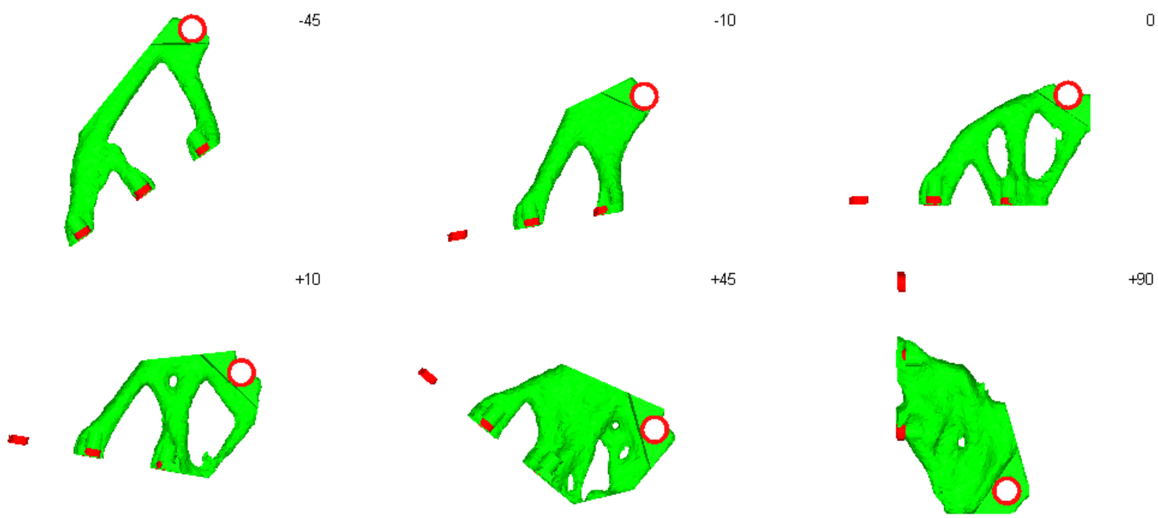

Figure 12: A bracket structure optimized with varying the build orientation at angles of -45, -10, 0, 10, 45, 90, from normal (all shown with the build orientation in the vertical direction)

In addition, the same bracket is evaluated using the two methods for OHA consideration (constraint and penalty approach) as well as some of the additional options. The resulting material distribution of the free run is compared to the constraint with RAMP and SIMP as well as STEP=1 and STEP=2 in Figure 13 below.

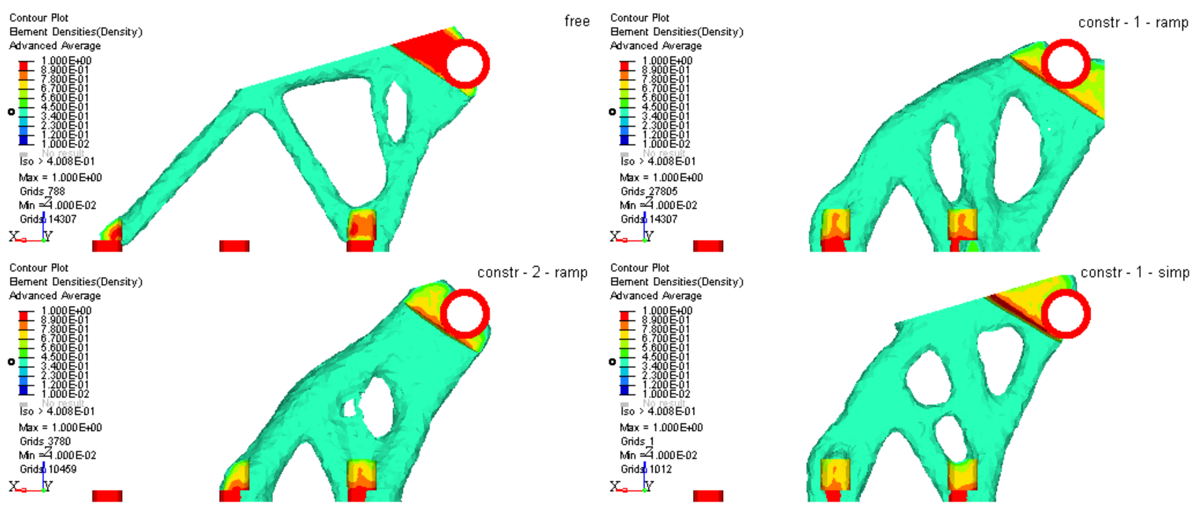

Figure 13: A comparison of the optimum material distribution of the free run and the full constraint with various options (RAMP, SIMP, STEP=1, STEP=2).

The penalty method with increasing penalty factor is compared to the free run in Figure 14. Note, that only LOW, MED and HIGH are plotted here as HIGH already exhibits some numerical issues and increasing it further would only worsen this. As discussed before, the penalty method can remove some structurally less important members but struggles to remove all overhanging members. Increasing the penalty factor further will lead to numerical issues as can be seen here. Note also, that the number of regions that require support are not drastically reduced in this example using the penalty method compared to the free run. As can be seen in Figure 13, the general outline of the design is similar between the free run and the constraint run. The penalty method will work much better if the design differs a lot as is the case in the example shown in Figure 5. In that case, the algorithm can better interpolate between the two.

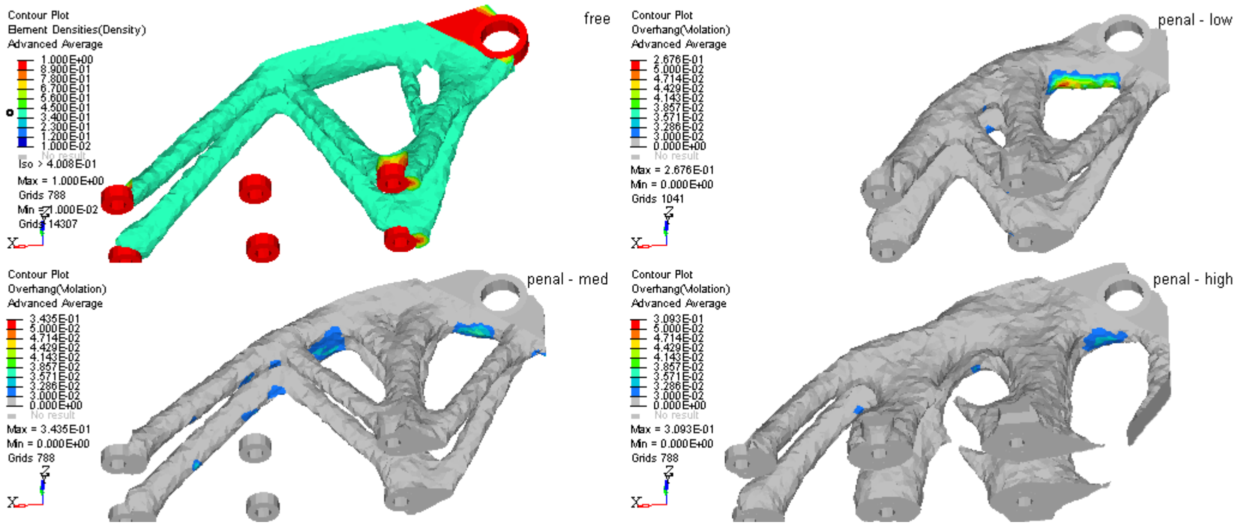

Figure 14: A comparison of the optimum material distribution of the free run and the penalty approach with increasing penalty factor (LOW, MED, HIGH). In this case, LOW penalization gave the best result and HIGH led to numerical issues.

Figure 15 shows the objective function plotted against the iterations for the different methods and options mentioned above. Note that the structural compliance depends significantly on the amount of penalization or whether the full constraint is applied in the optimization formulation. Introducing some amount of penalization to the structure should provide a lower minimum mass than the fully constrained OHA, while some members will still require support when the penalty approach is used. As can be seen in this example, applying more and more penalization can lead to numerical issues and results that are worse than the full constraint. If the user would like to remove more overhanging regions than the penalty method is capable of removing, the full constraint should be used. In this situation, the HIGH penalization option resulted in a much higher mass than the fully constrained algorithm since the amount of penalization for removing the members was too high. By enabling a LOW or MED penalization, a better result was obtained in this case. With regards to the full constraint, STEP=1 and 2 were tested in this example. In this case, STEP=2 yielded superior results to STEP=1 (the default). For STEP=1, the SIMP penalization scheme was also tested for the full constraint which yielded a similar mass as RAMP with STEP=2 while requiring far less iterations. This shows that it can be beneficial to try these additional options too.

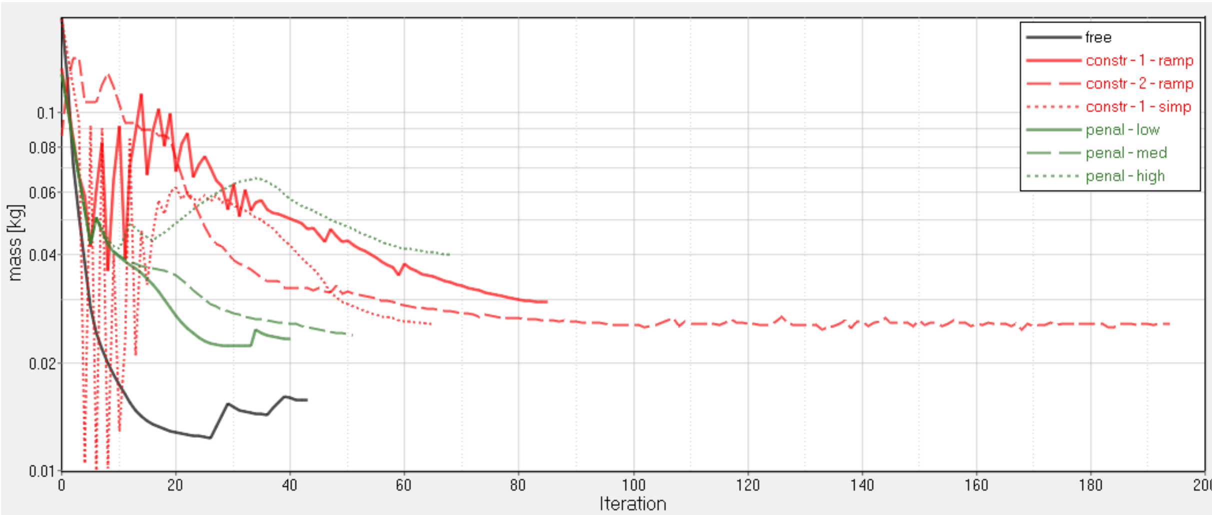

Figure 15: Convergence history for the different algorithms with OHA consideration.

As is clear from the plot of convergence history for the different overhang algorithms, the full constraint method for STEP=1 contains jumps in the objective function when material is added and removed from members to satisfy the OHA constraint strictly. A removal of material can lead to members suddenly being unsupported which in turn leads to their removal and so forth. This potential domino effect can lead to fluctuations in the objective function. The constraint method with STEP=2 applies an automatic step size control, which is able to smooth the jumps in objective function value, but comes at the cost of an increased number of iterations to converge. The penalization method adds a penalty to the objective function for overhanging structures which leads to a smooth objective, because members are not removed if they are not self-supporting. The reason the objective function for the penalty method increases in the beginning of the optimization is due to the fact that the penalization is gradually increased which leads to numerical stability and better results.

Guidelines for OHA consideration in OptiStruct

As can be concluded from the discussions above, considering OHA in topology optimization requires some knowledge and thought from the user. The following guidelines are designed to help the user to effectively and successfully run topology optimizations while considering OHA.

1. As a first step the user is encouraged to run a topology optimization without any OHA consideration. From the result one can estimate the impact the consideration of the OHA might have. Also, it might be possible to see what the best build direction might be, i.e. a build direction that would cause the smallest impact on the performance.
2. Before running the optimization while considering the OHA, the user should make sure that sufficient regions are considered supported, i.e. regions from which material can grow without being seen as a violation of the OHA consideration. To this end, the user should run a check run with the OVERHANG line defined and review the Overhang

(Predefined support) which is output to the h3d file. If the contour is not satisfactory, various options have been described in this paper to mitigate this.

3. Once the predefined support looks satisfactory, the user is encouraged to run the full constraint first to see if the impact of the constraint on the performance is in fact acceptable as this would mean a support free structure.
4. The user is further encouraged to try different options for the full constraint such as PENSCH=SIMP and/or STEP=2 to attempt to improve the outcome.
5. If the reduction in performance is too large compared to the unconstrained case, the user may utilize the penalization method, by starting with a small amount of penalization and increasing as necessary. This way, one can investigate the trade-off between a structurally better design and one that is easier and faster to manufacture. The user should keep in mind that too much penalization can cause numerical issues and lead to bad results. It should also be kept in mind, that the penalty method typically has more success if the difference between the design from the free run and the full constraint differ significantly. If they are similar, the penalty method will struggle to find a design that lies “in between”.

Concluding Remarks

In this white paper, the OHA formulations in OptiStruct have been reviewed and several test cases examined. Since the OHA consideration is an additional constraint added to the optimization problem, the structural performance or mass of the part will be impacted compared to that obtained without any manufacturing constraints applied. The CONSTR parameter employs a full constraint, while the PENAL method introduces varying levels of penalization to the objective function to reduce the violating overhanging areas. It is up to the user to determine whether some areas of the design space are allowed some degree of overhang violation (and more support structures to manufacture). A CAE analyst would benefit from understanding the difference in algorithms explained previously as well as the additional options offered, since these options significantly affect final part performance and determine amount of support necessary to print the parts. The OHA options discussed are supported from OptiStruct v2017.2.3 with a few updates for v2018. Only the full constraint when the SIMP penalty scheme is used will produce better results in v2018. Furthermore, the ALM Support(Support) and ALM Support(Predefined Support) have been renamed to Support(Violation) and Support(Predefined Support), respectively. Lastly, the SUPPSET field for additional support defined through a GRID set was added in v2018.

The best setup for the overhang consideration will depend on factors related to the individual process setup, for instance, the volume of parts produced, amount of post processing required, and complexity of the final geometry. For different AM methods and material attributes, the allowable OHA can also vary, as some metal powders such as titanium are able to be printed at a higher angle from the build direction than others (such as aluminum). In the case where the allowable OHA is increased relative to the normal from the build plane, the objective function may decrease as the strictness of the constraint is relaxed accordingly. In any case, the choice of the build direction is also a crucial decision and determines the impact of the overhang consideration from the start. The OHA consideration for topology optimization has also been included in Altair’s

Inspire package, beginning with the 2018 version. Within Inspire, concepts can be investigated quickly with the OHA being considered on a design space and specifying an allowable angle and build plane. Two methods for the constraint, Strict and Lenient, are available for the overhang algorithm (using constraint method 1 and penalization method HIGH). Designers using Inspire will find it quicker to generate designs for AM especially with the inclusion of the PolyNURBs tool in the Inspire package for realization of the part geometry. Customers can also import OptiStruct result data back into Inspire and trace over PolyNURB geometry for smoothed results realization for AM.

In conclusion, the OHA consideration for topology optimization is a newly implemented feature within OptiStruct which gives the user an option to remove or reduce overhanging areas in the design space such that they can be more easily produced by AM technologies. These algorithms require some thought and effort on the part of the user, but offers significant potential into optimizing structures for an allowable build angle. We can offer this solution in OptiStruct as a competitive optimization tool for Altair's customers who are currently investigating AM software workflow development and methods for their AM processes.

References

Gaynor A.T. and Guest J.K. (2016). Topology optimization considering overhang constraints: Eliminating sacrificial support material in additive manufacturing through design. Structural and Multidisciplinary Optimization 54(5):1157-1172.