Software Spec – NumerovSolver or NumerovIntegrator Class

Numerov’s method is useful for solving DEs of the form:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Class NumerovIntegrator

Public Shared Function Integrate(mesh as IRadialMesh, v as double(), u as double(), istart as integer, iend as integer, ystart as double, yprimestart as double) as NumerovIntegrationResult

Branches internally depending on the value of u, and istart < or > iend, and whether or not mesh is uniform.

Exceptions:

Mesh is Nothing 🡪 Throw InvalidArgumentException

Mesh has fewer than 3 points 🡪 Throw InvalidArgumentException

Dimensions of V or U is different than Mesh 🡪 Throw InvalidArgumentException

If iStart or iEnd out of range of Mesh dimensions, 🡪 Throw InvalidArgumentException

Ystart or yprimestart = NaN, MAXVALUE or MINVALUE 🡪 Throw InvalidArgumentException

If iStart > iEnd, integration is in the backward direction.

Formulas:

If the interpoint spacing is non-uniform such that

|  |  |  |
| --- | --- | --- |
| OLD VERSION, U = 0, Forward |  |  |
| NEW VERSION, Forward |  |  |
| OLD VERSION, u=0, Backward |  |  |
| NEW VERSION, Backward |  |  |
| NEW VERSION, u=0, Forward |  |  |
| NEW VERSION, u=0, Backward |  |  |

Need to unit test the old vs. the new versions against a known (analytic) solution and determine which one is correct.

Formulas for use on a uniform mesh:

|  |  |  |
| --- | --- | --- |
| U=0, forward |  |  |
| U=0, backward |  |  |

Structure NumerovIntegrationResult

Sub New(y as Double(), yPrime as Double(), istart as integer, iend as integer)

Public Readonly Property y as Double()

Public Readonly Property yPrime as Double()

Public Readonly Property iStart as Integer

Public Readonly Property iEnd as Integer

Solving for y1:

|  |  |  |
| --- | --- | --- |
|  |  |  |

If *u* = 0,

|  |  |  |
| --- | --- | --- |
|  |  |  |