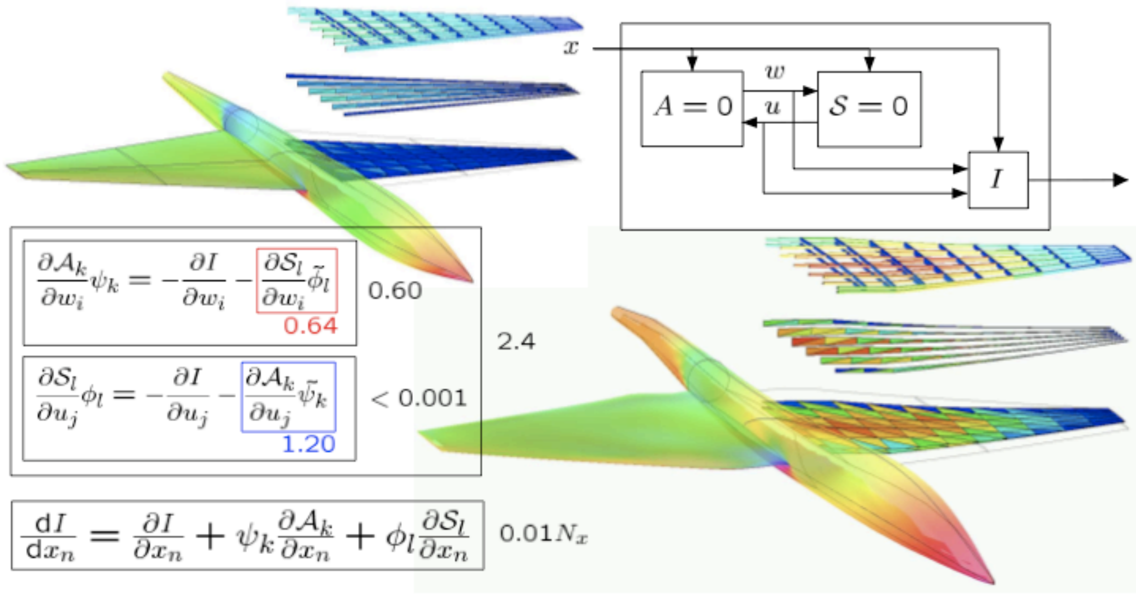
AA222: Trust Regions and Surrogate Models



AA222

Lecture 14

May 15, 2008

Introduction

It is sometimes helpful to construct a fit to an underlying objective function and constraints, and then conduct optimization on the fits rather than the actual functions. This has several advantages:

1. The computations of the actual function or simulation code can be done in parallel and the fit stored for later use.
2. The fit may provide smoothing for noisy functions and yield a reasonable, smooth, global representation of the results
3. One can fit objective and constraints separately, store the results, and later perform additional optimization studies (with different constraint values or variable bounds) without additional function evaluations. This is particularly good for sensitivity studies.

Trust-Region Surrogate Management

It is rare, though, that the initial fits will provide sufficient accuracy, particularly in the region of greatest interest, which may not be known at the start of the optimization work. It is useful to refine the fit in promising regions as the optimization proceeds. This may be done in several ways, but one particularly useful approach is described by Alexandrov and Dennis. It is generally known as Trust-Region-Based Surrogate Model Management.

The basic algorithm is as follows:

1. A region of the design space is selected (trust region). This may be specified with a center point and a radius or with minimum and maximum values for each design variable.

2. The function is sampled in the selected region. Sample points may be

selected using approaches described in Design of Experiments literature, or may be sampled randomly, or from a simple stencil.

3. A fit (or model or surrogate) is created based on the sampled points. This fit is often a simple quadratic regression, but could also be a least squares fit to over-sampled data, or a Kriging fit that includes previous sample points as well as new points.

4. A search method is used to find the minimum value of the model in the selected region.

5. The actual function (not the fit) is then evaluated at the predicted best point found from the optimization in the previous step.

   So far, this is just optimization of a fit. But the key idea of surrogate model management is to iteratively improve the fit, and the next steps are the interesting part of this method.

6. The trust region is redefined as follows. We center the new trust region about the best point that has been actually evaluated, and then we shrink or grow the extent of the region based on a heuristic rule. Studies have shown that the rule described for trust-region-based gradient optimization works reasonably well here, namely:

   - If $\tilde{f}(x_k)$ is the value of the fit at the predicted best point, x_k, and $f(x_k)$ the the actual function value at this point, then we construct the ratio of actual to predicted improvement:

   $$r = \frac{f(x_{k-1}) - f(x_k)}{f(x_{k-1}) - \tilde{f}(x_k)}$$

     where x_{k-1} is the location of the previous best point.
   - If the value of r is less than 0.25, then we did not do nearly as well as the fit predicted we would. Clearly the fit is not very accurate and

we need to reduce the size of the region, so that our fit does a better job. We set the value of the growth parameter, $h = h_{shrink}$.

- If the value of r is greater than 0.75, the fit has done a pretty good job. We give the region more room to move by expanding the size of the region: $h = h_{grow}$.
- If the value of r is between 0.25 and 0.75, the region is OK, but not doing a great job, so the region size is not changed: $h = 1$.

7. We now grow the size of the trust region by the factor h, which redefines the bounds. Typical values include: $h_{shrink} \approx 0.25$ and $h_{grow} \approx 2.0$.

8. We repeat steps 5-7 until the region is sufficiently small or we run out of time.

   One of the interesting aspects of this algorithm, is that although it seems very simple and heuristic, it has been proven (Torszon and Dennis) that the algorithm converges. The proof is rather weak in that it only says that the number of steps is not infinite and in practice the algorithm is not very efficient (especially when the actual function does not resemble the fit), but it is quite robust and is used in many applications.

Incredibly Simple Iterative Sampling (ISIS)

Many different forms of optimization based on the trust-region model management idea have been developed. These involve different types of fits used for the surrogate model and different optimization methods for finding the predicted minimum of the fit. One particularly simple choice for the underlying fit is an interpolatory method that is monotonic. Interpolatory monotonic fits pass through the given points and do not create any extremum points that are not at the sample points – so they do not overshoot. A monotonic spline is one example. Linear interpolation is another. In fact, in the ISIS algorithm for optimization, we do not even need to specify what fit we are using, only that we can imagine that there is an interpolatory and monotonic fit.

Here is the basic idea: If the fit is monotonic, we know that optimization of the fit cannot yield a better point than has already been sampled. (That's what monotonic implies.) So the optimization step is particularly simple: we just pick the best point in the sample and that is the predicted new best point. And if that is what we are going to find, there is actually no need to fit the function at all. We can then go through all of the steps of the trust region algorithm, but we can essentially skip steps 3, 4, and 5. The value of r in step 6 is always 1.0, so the growth factor is chosen a bit differently: If the new best point is the same as the old best point, we do not move the trust region, but we shrink it by $h \approx 0.5$. If the new best point is better than the last one we re-center the trust region and grow it by $h \approx 1.5$. That's it.

As expected, ISIS is not terribly efficient, but it is quite reliable, provably convergent, and incredibly simple.