Background Functions

```
GenDandB[sys_] := Module[
  {StencilPoints = sys[[1]], DerivativeOrder = sys[[2]], n = sys[[3]], dS, i, j, h, S, σ,
   Σ, ρ, Π, r, m, PDE, Vars, numPD, multiIndices, denoms, k, PDECoeff, d, b, nPoints},
  nPoints = Length[StencilPoints];
  h = (150 - 50) / 640;
  S[i_] := 150; S[2] = 150; S[3] = 150;
  σ = 0.2; ρ = 0.2; r = 0.03;
  PDE = Sum[r S[i] dS[i], {i, n}] +
    1 / 2 Sum[Σ[i] Σ[j] Π[i, j] S[i] S[j] dS[i] dS[j], {i, n}, {j, n}];
  (*BS log:PDE=Sum[(r-Σ[i]^2/2)dS[i],{i,n}]+
     1/2Sum[Σ[i]Σ[j]Π[i,j]dS[i]dS[j],{i,n},{j,n}];*)
  Σ[i_] := σ; Π[i_, j_] := {1  i == j
                            ρ  True ;
  (*PDE coefficient function*)
  Vars = Table[dS[i], {i, n}];
  PDE = CoefficientList[PDE, Vars];
  PDECoeff[index_] :=
    {-r                                Total[Abs[index]] == 0
     Part @@ Prepend[index + 1, PDE]   Max[index - Dimensions[PDE]] < 0 ;
     0                                 True
  (*Generate MultiIndices for derivative terms*)
  multiIndices =
   Flatten[Table[Reverse [Compositions[i, n]], {i, DerivativeOrder}], 1];
  numPD = Length[multiIndices];
  denoms = 1 / Times @@ # & /@ (multiIndices!);
  (*Fill d and b *)
  Off[Power::indet];
  d = Table[denoms[[k]]
     Times @@ ((h StencilPoints[[i]]) ^ multiIndices[[k]] /. Indeterminate → 1),
    {k, numPD}, {i, nPoints}];
  b = Table[PDECoeff[multiIndices[[k]]], {k, numPD}];
  On[Power::indet];
  {d, b}
 ];
GenDiagonals[sys_, w_] :=
 Module[{StencilPoints = sys[[1]], DerivativeOrder = sys[[2]], n = sys[[3]], dS,
   i, j, h, S, σ, Σ, ρ, Π, r, m, PDE, h2, Vars, numPD, multiIndices, denoms,
   c, nn, Smin, sMax, k, PDECoeff, d, b, nPoints, Weights, Res, indices},
  Res = {};
  nPoints = Length[StencilPoints];
  nn = 10; Smin = 50; sMax = 150;
  h = (150 - 50) / nn; h2 = h / 8;
  σ = 0.2; ρ = 0.4; r = 0.03;
  PDE = Sum[r S[i] dS[i], {i, n}] +
```

```
    1 / 2 Sum[Σ[i] Σ[j] Π[i, j] S[i] S[j] dS[i] dS[j], {i, n}, {j, n}];
  (*BS log:PDE=Sum[(r-Σ[i]^2/2)dS[i],{i,n}]+
     1/2Sum[Σ[i]Σ[j]Π[i,j]dS[i]dS[j],{i,n},{j,n}];*)
  Σ[i_] := σ; Π[i_, j_] := { 1   i == j
                           { ρ   True ;
  (*PDE coefficient function*)
  Vars = Table[dS[i], {i, n}];
  PDE = CoefficientList[PDE, Vars];
  PDECoeff[index_] :=
   { -r                              Total[Abs[index]] == 0
   { Part @@ Prepend[index + 1, PDE]  Max[index - Dimensions[PDE]] < 0 ;
   { 0                               True
  (*Generate MultiIndices for derivative terms*)
  multiIndices =
   Flatten[Table[Reverse [Compositions[i, n]], {i, DerivativeOrder}], 1];
  numPD = Length[multiIndices];
  denoms = 1 / Times @@ # & /@ (multiIndices!);
  (*Fill d and b *)
  indices = Transpose[Reverse[Transpose[Tuples[Table[i, {i, nn}], n]]]];
  Off[Power::indet];
  Do[For[j = 1, j ≤ n, j++, S[j] = Smin + h ind[[j]]];
    d = Table[denoms[[k]] Times @@ ((h2 StencilPoints[[i]]) ^ multiIndices[[k]] /.
          Indeterminate → 1), {k, numPD}, {i, nPoints}];
    b = Table[PDECoeff[multiIndices[[k]]], {k, numPD}];
    (*
    c=LeastSquares[d,b];
    Weights[ind]=c;
    AppendTo[Res,Norm[d.c-b]];
    *)
    (Weights[ind] = #[[2]]; AppendTo[Res, #[[1]]];) &[NNLSJo[d, b]];
    , {ind, indices}]
   On[Power::indet];
  {Res, Weights}
 ];
SplitOperator[wei_, right_, left_, dir_, indices_] :=
 Module[{i, ind, test, d, trisys, start},
  trisys = {};
  Do[test = ind - dir;
   If[Min[test] < 1 || Max[test] > 10,
    test = ind; start = ind; d = {};
    While[Min[test] > 0 && Max[test] < 11,
         AppendTo[d,
      {#[[left]], -0.03 / 3 - #[[left]] - #[[right]], #[[right]]} &[wei[test]]];
     test += dir];
    AppendTo[trisys, {start, d}]
   ], {ind, indices}];
  trisys
 ];
```

```
CheckWeights[StencilWeights_, sys_] := Module[{d, b, c, m},
   {d, b} = GenDandB[sys];
   c = LeastSquares[d, b];
   Print["Norm: ", Norm[d.StencilWeights - b]];
   Print["LS-Norm: ", Norm[d.c - b]];
   c];
GenWeights[sys_] := Module[{d, b},
   {d, b} = GenDandB[sys];
   NNLSJo[d, b]
  ];
GenWeights[sys_, s_] := Module[{d, b},
   {d, b} = GenDandB[sys];
   NNLSJo[d, b, s]
  ];
PDECoefMatrix[S_] := Module[{σ, ρ, r, dS, PDE, Π, Σ, Vars, PDECoeff, i, n = Length[S]},
   σ = 0.2; ρ = 0.9; r = 0.03;
    PDE = Sum[r S[[i]] dS[i], {i, n}] +
      1 / 2 Sum[Σ[i] Σ[j] Π[i, j] S[[i]] S[[j]] dS[i] dS[j], {i, n}, {j, n}];
   (*BS log:PDE=Sum[(r-Σ[i]^2/2)dS[i],{i,n}]+
      1/2Sum[Σ[i]Σ[j]Π[i,j]dS[i]dS[j],{i,n},{j,n}];*)
   Σ[i_] := σ; Π[i_, j_] := {1  i == j
                             ρ  True ;
   (*PDE coefficient function*)
   Vars = Table[dS[i], {i, n}];
   PDE = CoefficientList[PDE, Vars];
   PDECoeff[index_] :=
    {-r                                Total[Abs[index]] == 0
     Part @@ Prepend[index + 1, PDE]   Max[index - Dimensions[PDE]] < 0 ;
     0                                 True
   (*Check type of pde*)
   Table[1 / (2 - KroneckerDelta[i, j]) * PDECoeff[
      Table[KroneckerDelta[i, k] + KroneckerDelta[j, k], {k, n}]], {i, n}, {j, n}]];
```