Sparse Grids for Anisotropic Problems (Long Version)

John Burkardt (ARC/ICAM)
Virginia Tech

.....

Sparse Grids:
Mixed Families, Growth Rules, Anisotropic Problems
http://people.sc.fsu.edu/~burkardt/presentations/icms_2010.pdf

.....

ARC: Advanced Research Computing
ICAM: Interdisciplinary Center for Applied Mathematics

.....

Funded in part by a grant from Sandia National Laboratories.

VirginiaTech

The Whole Talk in One Page

Sparse grids are a tool for producing integrals or interpolants when the underlying function lies in a high dimensional space.

Sparse grids avoid the exponentially rising cost associated with simple product grids, and can produce more accurate results than a Monte Carlo approach if the function has bounded derivatives of sufficient order.

I am involved in a project that is developing a library of sparse grid codes which allow the user to choose, for any spatial component, the quadrature family, rate of growth, and anisotropy weight.

VirginiaTech

Nested Families of Quadrature Rules

A quadrature rule $\mathcal{Q}$ is a procedure for estimating the integral of a function **f(x)** over a domain script $\mathcal{D}$.

If **f(x)** is known to be smooth, a good strategy employs a nested family of interpolatory quadrature rules of increasing precision.

Q0
Q0+(Q1-Q0)
Q0+(Q1-Q0)+(Q2-Q1)
Q0+(Q1-Q0)+(Q2-Q1)+(Q3-Q2)
...

- allows the reuse of computed data;
- adaptively reaches the appropriate level of precision;
- provides an estimate of the error.

VirginiaTech

Product Rules Are Expensive

For the 1D problem, there is a rich selection of families of quadrature rules, allowing bounded or unbounded domains, a variety of weight functions, the use of derivative information, and composite rules for functions which are not smooth.

For problems in **M** dimensions, 1D rules can be factors in a product rule $Q = Q_1 \otimes Q_2 \otimes ... \otimes Q_N$ if the domain is a product region and the weight functions are separable.

Product rules can achieve desired precision levels, but they do so inefficiently. To integrate all monomials up to total degree **L** in **M** space, a product rule uses about $\mathbf{L}^{\mathbf{M}}$ points.

This cost is unaffordable, but worse, *unnecessary*!

VirginiaTech

Monomials of Total Degree 5 or Less

A reasonable goal for a family of quadrature rules is that each member exactly integrate all monomials up to a given total order. In 2D, a rule that could integrate up to 5th order would need to capture the monomials shown in blue:

7	!	y^7	xy^7	x^2y^7	x^3y^7	x^4y^7	x^5y^7	x^6y^7	x^7y^7
6	!	y^6	xy^6	x^2y^6	x^3y^6	x^4y^6	x^5y^6	x^6y^6	x^7y^6
5	!	y^5	xy^5	x^2y^5	x^3y^5	x^4y^5	x^5y^5	x^6y^5	x^7y^5
4	!	y^4	xy^4	x^2y^4	x^3y^4	x^4y^4	x^5y^4	x^6y^4	x^7y^4
3	!	y^3	xy^3	x^2y^3	x^3y^3	x^4y^3	x^5y^3	x^6y^3	x^7y^3
2	!	y^2	xy^2	x^2y^2	x^3y^2	x^4y^2	x^5y^2	x^6y^2	x^7y^2
1	!	y	xy	x^2y	x^3y	x^4	x^5y	x^6y	x^7y
0	!	1	x	x^2	x^3	x^4	x^5	x^6	x^7
P	!	0	1	2	3	4	5	6	7

VirginiaTech

A Product Rule Overshoots the Goal

A product rule results in a square of precision, not a triangle. Red monomials on "incomplete" diagonals include error terms of the same order, so it is precision that doesn't result in accuracy.

As the dimension **M** increases, red exponentially dominates blue.

7	!	y^7	xy^7	x^2y^7	x^3y^7	x^4y^7	x^5y^7	x^6y^7	x^7y^7
6	!	y^6	xy^6	x^2y^6	x^3y^6	x^4y^6	x^5y^6	x^6y^6	x^7y^6
5	!	y^5	xy^5	x^2y^5	x^3y^5	x^4y^5	x^5y^5	x^6y^5	x^7y^5
4	!	y^4	xy^4	x^2y^4	x^3y^4	x^4y^4	x^5y^4	x^6y^4	x^7y^4
3	!	y^3	xy^3	x^2y^3	x^3y^3	x^4y^3	x^5y^3	x^6y^3	x^7y^3
2	!	y^2	xy^2	x^2y^2	x^3y^2	x^4y^2	x^5y^2	x^6y^2	x^7y^2
1	!	y	xy	x^2y	x^3y	x^4	x^5y	x^6y	x^7y
0	!	1	x	x^2	x^3	x^4	x^5	x^6	x^7
P	!	0	1	2	3	4	5	6	7

VirginiaTech

The Smolyak Approach

The Smolyak formula indicates a way to combine several lower order product rules in such a way that the resulting precision more carefully fills in entire diagonals, with much less "wasted" precision.

This makes it possible to estimate integrals in high dimension while avoiding the explosion of work associated with a single full product rule.

VirginiaTech

Compare Product and Smolyak Precisions

Consider *isotropic* product and Smolyak rules.

If we can afford 1,000,000 function evaluations in M-space, what level of precision P can we expect in our integral approximation?

Suppose we use a non-Gaussian 1D rule in each case. The product rule precision is roughly $1,000,000^{(1/M)}$.

M	P(Product Rule)	P(Sparse Grid)
10	4.0	15
20	2.0	9 or 11
30	1.6	9 or 11
50	1.3	7 or 9
100	1.1	7

After dimension 20, you can't afford a 2-point product rule!

VirginiaTech

Construction of a Sparse Grid

Smolyak's recipe for a sparse grid involves a weighted combination of low order anisotropic product rules.

We have an indexed family of 1D quadrature rules $\mathcal{Q}^L$.
We form rules for dimension **M**, indexed by level **L**.
Here $\mathbf{i} = i_1 + \cdots + i_M$, where i_j is the "level" of the j-th 1D rule.

$$\mathcal{A}(L, M) = \sum_{L-M<|\mathbf{i}|\leq L} (-1)^{L-|\mathbf{i}|} \begin{pmatrix} M-1 \\ L-|\mathbf{i}| \end{pmatrix} (\mathcal{Q}^{i_1} \otimes \cdots \otimes \mathcal{Q}^{i_M})$$

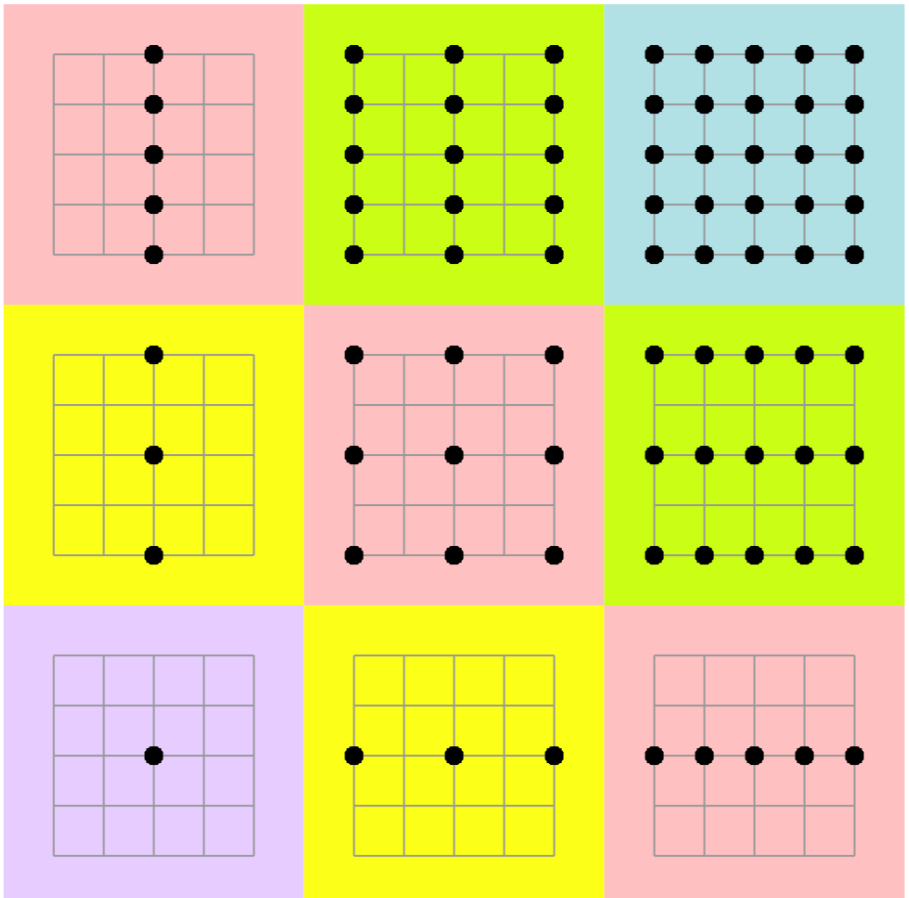
$\mathcal{A}(L, M)$ is a weighted sum of product rules for **M** values of **i**.

We are free to choose the domains, weights, families and growth rates of the 1D factors of these product rules.

VirginiaTech

A Menu of Product Rules

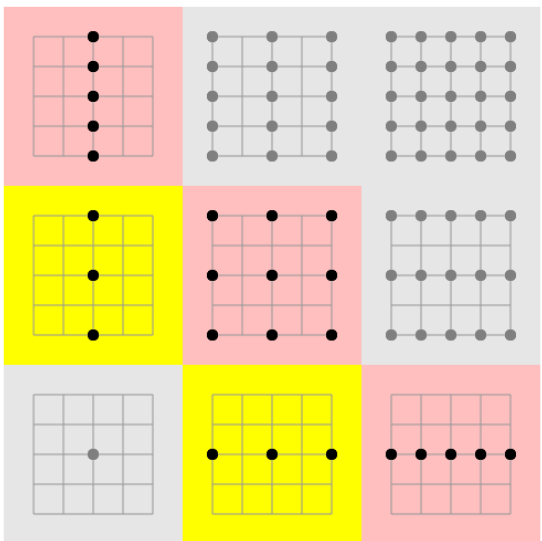
Rules of the same color capture monomials of same total degree.

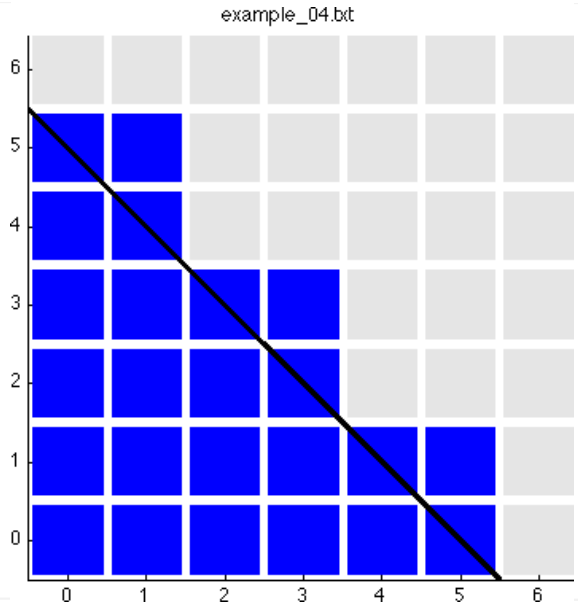


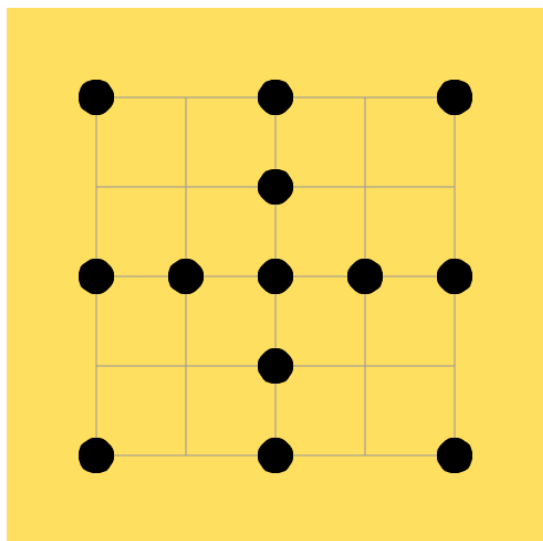
VirginiaTech

Sparse Grid = Sum of Selected Product Rules

`Combination` `Precision` `Grid`



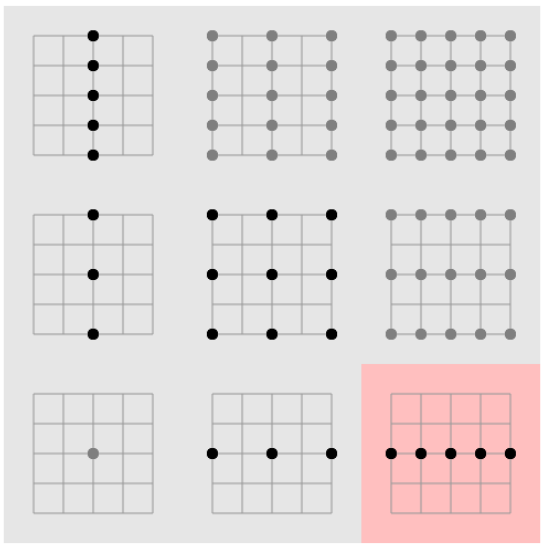




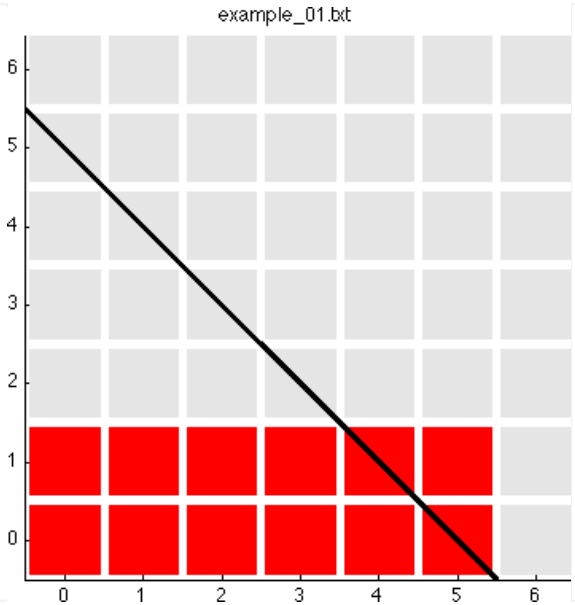
VirginiaTech

Add 5x1 Rule:

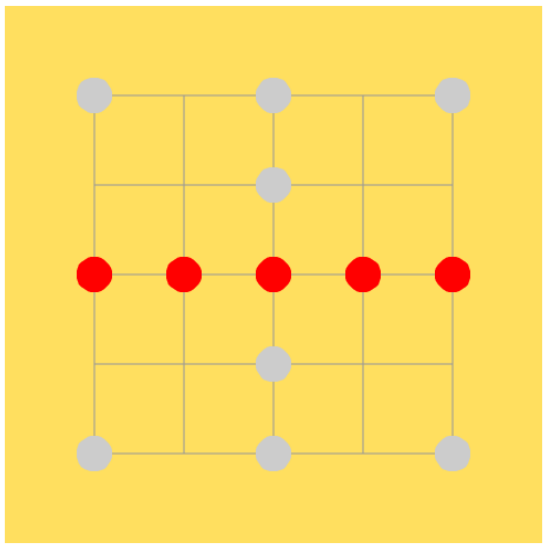
Combination



Precision



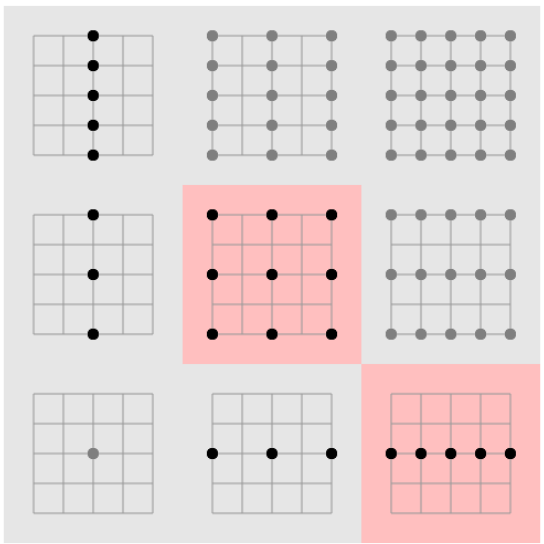
Grid



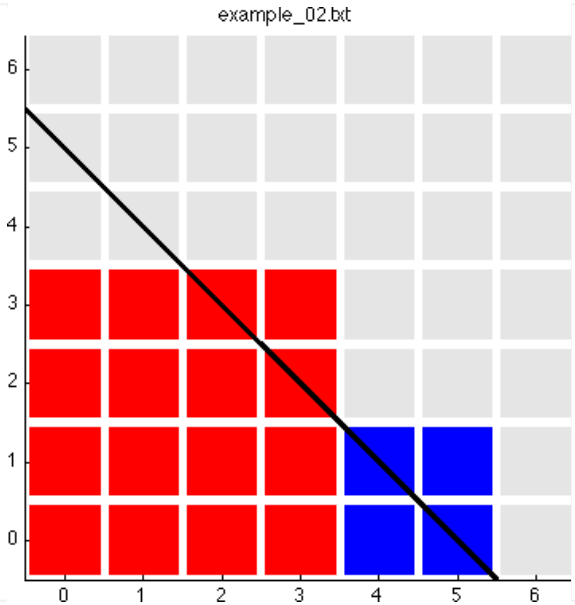
VirginiaTech

Add 3x3 Rule:

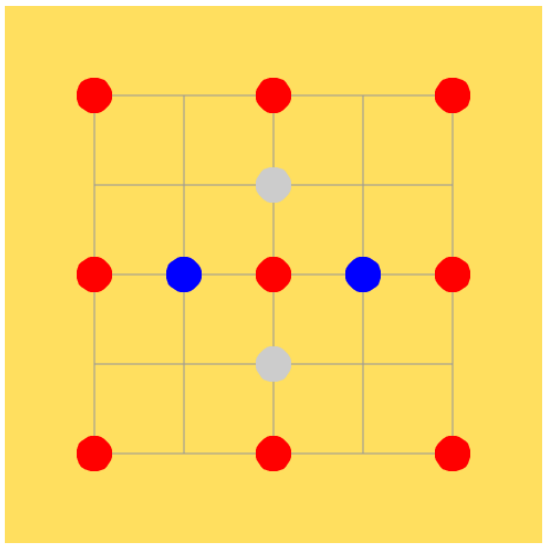
Combination



Precision



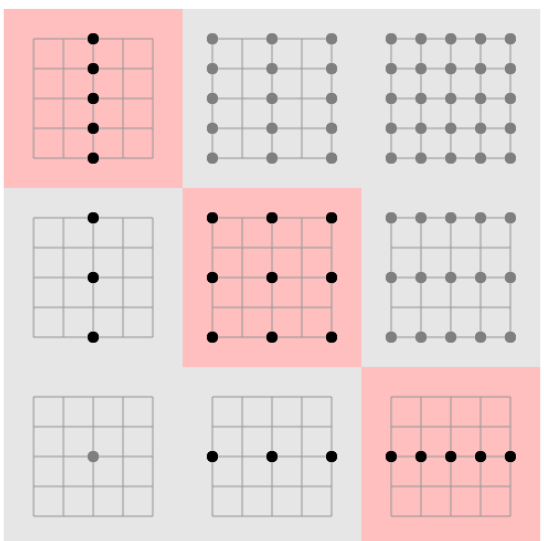
Grid



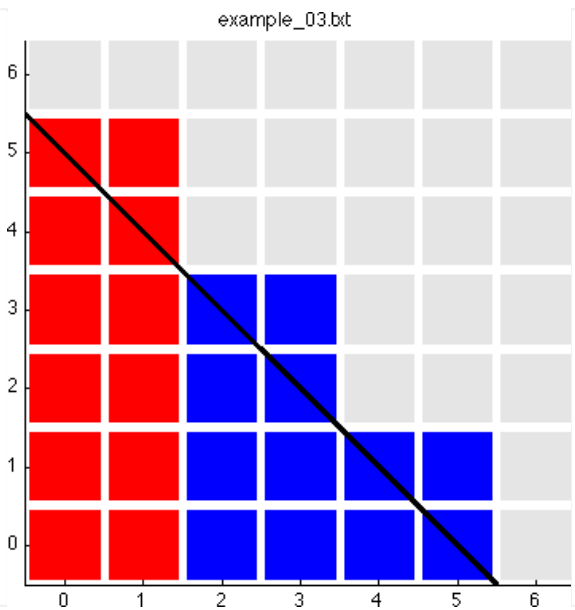
VirginiaTech

Add 1x5 Rule:

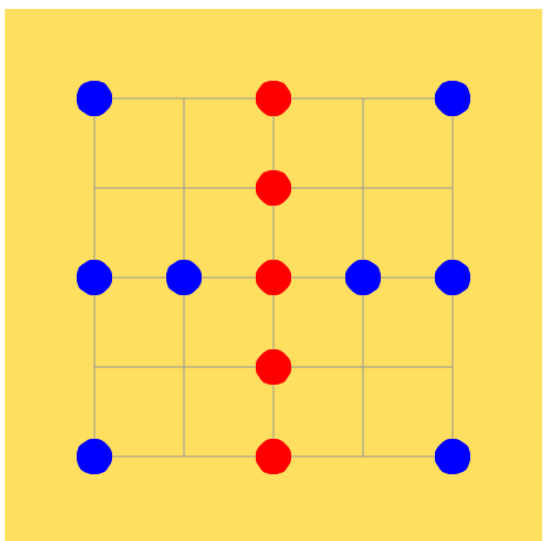
Combination



Precision



Grid



The Results

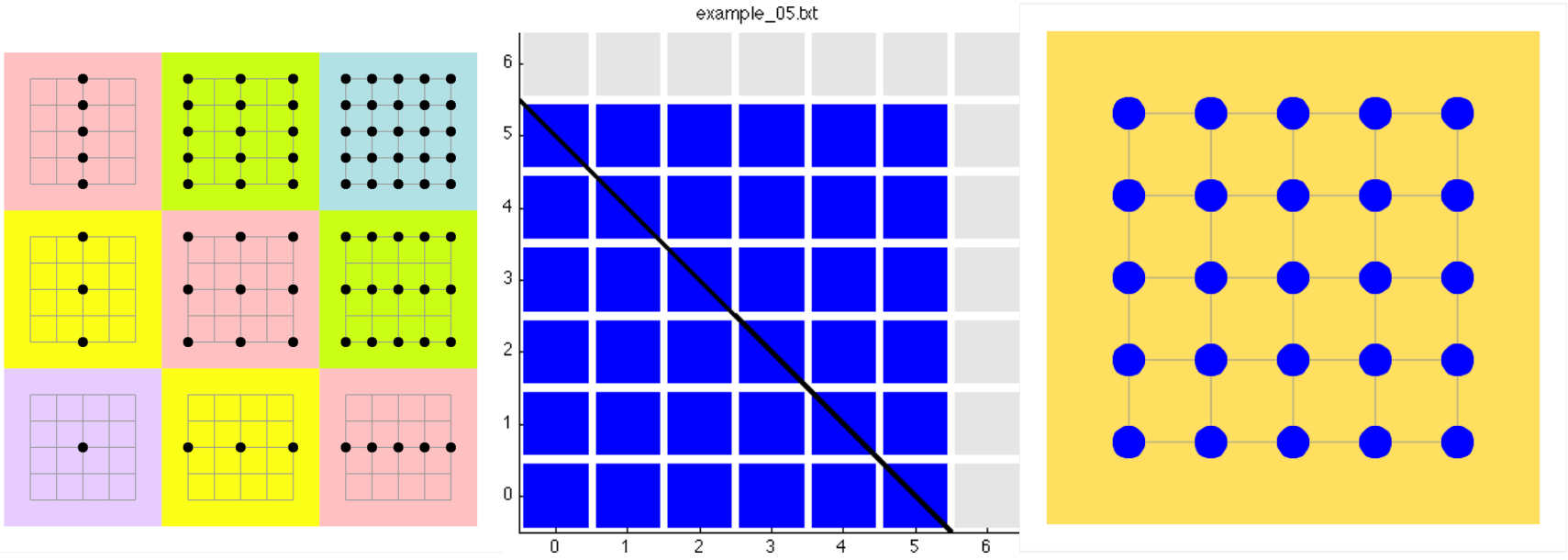
To complete this 2D sparse grid, we include contributions from the two lower order rules, 3x1 and 1x3. Here, we are using a nested family of rules, so the resulting grid does not change, although these lower order rules do affect the computed weights applied to the grid points.

From the precision plot, we can see that the sparse grid claims to be precise for all monomials of total degree 5. It has achieved this precision goal using fewer points than a simple 5x5 product rule.

VirginiaTech

Product Grid:

Combination Precision Grid



SGMGA

The software library embodying the sparse grid rules is called **SGMGA**. This name is meant to record the fact that it is a library that can handle:

1. **S**parse **G**rids
2. **M**ixed families
3. **G**rowth rules
4. **A**nisotropic weighting

VirginiaTech

#1: Sparse Grid

The 1D quadrature families may be chosen from:

1. Clenshaw-Curtis
2. Fejer Type 2
3. Gauss-Patterson
4. Gauss-Legendre
5. Gauss-Hermite
6. generalized Gauss-Hermite
7. Gauss-Laguerre
8. generalized Gauss-Laguerre
9. Gauss-Jacobi
10. user-defined Golub Welsch rules

VirginiaTech

#2: Mixed Families

For each dimension, the user may choose a particular 1D quadrature family.

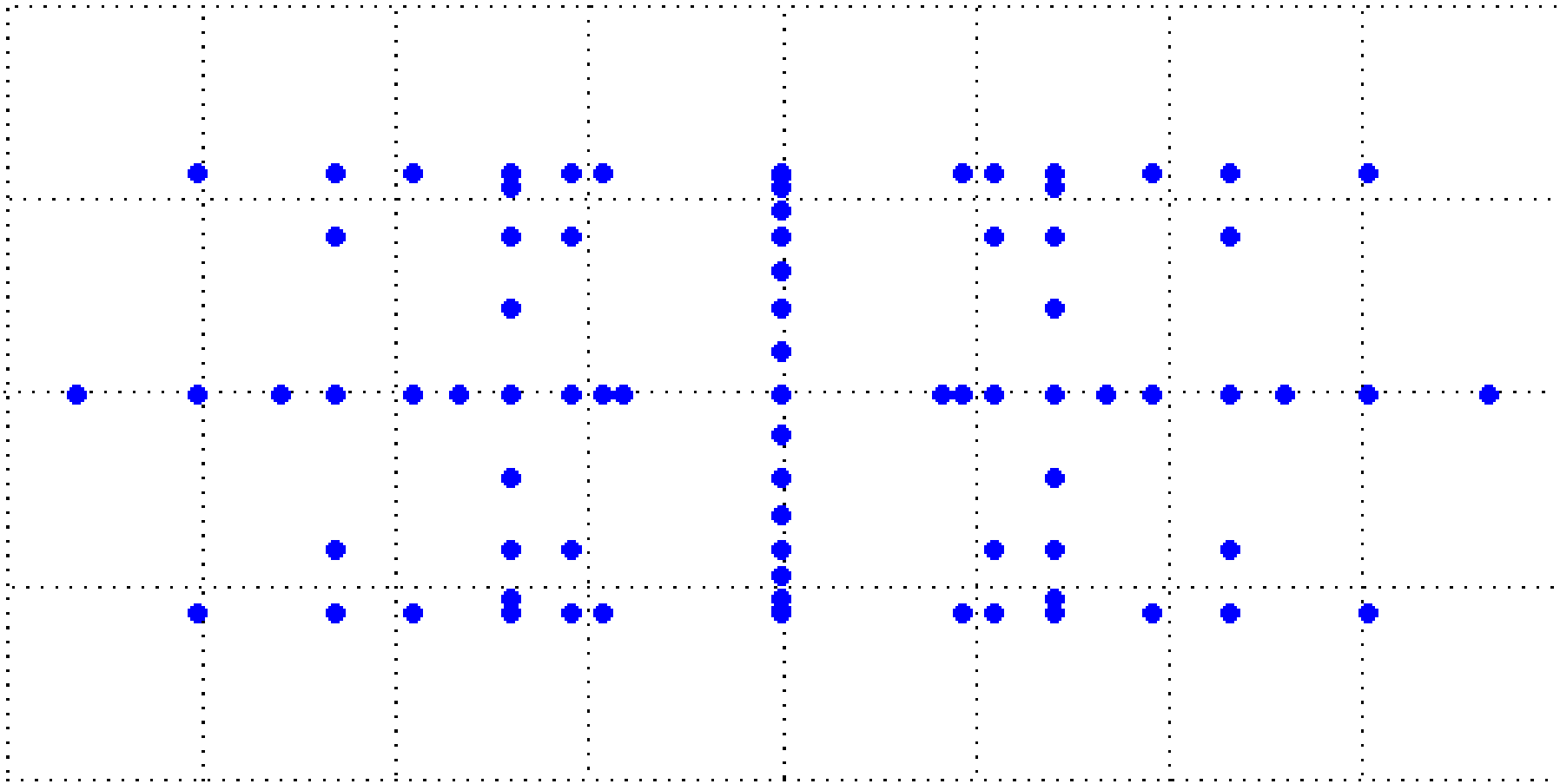
If a family is parameterized, the associated parameters may be specified for each dimension.

(generalized Laguerre and Hermite rules, and the Jacobi rules.)

This makes it possible to correctly treat cases involving a mixture of uniform and normally distributed quantities, for instance, or cases in which a single distribution is used, but the parameters defining that distribution vary from one dimension to the next.

VirginiaTech

Gauss-Hermite X Clenshaw Curtis, Level 4



VirginiaTech

#3: Growth-Rules

For each problem dimension, the chosen quadrature family comprises an indexed sequence of rules of increasing order.

The **growth rule** relates the index **L** to the order **O**.

For the nested Gauss Patterson family, for instance, this is typically:

$$O = 2^{L+1} - 1$$

The user may override the default growth rule.

VirginiaTech

Growth Rules: Slow Growth for Nested Grids

Using a nested family of 1D rules reduces the number of abscissas in the resulting sparse grid.

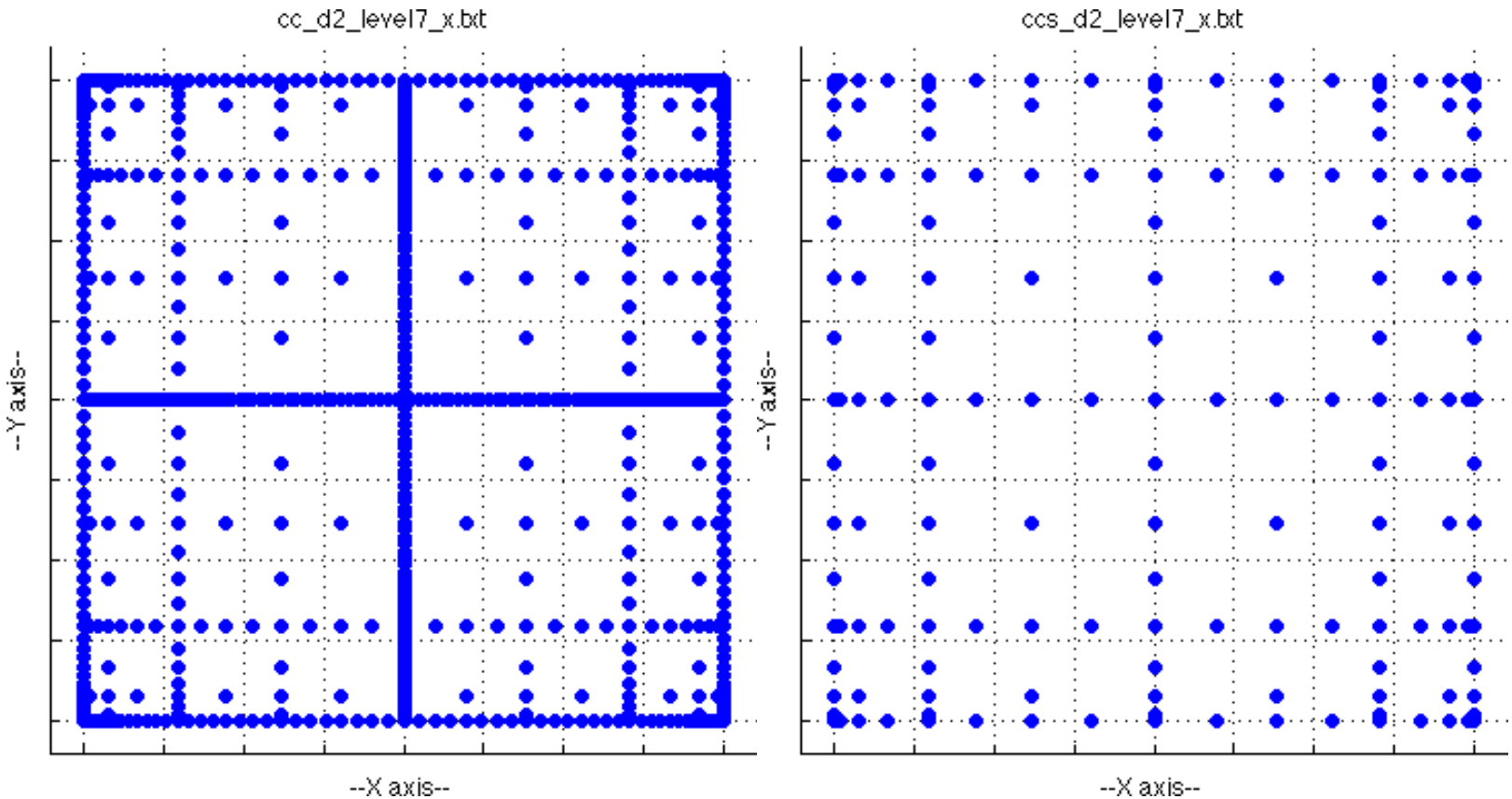
The classical construction doubles the point count at each level.

We have implemented a **slow growth** version of such rules; at each level, we use the lowest order nested rule that will achieve the necessary precision.

The advantages of the slow growth approach are most apparent in low dimensions, or for high order.

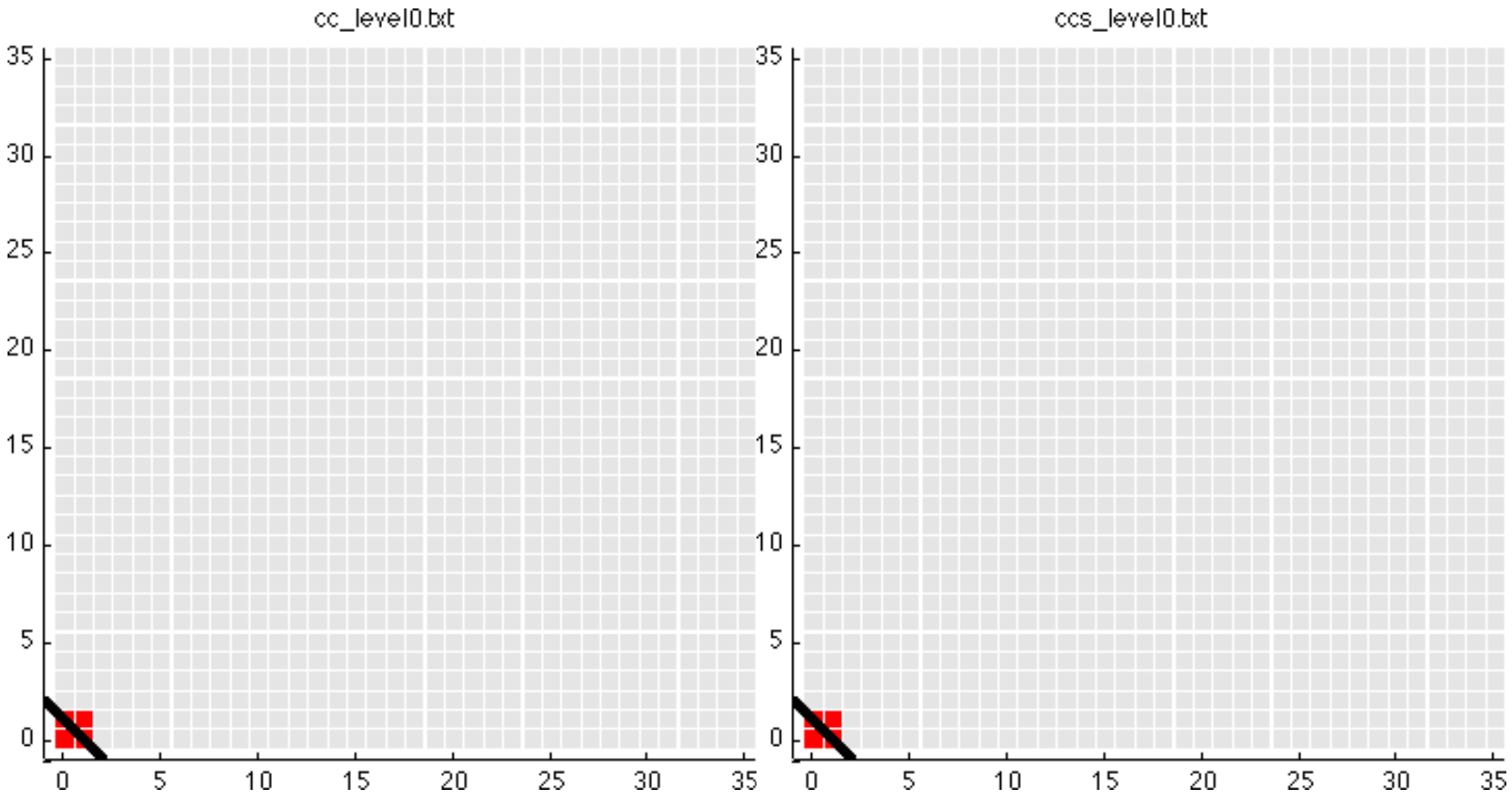
VirginiaTech

Growth Rules: Level 7 CC, Default versus Slow Growth



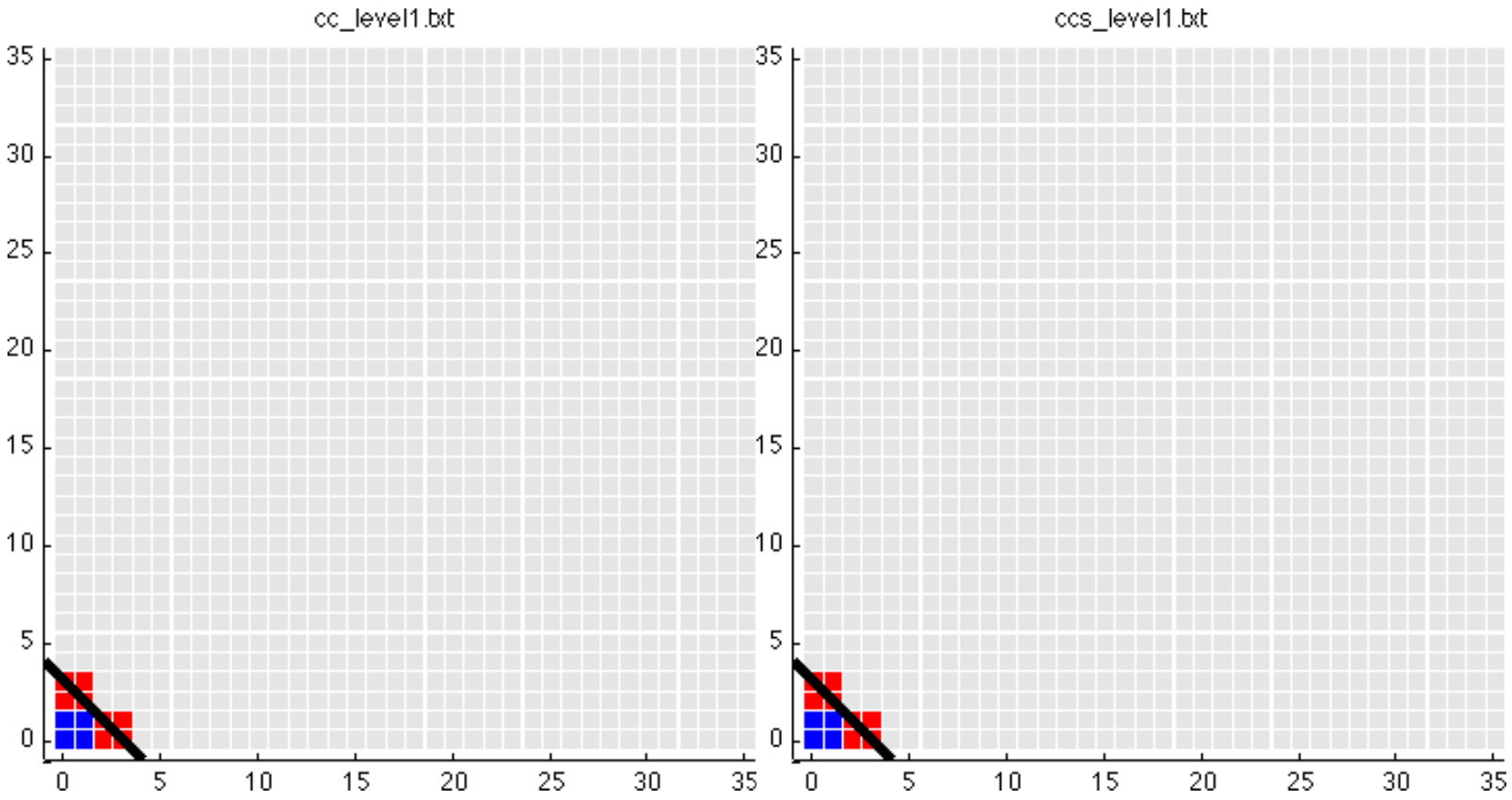
VirginiaTech

Default vs Slow Clenshaw Curtis, Level 0



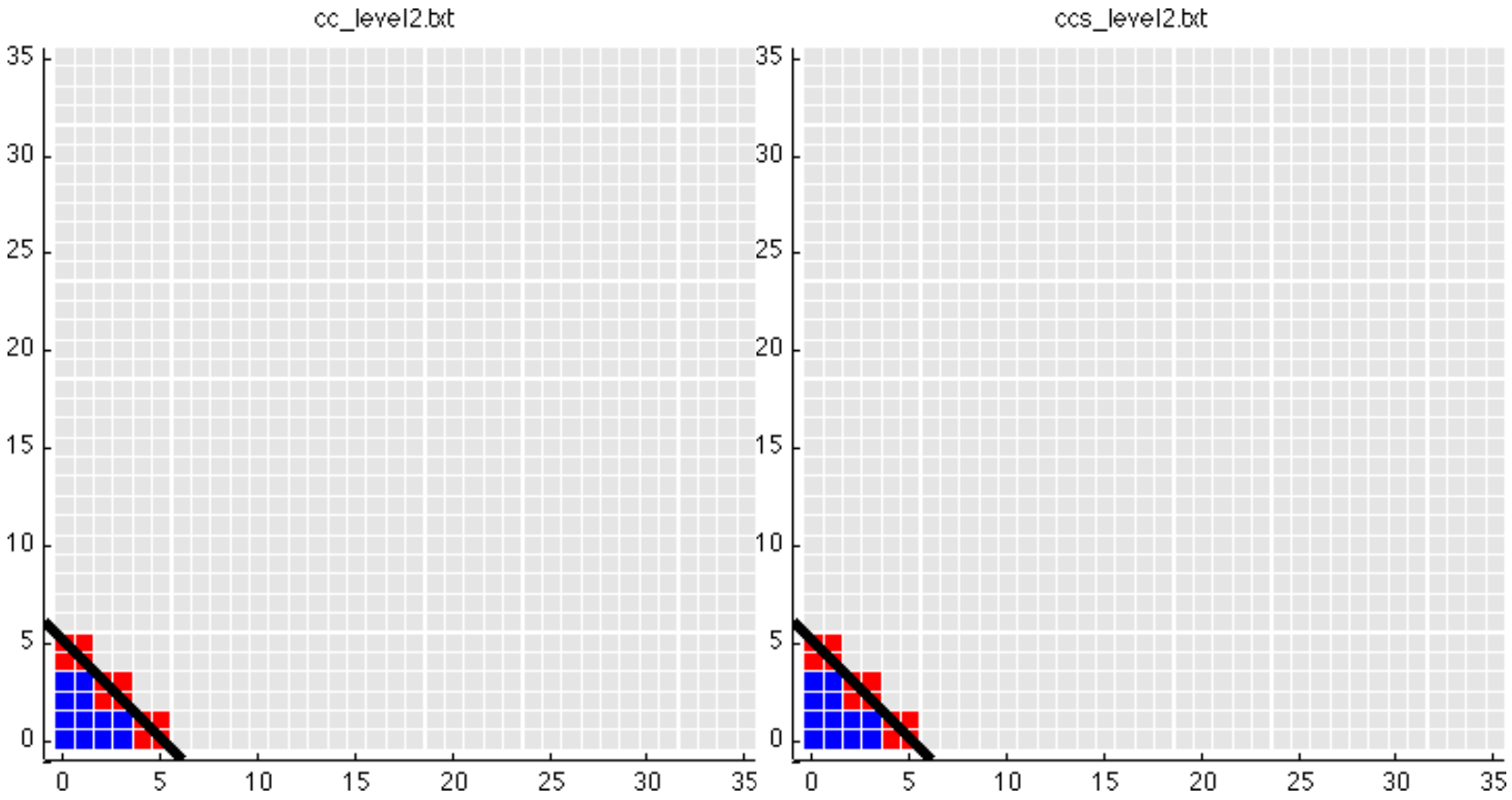
VirginiaTech

Default vs Slow Clenshaw Curtis, Level 1



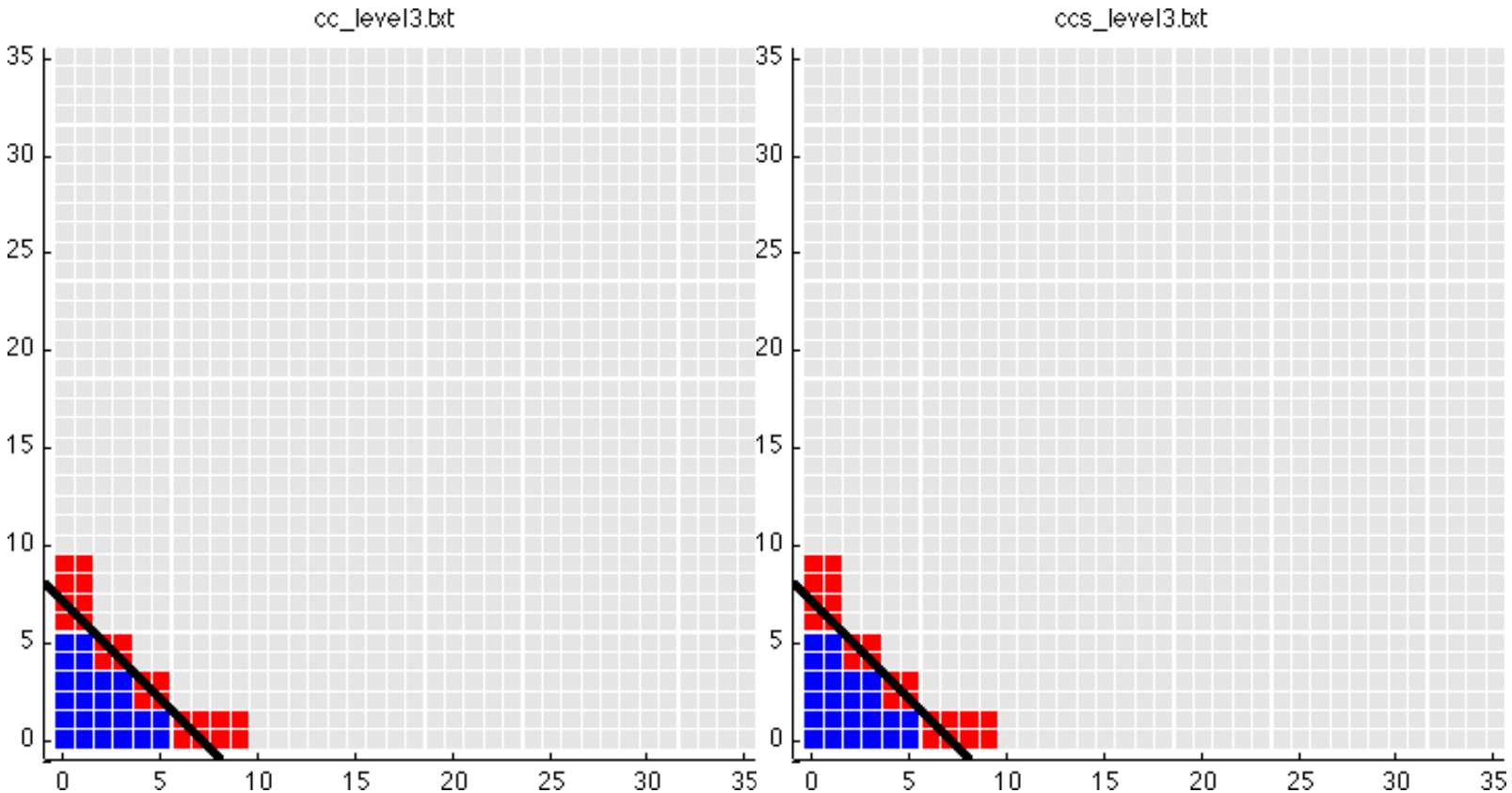
VirginiaTech

Default vs Slow Clenshaw Curtis, Level 2



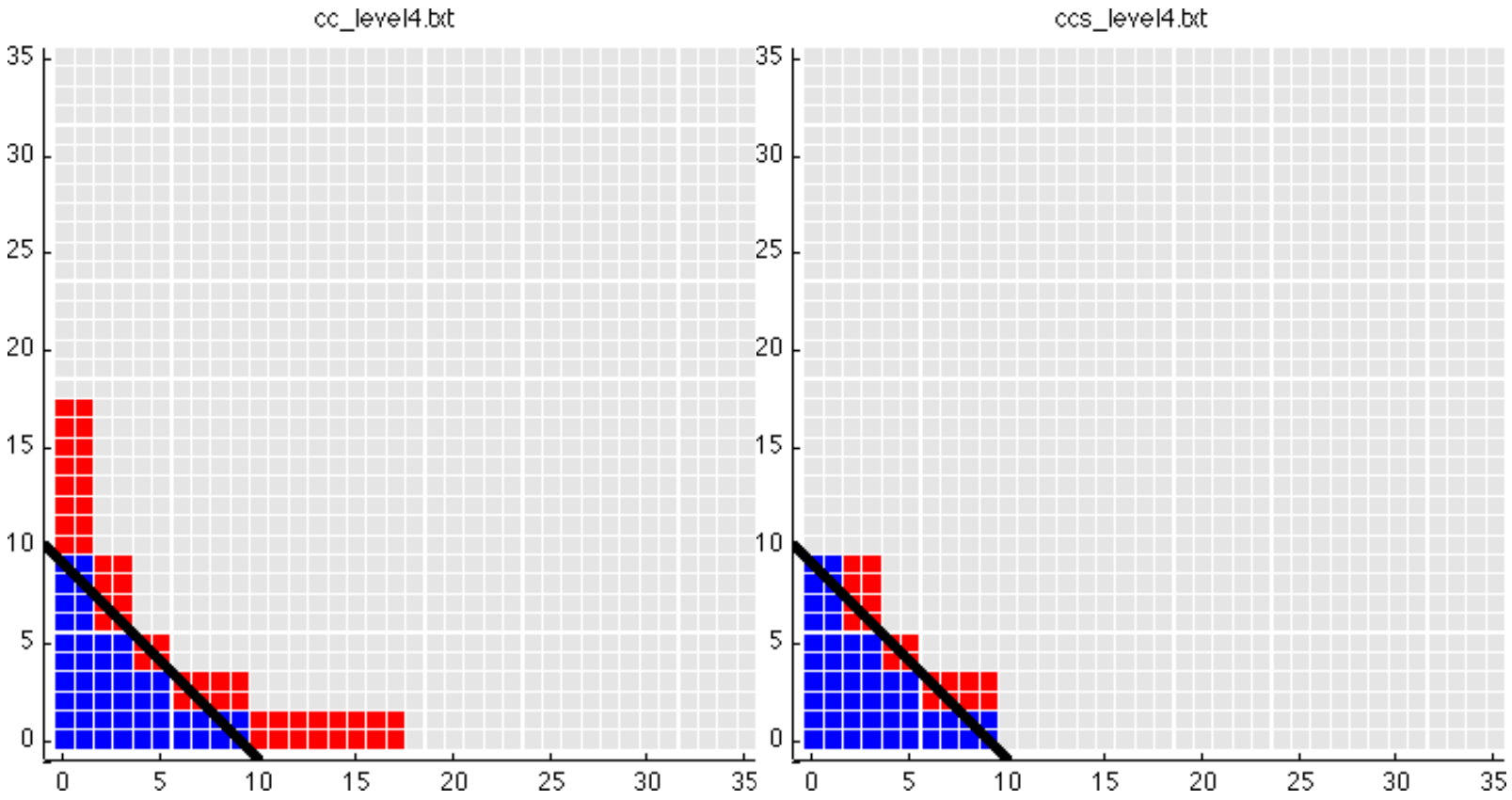
VirginiaTech

Default vs Slow Clenshaw Curtis, Level 3

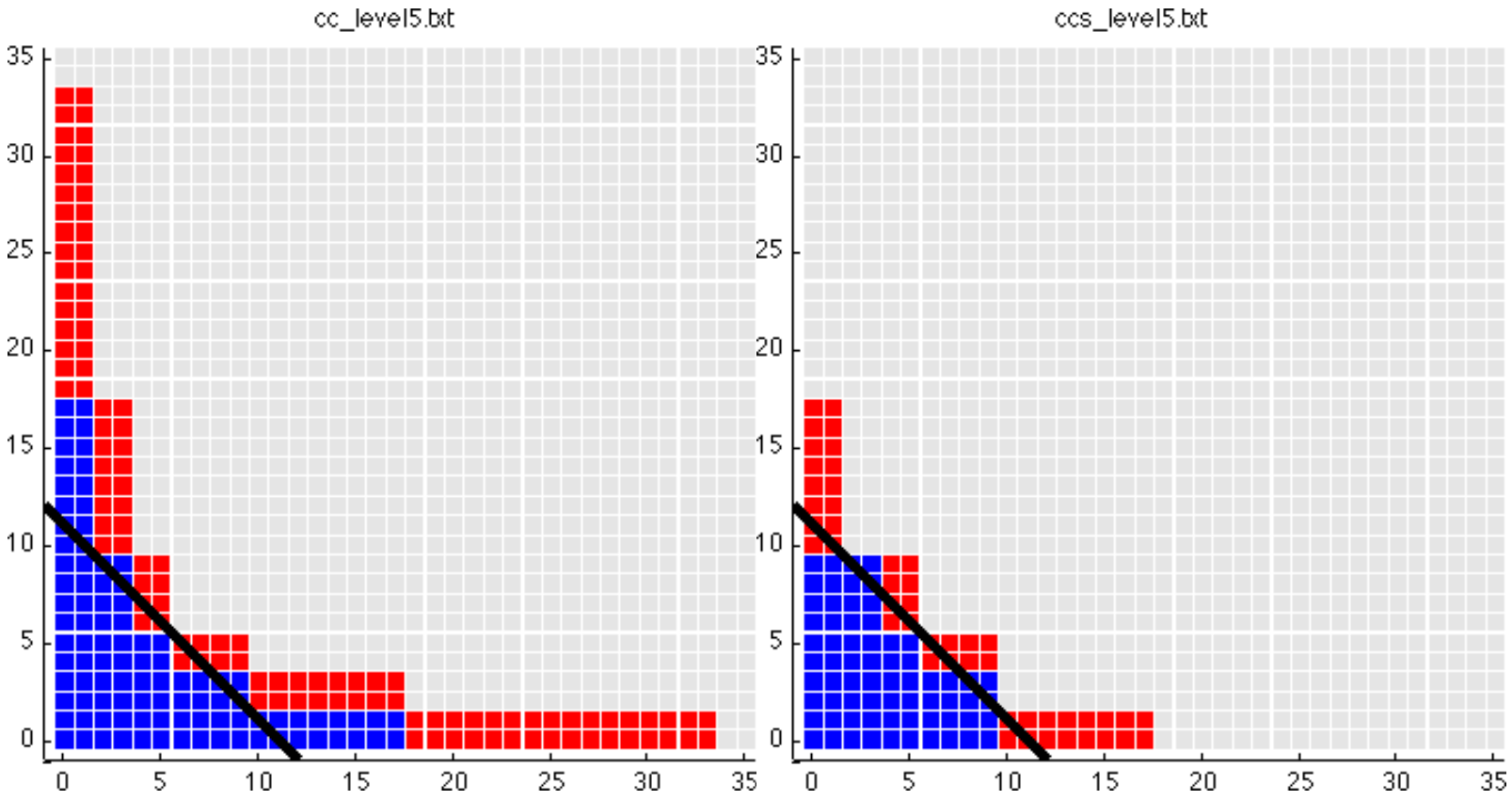


VirginiaTech

Default vs Slow Clenshaw Curtis, Level 4

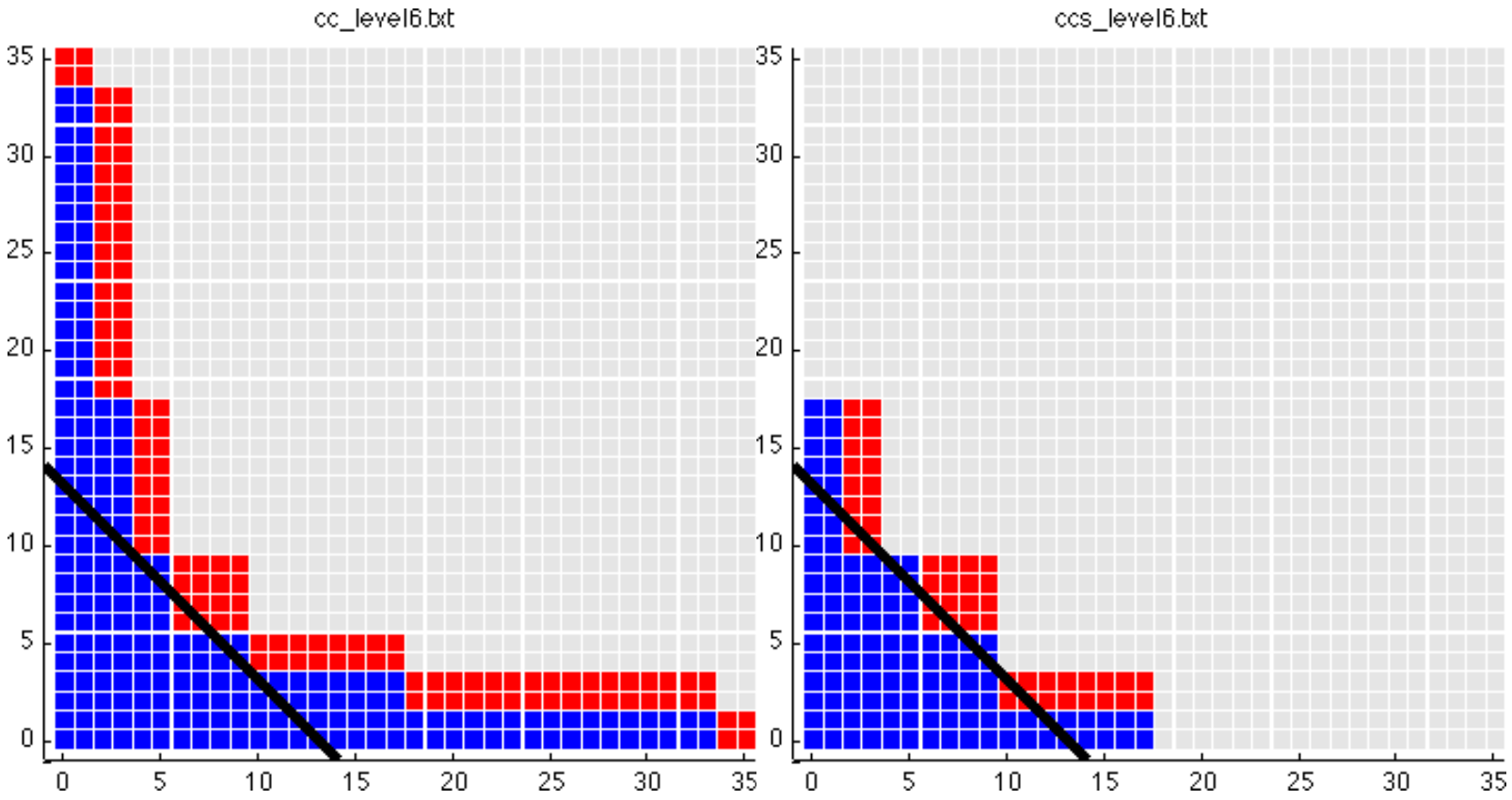


Default vs Slow Clenshaw Curtis, Level 5



VirginiaTech

Default vs Slow Clenshaw Curtis, Level 6



VirginiaTech

#4: Anisotropic Rules

Many problems in which stochastic factors have been included will have tens or even hundreds of dimensions.

Most physically meaningful problems will exhibit **anisotropy**, that is, certain dimensions will dominate the behavior of the system, while many dimensions will have relatively little influence.

If this anisotropy is anticipated, it can be built into the sparse grid construction process in advance.

If unexpected anisotropy is encountered, the sparse grid construction could be modified to adapt to this discovery.

VirginiaTech

The Anisotropic Smolyak Formula

We form isotropic sparse grids from product grids whose level vectors satisfy the linear constraints:

$$L - M < \sum_j i_j \leq L$$

We introduce anisotropy using a weight vector α. The new constraint is

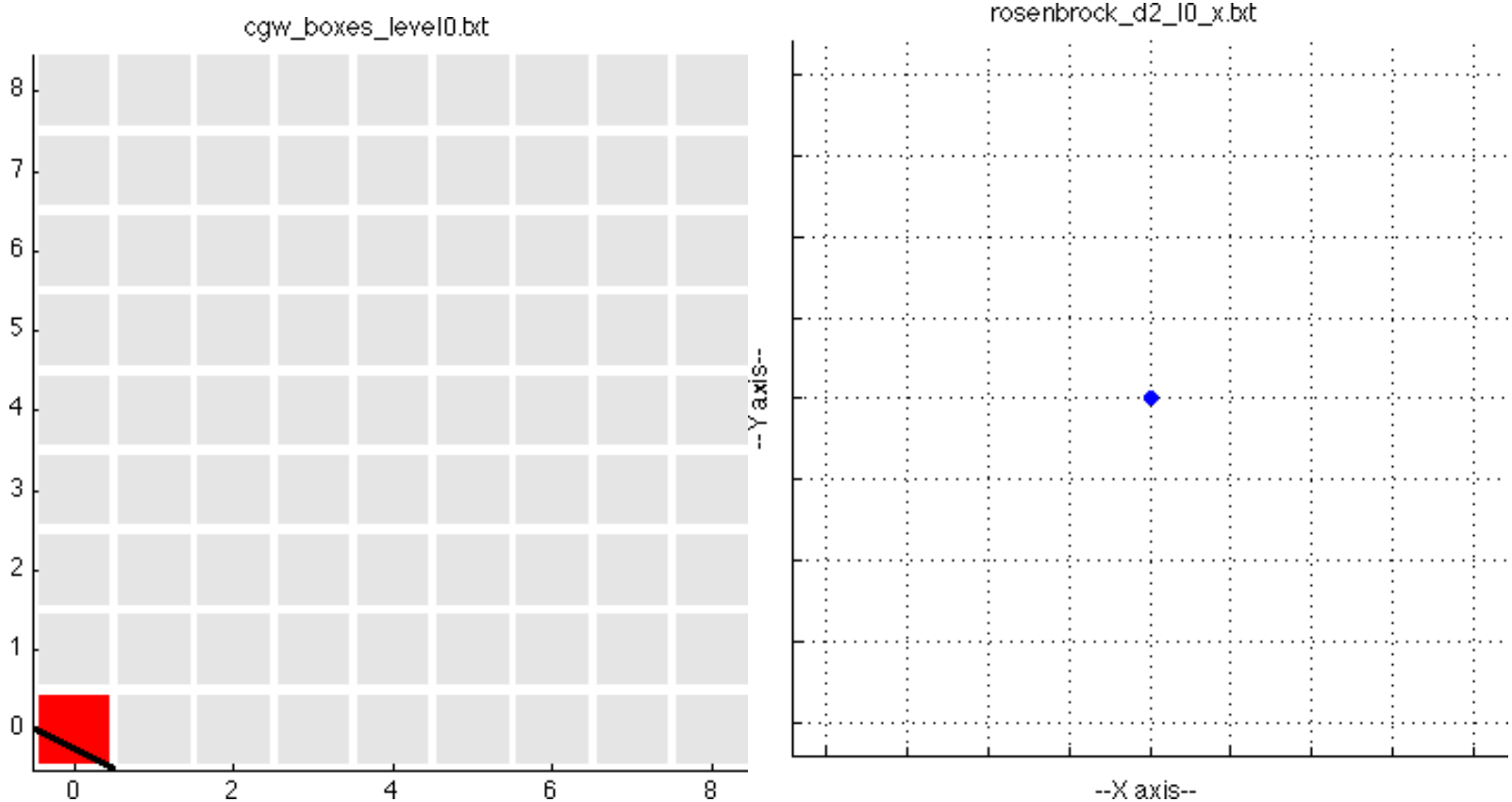
$$L \cdot \min(\alpha) - \sum_j \alpha_j < \sum_j \alpha_j \cdot i_j \leq L \cdot \min(\alpha)$$

with the combining cofficient defined by

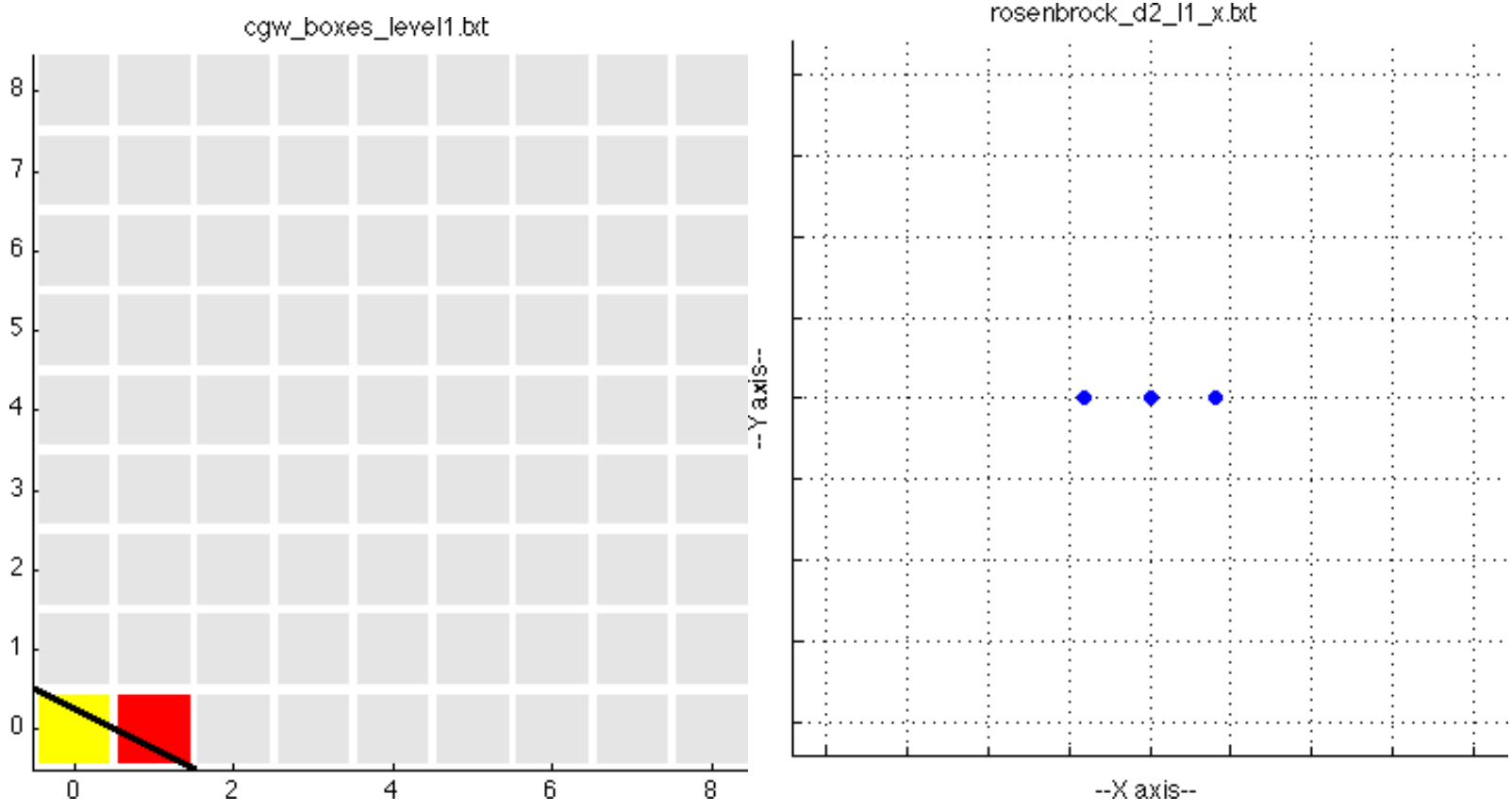
$$c_\alpha(\mathbf{i}) = \sum_{\substack{\mathbf{j} \in \{0,1\}^d \\ \mathbf{i}+\mathbf{j} \text{ satisfies constraint}}} (-1)^{|\mathbf{j}|}$$

VirginiaTech

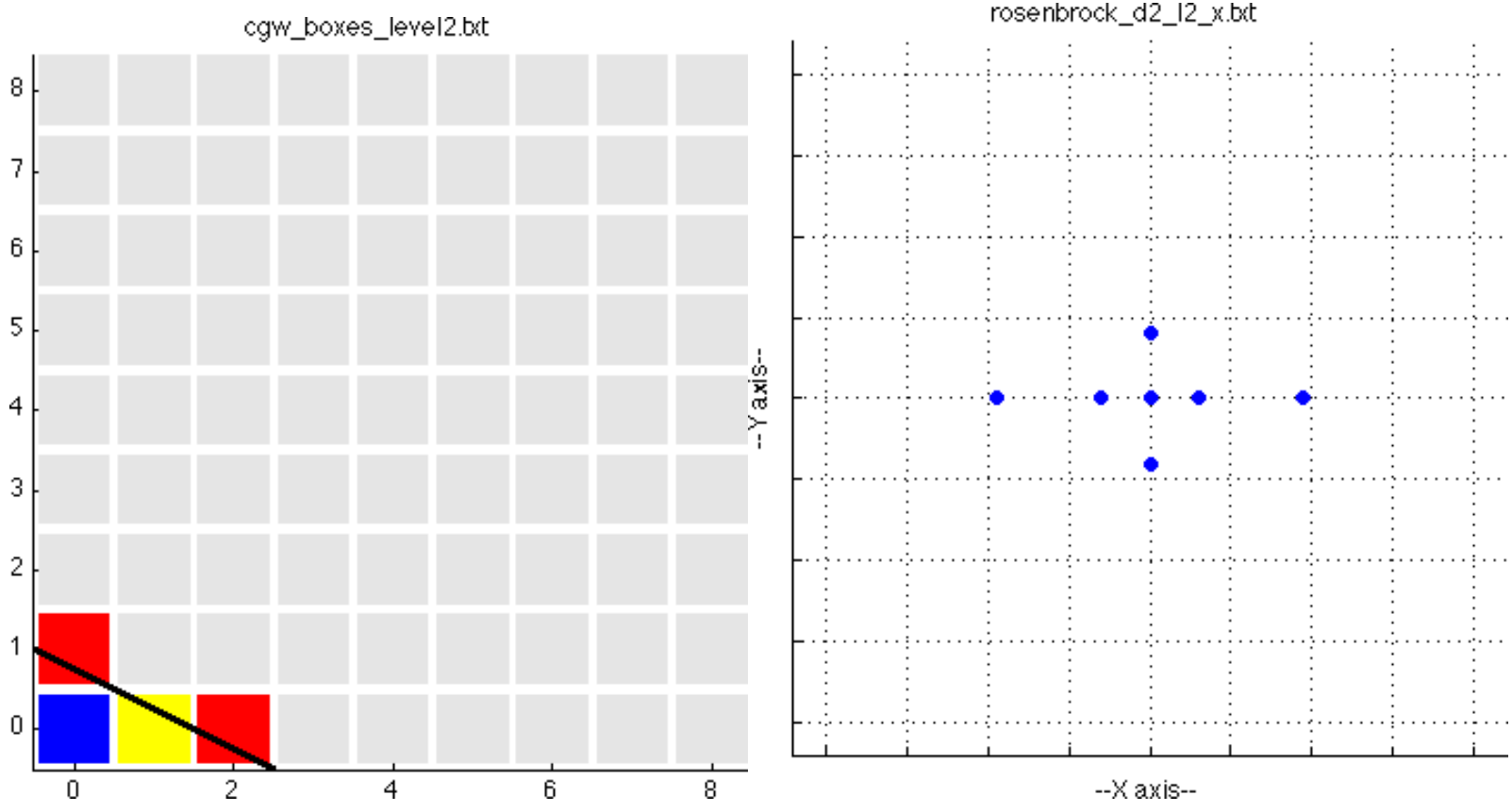
[2,1] Hermite, Level 0



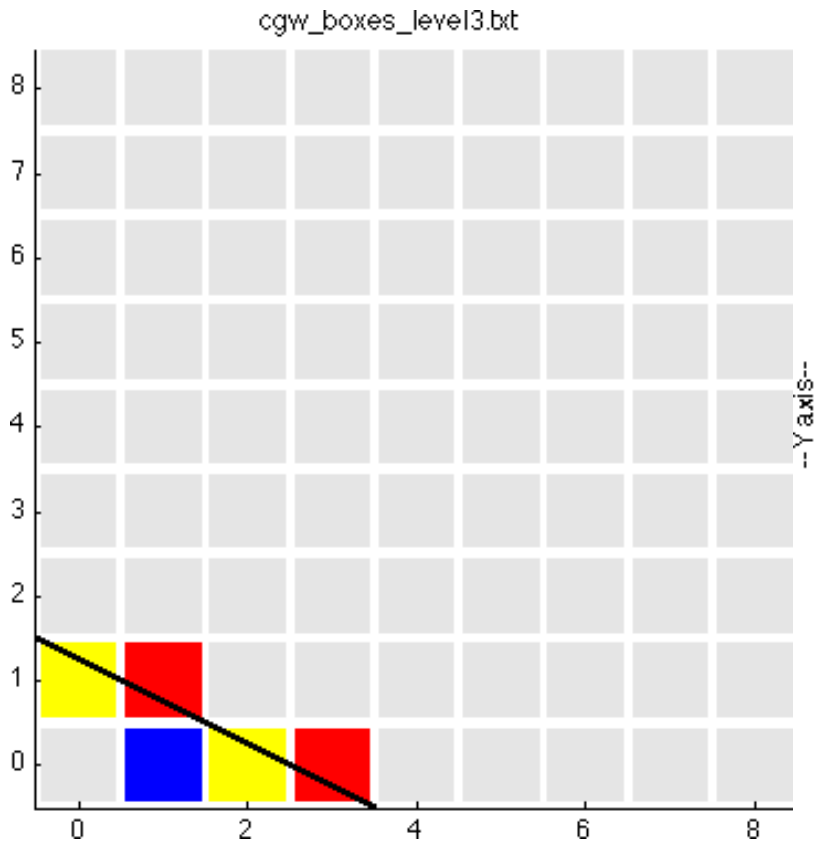
[2,1] Hermite, Level 1

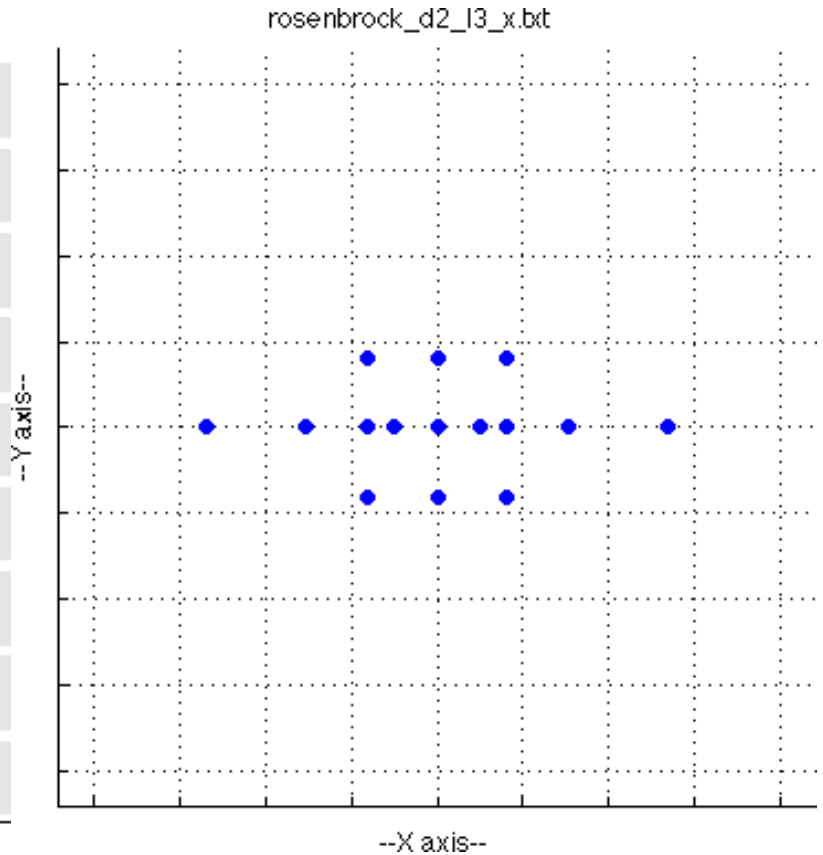


[2,1] Hermite, Level 2



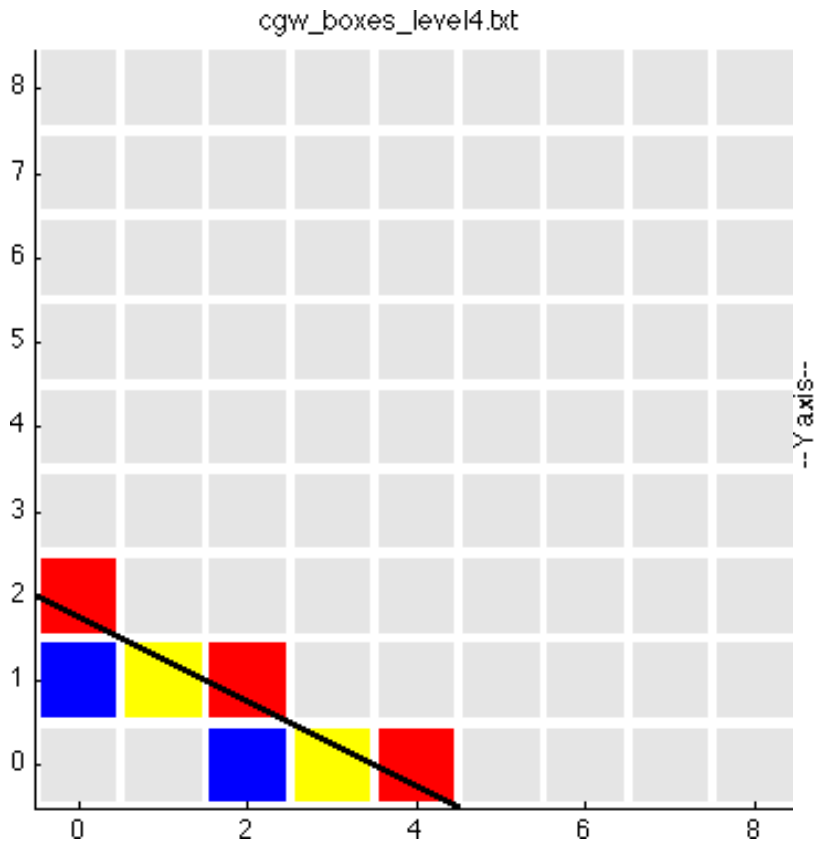
[2,1] Hermite, Level 3

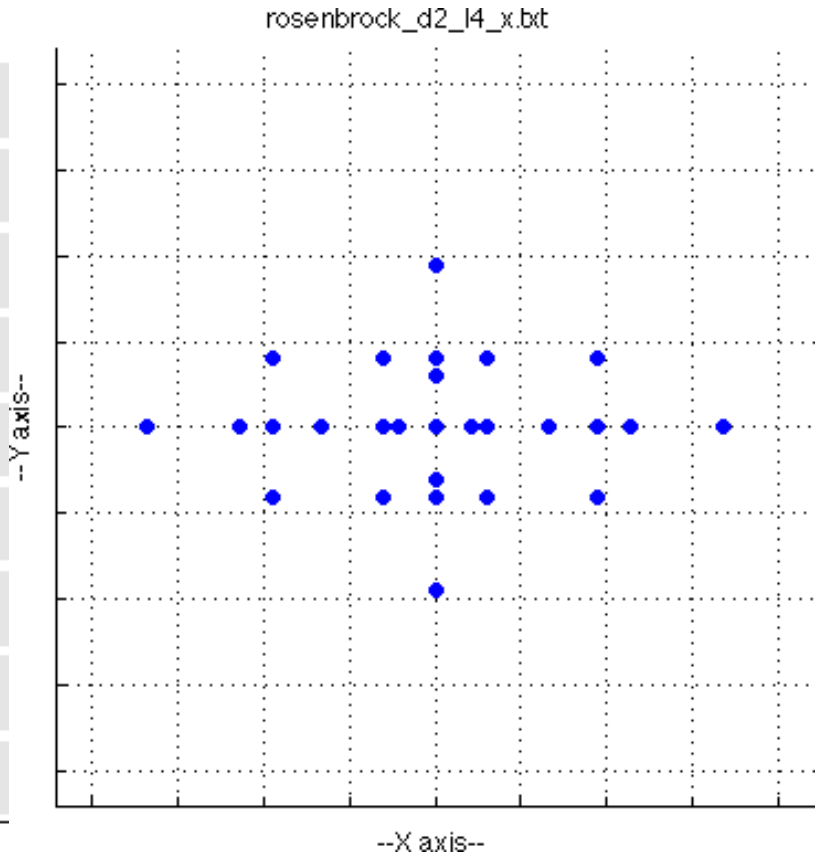




VirginiaTech

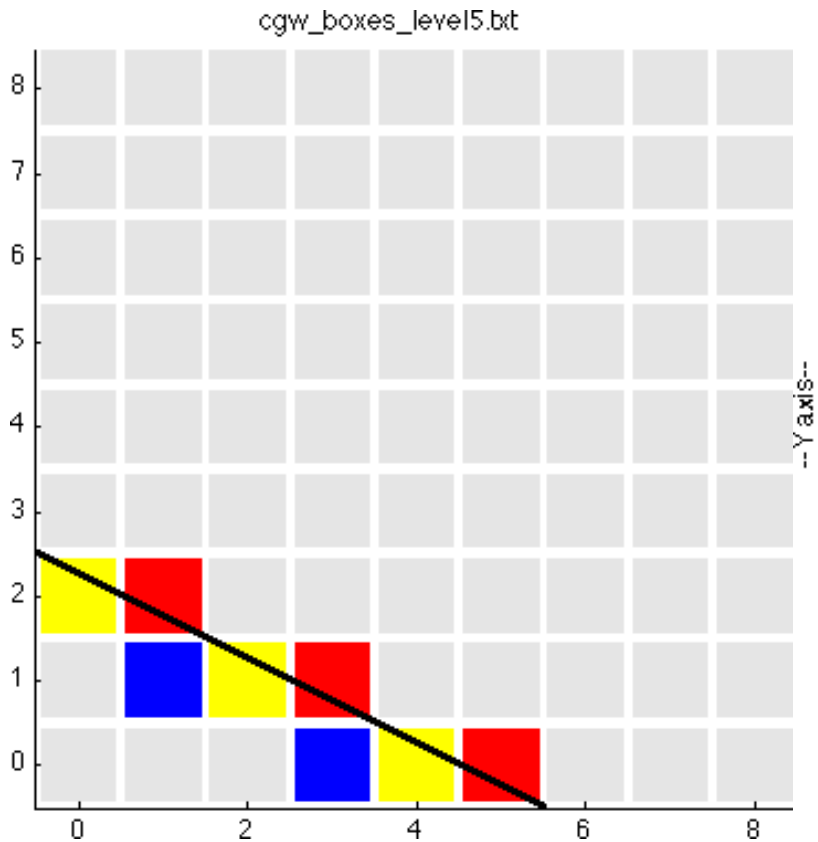
[2,1] Hermite, Level 4

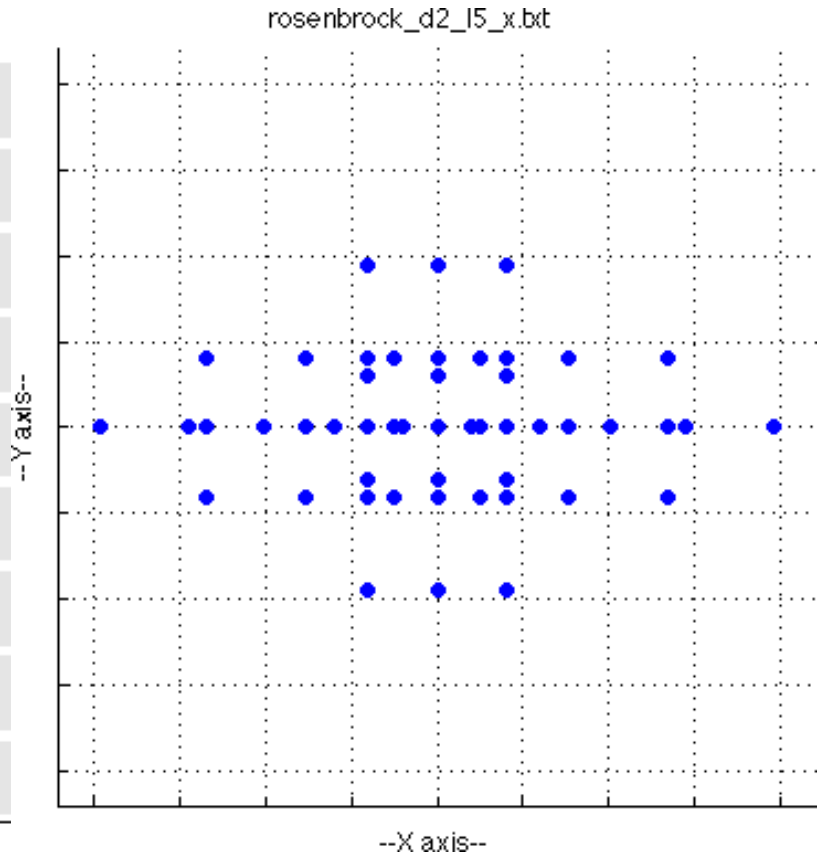




VirginiaTech

[2,1] Hermite, Level 5





VirginiaTech

CONCLUSION: Adaptive Anisotropy

The anisotropy allowed by the SGMGA program must be prescribed beforehand, in terms of relative weights for each dimension, which define a linear constraint on level vectors.

The sparse grid is then formed from product grids whose level vectors lie between corresponding sets of parallel lines (in 2D) or hyperplanes.

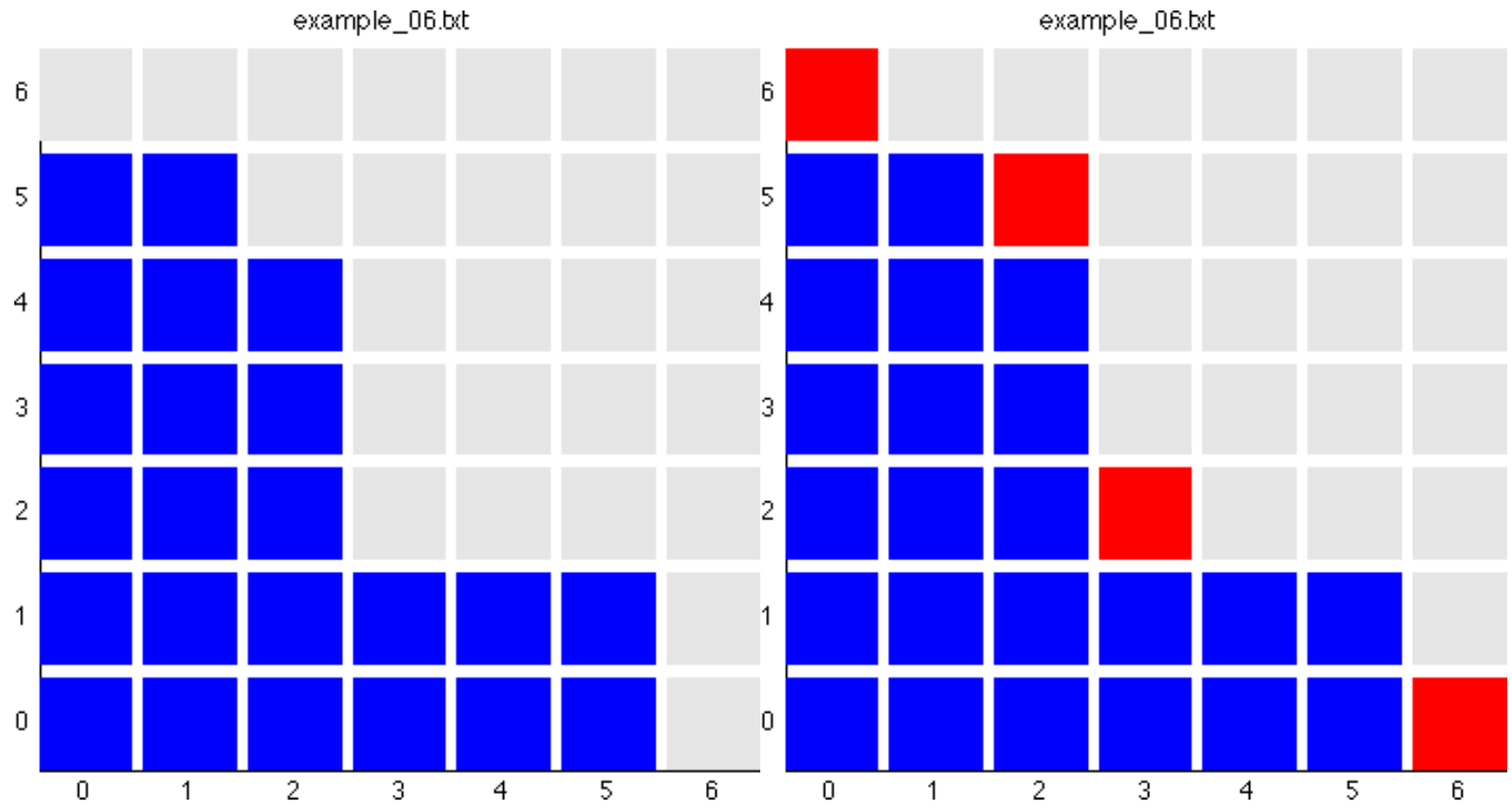
The anisotropic formula can be applied to more general groupings of level vectors.

An approach that would make sense would be to adaptively seek level vectors that could be added to the current rule.

VirginiaTech

CONCLUSION: Adaptive Anisotropy

Consider adding product rules at "corners" of the diagram.



CONCLUSION: Adaptive Anisotropy

The formula for the combining coefficient in the anisotropic case is remarkably similar: the coefficient of a level vector is the signed count of all level vectors which still satisfy the anisotropic level constraint after some entries have been increased by 1.

Unfortunately, in **M** dimensions, a naive approach to checking this simple linear condition would require 2^M checks.

We are investigating efficient ways to count the perturbed level vectors that avoid the recurrence of the dimensionality explosion.

VirginiaTech

CONCLUSION: Remarks

It would be useful to have analogues of the Gauss-Patterson family for problems involving Hermite or Laguerre weight functions.

The sparse grid approach can also be used for interpolation (Klimke/Wohlmuth SPINTERP)

For irregular functions, there is an approach using hierarchical set of piecewise linear interpolants.

VirginiaTech

CONCLUSION: References

Some technical notes at
http://people.sc.fsu.edu/~burkardt/presentations:

- **icms_2010.pdf**, these slides
- **sgmga_1d_rules.pdf**, *1D quadrature rules for sparse grids;*
- **sgmga_ccs.pdf**, *Slow exponential growth for Clenshaw Curtis sparse grids;*
- **sgmga_coefficient.pdf**, *The combining coefficient for anisotropic sparse grids;*
- **sgmga_counting.pdf**, *Counting the abscissas in sparse grids;*
- **sgmga_gps.pdf**, *Slow exponential growth for Gauss Patterson sparse grids;*
- **sgmga_precision.pdf**, *On the precision of certain multidimensional quadrature schemes.*

VirginiaTech

CONCLUSION: References

Volker **Barthelmann**, Erich **Novak**, Klaus **Ritter**,
High Dimensional Polynomial Interpolation on Sparse Grids,
Advances in Computational Mathematics, 12(4) 2000, p273-288.

John **Burkardt**, Max **Gunzburger**, Clayton **Webster**,
Reduced Order Modeling of Some Nonlinear Stochastic PDE's,
IJNAM, 4(3-4) 2007, p368-391.

Thomas **Gerstner**, Michael **Griebel**,
Numerical Integration Using Sparse Grids,
Numerical Algorithms, 18(3-4), 1998, p209-232.

Andreas **Klimke**, Barbara **Wohlmuth**,
Algorithm 847: SPINTERP: Piecewise Multilinear Hierarchical Sparse Grid Interpolation in MATLAB,
ACM Transactions on Mathematical Software, 31,(4), 2005, p561-579.

VirginiaTech

CONCLUSION: References

Fabio **Nobile**, Raul **Tempone**, Clayton **Webster**,
An Anisotropic Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data,
SIAM Journal on Numerical Analysis,
Volume 46, Number 5, 2008, pages 2411-2442.

Knut **Petras**,
Smolyak Cubature of Given Polynomial Degree with Few Nodes for Increasing Dimension,
Numerische Mathematik,
Volume 93, Number 4, February 2003, pages 729-753.

Sergey **Smolyak**,
Quadrature and Interpolation Formulas for Tensor Products of Certain Classes of Functions,
Doklady Akademii Nauk SSSR, 4, 1963, p240-243.

VirginiaTech