

SunwayMatrixSolver -- SMS

Generated by Doxygen 1.8.15

1 Namespace Index	1
1.1 Namespace List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 File Index	7
4.1 File List	7
5 Namespace Documentation	9
5.1 SMS Namespace Reference	9
5.1.1 Typedef Documentation	11
5.1.1.1 labelField	11
5.1.1.2 scalarField	11
5.1.2 Function Documentation	11
5.1.2.1 constructInterfacesFromOpenFOAM()	11
5.1.2.2 constructMatrixFromOpenFOAM()	12
5.1.2.3 constructVectorFromOpenFOAM()	12
5.1.2.4 construct_solver_mg__()	12
5.1.2.5 construct_solver_pbicgstab__()	13
5.1.2.6 construct_sw_matrix__()	13
5.1.2.7 construct_sw_matrix_interfaces__()	14
5.1.2.8 coo2ldu()	14
5.1.2.9 csr2ldu()	15
5.1.2.10 dot()	16
5.1.2.11 fill_sw_matrix_coefficients__()	16
5.1.2.12 fill_sw_matrix_interfaces_coefficients__()	17
5.1.2.13 mag()	17
5.1.2.14 max()	18
5.1.2.15 min()	18
5.1.2.16 printInterfaces()	18
5.1.2.17 printLDUMatrix()	19
5.1.2.18 printVector()	20
5.1.2.19 reduce()	20
5.1.2.20 sleep()	21
5.1.2.21 sortData()	21
5.1.2.22 sw_matrix_destroy__()	22
5.1.2.23 sw_solve_mg__()	22
5.1.2.24 sw_solve_pbicgstab__()	22
5.1.2.25 sw_solver_destroy_mg__()	22
5.1.2.26 sw_solver_destroy_pbicgstab__()	22

5.1.2.27 <code>sw_solver_mg_set_maxiter_()</code>	22
5.1.2.28 <code>sw_solver_mg_set_miniter_()</code>	23
5.1.2.29 <code>sw_solver_mg_set_nfinestsweeps_()</code>	23
5.1.2.30 <code>sw_solver_mg_set_npostsweeps_()</code>	23
5.1.2.31 <code>sw_solver_mg_set_npresweeps_()</code>	23
5.1.2.32 <code>sw_solver_mg_set_reltol_()</code>	23
5.1.2.33 <code>sw_solver_mg_set_tol_()</code>	23
5.1.2.34 <code>sw_solver_pbicgstab_set_maxiter_()</code>	24
5.1.2.35 <code>sw_solver_pbicgstab_set_miniter_()</code>	24
5.1.2.36 <code>sw_solver_pbicgstab_set_precond_()</code>	24
5.1.2.37 <code>sw_solver_pbicgstab_set_reltol_()</code>	24
5.1.2.38 <code>sw_solver_pbicgstab_set_tol_()</code>	24
6 Class Documentation	25
6.1 SMS::matrix::agglomeration Class Reference	25
6.1.1 Constructor & Destructor Documentation	26
6.1.1.1 <code>agglomeration()</code>	26
6.1.1.2 <code>~agglomeration()</code>	26
6.1.2 Member Function Documentation	26
6.1.2.1 <code>agglomerate()</code>	26
6.1.2.2 <code>agglomerateMatrix()</code>	26
6.1.2.3 <code>coarseMatrixLevels()</code>	27
6.1.2.4 <code>faceRestrictAddressing()</code>	27
6.1.2.5 <code>prolongField()</code>	27
6.1.2.6 <code>restrictAddressing()</code>	27
6.1.2.7 <code>restrictFaceField()</code>	28
6.1.2.8 <code>restrictField()</code>	28
6.1.2.9 <code>SET_maxLevels()</code>	29
6.1.2.10 <code>SET_nCellsInCoarsestLevel()</code>	29
6.1.2.11 <code>size()</code>	29
6.2 SMS::chebySmoother Class Reference	30
6.2.1 Constructor & Destructor Documentation	31
6.2.1.1 <code>chebySmoother()</code>	31
6.2.2 Member Function Documentation	31
6.2.2.1 <code>init()</code>	31
6.2.2.2 <code>smooth() [1/2]</code>	31
6.2.2.3 <code>smooth() [2/2]</code>	32
6.2.3 Member Data Documentation	32
6.2.3.1 <code>boostFactorCheby_</code>	32
6.2.3.2 <code>eigFirstTimeComputed_</code>	32
6.2.3.3 <code>eigRatioCheby_</code>	32
6.2.3.4 <code>eigRatioCoarest_</code>	33

6.2.3.5 maxEigPCG_	33
6.2.3.6 nDiagPCGs_	33
6.2.3.7 preSmoothUsing_	33
6.3 SMS::eigenDiagPCG Class Reference	33
6.3.1 Constructor & Destructor Documentation	34
6.3.1.1 eigenDiagPCG()	34
6.3.2 Member Function Documentation	34
6.3.2.1 allocateSym2D()	34
6.3.2.2 computeMaxEig()	34
6.3.2.3 computeValueForMatrix()	35
6.3.2.4 deleteSym2D()	35
6.3.2.5 determineEigRange()	35
6.3.2.6 diagPCGLoops()	35
6.3.2.7 h14Sturm()	36
6.3.2.8 maxEigenValue()	36
6.3.3 Member Data Documentation	37
6.3.3.1 maxEigenValue_	37
6.4 SMS::interfaces Class Reference	37
6.4.1 Constructor & Destructor Documentation	39
6.4.1.1 interfaces() [1/2]	39
6.4.1.2 interfaces() [2/2]	39
6.4.1.3 ~interfaces()	39
6.4.2 Member Function Documentation	39
6.4.2.1 initMatrixInterfaces()	39
6.4.2.2 patchList() [1/2]	40
6.4.2.3 patchList() [2/2]	40
6.4.2.4 size()	40
6.4.2.5 updateMatrixInterfaces()	41
6.4.3 Member Data Documentation	41
6.4.3.1 destRank_	42
6.4.3.2 locPosition_	42
6.4.3.3 patches_	42
6.4.3.4 recvBuffer_	42
6.4.3.5 sendBuffer_	42
6.4.3.6 sendRecvRequests_	42
6.5 SMS::labelPair Class Reference	42
6.5.1 Constructor & Destructor Documentation	43
6.5.1.1 labelPair()	43
6.5.2 Member Function Documentation	43
6.5.2.1 facel() [1/2]	43
6.5.2.2 facel() [2/2]	44
6.5.2.3 first() [1/2]	44

6.5.2.4 first() [2/2]	44
6.5.2.5 operator==()	44
6.5.2.6 second() [1/2]	45
6.5.2.7 second() [2/2]	45
6.5.3 Member Data Documentation	45
6.5.3.1 facel_	45
6.5.3.2 first_	45
6.5.3.3 second_	45
6.6 SMS::lduAgglomeration Class Reference	46
6.6.1 Constructor & Destructor Documentation	47
6.6.1.1 lduAgglomeration()	47
6.6.1.2 ~lduAgglomeration()	47
6.6.2 Member Function Documentation	47
6.6.2.1 agglomerate() [1/2]	48
6.6.2.2 agglomerate() [2/2]	48
6.6.2.3 agglomerateLduAddressing()	49
6.6.2.4 agglomerateMatrix()	50
6.6.2.5 coarseMatrixLevels()	51
6.6.2.6 combineLevels()	51
6.6.2.7 compactLevels()	51
6.6.2.8 continueAgglomerating()	51
6.6.2.9 faceRestrictAddressing()	51
6.6.2.10 matrixLevel()	52
6.6.2.11 restrictAddressing()	52
6.6.2.12 SET_maxLevels()	52
6.6.2.13 SET_nCellsInCoarsestLevel()	52
6.6.2.14 size()	52
6.6.3 Member Data Documentation	53
6.6.3.1 coarseMatrixLevels_	53
6.6.3.2 faceRestrictAddressing_	53
6.6.3.3 finestMatrix_	53
6.6.3.4 forward_	53
6.6.3.5 maxLevels_	53
6.6.3.6 mergeLevels_	53
6.6.3.7 nCells_	53
6.6.3.8 nCellsInCoarsestLevel_	54
6.6.3.9 nCreatedLevels_	54
6.6.3.10 patchFaceRestrictAddressing_	54
6.6.3.11 restrictAddressing_	54
6.7 SMS::lduDiagPrecond Class Reference	54
6.7.1 Constructor & Destructor Documentation	55
6.7.1.1 lduDiagPrecond()	55

6.7.1.2 ~IduDiagPrecond()	56
6.7.2 Member Function Documentation	56
6.7.2.1 precondition()	56
6.7.3 Member Data Documentation	56
6.7.3.1 rD_	57
6.8 SMS::IduDICPrecond Class Reference	57
6.8.1 Constructor & Destructor Documentation	59
6.8.1.1 IduDICPrecond()	59
6.8.1.2 ~IduDICPrecond()	59
6.8.2 Member Function Documentation	59
6.8.2.1 calcReciprocalD()	59
6.8.2.2 precondition()	60
6.8.3 Member Data Documentation	60
6.8.3.1 APtr_	60
6.8.3.2 rD_	60
6.9 SMS::IduDILUPrecond Class Reference	61
6.9.1 Constructor & Destructor Documentation	63
6.9.1.1 IduDILUPrecond()	63
6.9.1.2 ~IduDILUPrecond()	63
6.9.2 Member Function Documentation	63
6.9.2.1 calcReciprocalD()	63
6.9.2.2 precondition()	64
6.9.3 Member Data Documentation	64
6.9.3.1 APtr_	64
6.9.3.2 rD_	64
6.10 SMS::IduGaussSeidelSmoothen Class Reference	65
6.10.1 Constructor & Destructor Documentation	65
6.10.1.1 IduGaussSeidelSmoothen()	66
6.10.1.2 ~IduGaussSeidelSmoothen()	66
6.10.2 Member Function Documentation	66
6.10.2.1 init()	66
6.10.2.2 smooth() [1/2]	66
6.10.2.3 smooth() [2/2]	67
6.11 SMS::IduMatrix Class Reference	67
6.11.1 Constructor & Destructor Documentation	71
6.11.1.1 IduMatrix() [1/5]	71
6.11.1.2 IduMatrix() [2/5]	71
6.11.1.3 IduMatrix() [3/5]	71
6.11.1.4 IduMatrix() [4/5]	71
6.11.1.5 IduMatrix() [5/5]	71
6.11.1.6 ~IduMatrix()	72
6.11.2 Member Function Documentation	72

6.11.2.1 calcLosort()	72
6.11.2.2 calcLosortStart()	72
6.11.2.3 calcOwnerStart()	73
6.11.2.4 diag() [1/2]	73
6.11.2.5 diag() [2/2]	73
6.11.2.6 initInterfaces()	74
6.11.2.7 losortAddr()	74
6.11.2.8 losortStartAddr()	74
6.11.2.9 lower()	74
6.11.2.10 lowerAddr()	75
6.11.2.11 matrixInterfaces() [1/3]	75
6.11.2.12 matrixInterfaces() [2/3]	76
6.11.2.13 matrixInterfaces() [3/3]	76
6.11.2.14 nCells()	76
6.11.2.15 ownerStartAddr()	77
6.11.2.16 SET_diag() [1/2]	77
6.11.2.17 SET_diag() [2/2]	77
6.11.2.18 SET_lower() [1/2]	78
6.11.2.19 SET_lower() [2/2]	78
6.11.2.20 SET_lowerAddr()	78
6.11.2.21 SET_upper() [1/2]	79
6.11.2.22 SET_upper() [2/2]	79
6.11.2.23 SET_upperAddr()	79
6.11.2.24 setMatrixCoefficients() [1/2]	80
6.11.2.25 setMatrixCoefficients() [2/2]	80
6.11.2.26 setMatrixTopology() [1/2]	81
6.11.2.27 setMatrixTopology() [2/2]	81
6.11.2.28 setSymm()	81
6.11.2.29 size() [1/2]	82
6.11.2.30 size() [2/2]	82
6.11.2.31 spMV()	82
6.11.2.32 symm()	83
6.11.2.33 updateInterfaces()	84
6.11.2.34 upper()	84
6.11.2.35 upperAddr()	85
6.11.3 Member Data Documentation	85
6.11.3.1 diagPtr_	85
6.11.3.2 interfacesPtr_	85
6.11.3.3 losortPtr_	85
6.11.3.4 losortStartPtr_	86
6.11.3.5 lowerAddrPtr_	86
6.11.3.6 lowerPtr_	86

6.11.3.7 nCells_	86
6.11.3.8 ownerStartPtr_	86
6.11.3.9 upperAddrPtr_	86
6.11.3.10 upperPtr_	86
6.12 SMS::matrix Class Reference	87
6.12.1 Constructor & Destructor Documentation	87
6.12.1.1 ~matrix()	87
6.12.2 Member Function Documentation	88
6.12.2.1 diag()	88
6.12.2.2 initInterfaces()	88
6.12.2.3 matrixInterfaces()	88
6.12.2.4 size()	88
6.12.2.5 spMV()	89
6.12.2.6 symm()	89
6.12.2.7 updateInterfaces()	90
6.13 SMS::MGPrecond Class Reference	90
6.13.1 Constructor & Destructor Documentation	91
6.13.1.1 MGPrecond()	91
6.13.1.2 ~MGPrecond()	91
6.13.2 Member Function Documentation	91
6.13.2.1 precondition()	92
6.13.2.2 set_nVcycles()	92
6.13.3 Member Data Documentation	92
6.13.3.1 APtr_	92
6.13.3.2 MGSolverPtr_	92
6.13.3.3 nVcycles_	92
6.14 SMS::MGSolver Class Reference	93
6.14.1 Constructor & Destructor Documentation	94
6.14.1.1 MGSolver()	94
6.14.1.2 ~MGSolver()	94
6.14.2 Member Function Documentation	94
6.14.2.1 initSmoothers()	94
6.14.2.2 initVcycle()	95
6.14.2.3 scalingFactor()	95
6.14.2.4 SET_cacheAgglomeration()	96
6.14.2.5 SET_nFinestSweeps()	96
6.14.2.6 SET_nPostSweeps()	96
6.14.2.7 SET_nPreSweeps()	96
6.14.2.8 SET_scaleCorrection()	96
6.14.2.9 solve()	97
6.14.2.10 solveCoarsestLevel()	98
6.14.2.11 Vcycle()	98

6.14.3 Member Data Documentation	99
6.14.3.1 agglomeration_	99
6.14.3.2 cacheAgglomeration_	99
6.14.3.3 finestMatrix_	99
6.14.3.4 nFinestSweeps_	99
6.14.3.5 nPostSweeps_	99
6.14.3.6 nPreSweeps_	99
6.14.3.7 scaleCorrection_	99
6.14.3.8 smoothers_	100
6.15 SMS::patch Class Reference	100
6.15.1 Constructor & Destructor Documentation	101
6.15.1.1 patch()	101
6.15.1.2 ~patch()	101
6.15.2 Member Function Documentation	101
6.15.2.1 faceCells() [1/2]	102
6.15.2.2 faceCells() [2/2]	102
6.15.2.3 faceRestrictAddressing() [1/2]	102
6.15.2.4 faceRestrictAddressing() [2/2]	103
6.15.2.5 myProcNo() [1/2]	103
6.15.2.6 myProcNo() [2/2]	103
6.15.2.7 neighbProcNo() [1/2]	104
6.15.2.8 neighbProcNo() [2/2]	104
6.15.2.9 patchCoeffs() [1/3]	104
6.15.2.10 patchCoeffs() [2/3]	105
6.15.2.11 patchCoeffs() [3/3]	105
6.15.2.12 size() [1/2]	106
6.15.2.13 size() [2/2]	106
6.15.3 Member Data Documentation	106
6.15.3.1 faceCellsPtr_	106
6.15.3.2 faceRestrictAddressingPtr_	106
6.15.3.3 myProcNo_	107
6.15.3.4 neighbProcNo_	107
6.15.3.5 patchCoeffsPtr_	107
6.15.3.6 size_	107
6.16 SMS::PBiCGStab Class Reference	107
6.16.1 Constructor & Destructor Documentation	108
6.16.1.1 PBiCGStab() [1/3]	108
6.16.1.2 PBiCGStab() [2/3]	108
6.16.1.3 PBiCGStab() [3/3]	109
6.16.1.4 ~PBiCGStab()	109
6.16.2 Member Function Documentation	109
6.16.2.1 SET_preconditioner()	109

6.16.2.2 solve()	109
6.16.3 Member Data Documentation	110
6.16.3.1 deletePrecondPtr_	110
6.16.3.2 precondPtr_	111
6.17 SMS::PCG Class Reference	111
6.17.1 Constructor & Destructor Documentation	112
6.17.1.1 PCG() [1/2]	112
6.17.1.2 PCG() [2/2]	112
6.17.1.3 ~PCG()	112
6.17.2 Member Function Documentation	112
6.17.2.1 solve()	113
6.17.3 Member Data Documentation	113
6.17.3.1 deletePrecondPtr_	114
6.17.3.2 precondPtr_	114
6.18 SMS::matrix::preconditioner Class Reference	114
6.18.1 Constructor & Destructor Documentation	114
6.18.1.1 ~preconditioner()	114
6.18.2 Member Function Documentation	114
6.18.2.1 precondition()	115
6.19 SMS::PtrList< T > Class Template Reference	115
6.19.1 Constructor & Destructor Documentation	116
6.19.1.1 PtrList() [1/2]	116
6.19.1.2 PtrList() [2/2]	116
6.19.1.3 ~PtrList()	117
6.19.2 Member Function Documentation	117
6.19.2.1 isEmpty()	117
6.19.2.2 operator[]()	117
6.19.2.3 removeLevel()	117
6.19.2.4 SET_size()	117
6.19.2.5 setLevel()	118
6.19.2.6 size()	118
6.19.3 Member Data Documentation	118
6.19.3.1 ptrs_	118
6.19.3.2 size_	119
6.20 SMS::matrix::smoother Class Reference	119
6.20.1 Constructor & Destructor Documentation	119
6.20.1.1 smoother()	119
6.20.1.2 ~smoother()	120
6.20.2 Member Function Documentation	120
6.20.2.1 init()	120
6.20.2.2 smooth()	120
6.21 SMS::smsMPI Class Reference	120

6.21.1 Member Function Documentation	121
6.21.1.1 exitMPI()	121
6.21.1.2 init()	121
6.21.1.3 myProcNo()	121
6.21.1.4 nProcs()	121
6.21.1.5 parRun()	122
6.21.1.6 smsLabel()	122
6.21.1.7 smsScalar()	122
6.21.2 Member Data Documentation	122
6.21.2.1 myProcNo_	122
6.21.2.2 nProcs_	122
6.21.2.3 parRun_	122
6.21.2.4 smsLabel_	122
6.21.2.5 smsScalar_	123
6.22 SMS::matrix::solver Class Reference	123
6.22.1 Constructor & Destructor Documentation	124
6.22.1.1 solver()	124
6.22.1.2 ~solver()	124
6.22.2 Member Function Documentation	124
6.22.2.1 maxIter()	124
6.22.2.2 minIter()	124
6.22.2.3 normFactor()	124
6.22.2.4 relTol()	125
6.22.2.5 SET_maxIter()	125
6.22.2.6 SET_minIter()	125
6.22.2.7 SET_relTol()	126
6.22.2.8 SET_tolerance()	126
6.22.2.9 solve()	127
6.22.2.10 tolerance()	127
6.22.3 Member Data Documentation	127
6.22.3.1 maxIter_	127
6.22.3.2 minIter_	127
6.22.3.3 relTol_	127
6.22.3.4 tolerance_	127
6.23 SMS::matrix::solverPerformance Class Reference	128
6.23.1 Constructor & Destructor Documentation	128
6.23.1.1 solverPerformance() [1/2]	128
6.23.1.2 solverPerformance() [2/2]	128
6.23.2 Member Function Documentation	129
6.23.2.1 checkConvergence() [1/3]	129
6.23.2.2 checkConvergence() [2/3]	129
6.23.2.3 checkConvergence() [3/3]	129

6.23.2.4 checkSingularity()	130
6.23.2.5 converged()	130
6.23.2.6 finalResidual() [1/2]	130
6.23.2.7 finalResidual() [2/2]	131
6.23.2.8 initialResidual() [1/2]	131
6.23.2.9 initialResidual() [2/2]	131
6.23.2.10 nIterations() [1/2]	131
6.23.2.11 nIterations() [2/2]	132
6.23.2.12 previousResidual() [1/2]	132
6.23.2.13 previousResidual() [2/2]	132
6.23.2.14 singular()	132
6.23.3 Member Data Documentation	132
6.23.3.1 converged_	132
6.23.3.2 finalResidual_	133
6.23.3.3 initialResidual_	133
6.23.3.4 nIterations_	133
6.23.3.5 previousResidual_	133
6.23.3.6 singular_	133
6.24 Timer Class Reference	133
6.24.1 Constructor & Destructor Documentation	134
6.24.1.1 ~Timer()	134
6.24.2 Member Function Documentation	134
6.24.2.1 closeTimer()	134
6.24.2.2 endTimer()	135
6.24.2.3 maxTimeSum()	135
6.24.2.4 openTimer()	135
6.24.2.5 printTimer() [1/2]	135
6.24.2.6 printTimer() [2/2]	136
6.24.2.7 startTimer()	136
6.24.2.8 Vector_Variance()	136
6.24.3 Member Data Documentation	136
6.24.3.1 _OpenTimer	137
6.24.3.2 count	137
6.24.3.3 timeStart	137
6.24.3.4 timeSum	137
6.25 SMS::Vector< T > Class Template Reference	137
6.25.1 Constructor & Destructor Documentation	138
6.25.1.1 Vector() [1/6]	138
6.25.1.2 Vector() [2/6]	139
6.25.1.3 Vector() [3/6]	139
6.25.1.4 Vector() [4/6]	139
6.25.1.5 Vector() [5/6]	139

6.25.1.6 <code>Vector()</code> [6/6]	139
6.25.1.7 <code>~Vector()</code>	139
6.25.2 Member Function Documentation	140
6.25.2.1 <code>begin()</code>	140
6.25.2.2 <code>end()</code>	141
6.25.2.3 <code>operator *()</code>	141
6.25.2.4 <code>operator *=()</code>	141
6.25.2.5 <code>operator+()</code>	141
6.25.2.6 <code>operator+=()</code> [1/2]	141
6.25.2.7 <code>operator+=()</code> [2/2]	141
6.25.2.8 <code>operator-()</code>	142
6.25.2.9 <code>operator-=()</code> [1/2]	142
6.25.2.10 <code>operator-=()</code> [2/2]	142
6.25.2.11 <code>operator=()</code> [1/2]	142
6.25.2.12 <code>operator=()</code> [2/2]	142
6.25.2.13 <code>operator[]()</code> [1/2]	142
6.25.2.14 <code>operator[]()</code> [2/2]	143
6.25.2.15 <code>SET_size()</code>	143
6.25.2.16 <code>SET_zero()</code>	143
6.25.2.17 <code>size()</code>	144
6.25.2.18 <code>Sum()</code>	145
6.25.2.19 <code>SumMag()</code>	145
6.25.2.20 <code>SumSqr()</code>	145
6.25.2.21 <code>SumSqrt()</code>	145
6.25.2.22 <code>values()</code>	146
6.25.3 Member Data Documentation	146
6.25.3.1 <code>length_</code>	146
6.25.3.2 <code>values_</code>	146
7 File Documentation	147
7.1 SunwayMatrixSolver/src/base/mpi/smsMPI.cpp File Reference	147
7.2 SunwayMatrixSolver/src/base/mpi/smsMPI.hpp File Reference	147
7.2.1 Detailed Description	149
7.2.2 Macro Definition Documentation	149
7.2.2.1 <code>MPI_LABEL</code>	149
7.2.2.2 <code>MPI_SCALAR</code>	149
7.2.2.3 <code>MYID</code>	149
7.2.2.4 <code>NPROCS</code>	149
7.2.2.5 <code>PARRUN</code>	150
7.3 SunwayMatrixSolver/src/base/PtrList/PtrList.hpp File Reference	150
7.3.1 Detailed Description	151
7.4 SunwayMatrixSolver/src/base/sms.cpp File Reference	151

7.5 SunwayMatrixSolver/src/base/sms.hpp File Reference	152
7.5.1 Detailed Description	153
7.5.2 Macro Definition Documentation	153
7.5.2.1 ALLOCATE_POINTER	153
7.5.2.2 CHECK_POINTER	154
7.5.2.3 COUT	154
7.5.2.4 DELETE_OBJECT_POINTER	154
7.5.2.5 DELETE_POINTER	154
7.5.2.6 ENDL	154
7.5.2.7 ERROR_EXIT	155
7.5.2.8 EXIT	155
7.5.2.9 forall	155
7.5.2.10 GREAT	155
7.5.2.11 SMALL	155
7.5.2.12 VGREAT	155
7.5.2.13 VSMALL	155
7.5.3 Variable Documentation	155
7.5.3.1 ERROR_EXIT	156
7.6 SunwayMatrixSolver/src/base/vectors/smsVector.hpp File Reference	156
7.6.1 Detailed Description	157
7.7 SunwayMatrixSolver/src/base/vectors/smsVectorOperations.cpp File Reference	157
7.8 SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/csrAgglomeration.cpp File Reference	158
7.9 SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/csrAgglomeration.hpp File Reference	158
7.10 SunwayMatrixSolver/src/matrix/csrMatrix/csrMatrix.cpp File Reference	158
7.11 SunwayMatrixSolver/src/matrix/csrMatrix/csrMatrix.hpp File Reference	158
7.12 SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/eigenDiagPCG.cpp File Reference	159
7.13 SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/eigenDiagPCG.hpp File Reference	159
7.14 SunwayMatrixSolver/src/matrix/interfaces/interfaces.cpp File Reference	161
7.15 SunwayMatrixSolver/src/matrix/interfaces/interfaces.hpp File Reference	162
7.16 SunwayMatrixSolver/src/matrix/interfaces/labelPair/labelPair.cpp File Reference	164
7.17 SunwayMatrixSolver/src/matrix/interfaces/labelPair/labelPair.hpp File Reference	164
7.18 SunwayMatrixSolver/src/matrix/interfaces/patch/patch.cpp File Reference	165
7.19 SunwayMatrixSolver/src/matrix/interfaces/patch/patch.hpp File Reference	166
7.20 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerate.cpp File Reference	167
7.21 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerateMatrix.cpp File Reference	168
7.22 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomeration.cpp File Reference	169
7.23 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomeration.hpp File Reference	170
7.24 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerationAddressing.cpp File Reference	172
7.25 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerationCombineLevels.cpp File Reference	172
7.26 SunwayMatrixSolver/src/matrix/lduMatrix/lduMatrix.cpp File Reference	173
7.27 SunwayMatrixSolver/src/matrix/lduMatrix/lduMatrix.hpp File Reference	174

7.28 SunwayMatrixSolver/src/matrix/lduMatrix/lduMatrixSpMV.cpp File Reference	176
7.29 SunwayMatrixSolver/src/matrix/smsMatrix.hpp File Reference	176
7.30 SunwayMatrixSolver/src/matrix/smsMatrixAgglomeration.cpp File Reference	178
7.31 SunwayMatrixSolver/src/matrix/smsMatrixPrecond.cpp File Reference	178
7.32 SunwayMatrixSolver/src/matrix/smsMatrixSolver.cpp File Reference	179
7.33 SunwayMatrixSolver/src/matrix/smsMatrixTest.cpp File Reference	180
7.34 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDiagonal/csrDiagPrecond.cpp File Reference	181
7.35 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDiagonal/csrDiagPrecond.hpp File Reference	181
7.36 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDIC/csrDICPrecond.cpp File Reference	181
7.37 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDIC/csrDICPrecond.hpp File Reference	181
7.38 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDILU/csrDILUPrecond.cpp File Reference	182
7.39 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDILU/csrDILUPrecond.hpp File Reference	182
7.40 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDiagonal/lduDiagPrecond.cpp File Reference	182
7.41 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDiagonal/lduDiagPrecond.hpp File Reference	183
7.42 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDIC/lduDICPrecond.cpp File Reference	185
7.43 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDIC/lduDICPrecond.hpp File Reference	186
7.44 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDILU/lduDILUPrecond.cpp File Reference	187
7.45 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDILU/lduDILUPrecond.hpp File Reference	188
7.46 SunwayMatrixSolver/src/preconditioners/MGPreconditioner/MGPreconditioner.cpp File Reference	190
7.47 SunwayMatrixSolver/src/preconditioners/MGPreconditioner/MGPreconditioner.hpp File Reference	190
7.48 SunwayMatrixSolver/src/smoothers/chebySmoothening/chebySmoothening.cpp File Reference	192
7.49 SunwayMatrixSolver/src/smoothers/chebySmoothening/chebySmoothening.hpp File Reference	193
7.50 SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmoothening/csrGaussSeidelSmoothening.cpp File Reference	195
7.51 SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmoothening/csrGaussSeidelSmoothening.hpp File Reference	195
7.52 SunwayMatrixSolver/src/smoothers/lduSmoothers/lduGaussSeidelSmoothening/lduGaussSeidelSmoothening.cpp File Reference	195
7.53 SunwayMatrixSolver/src/smoothers/lduSmoothers/lduGaussSeidelSmoothening/lduGaussSeidelSmoothening.hpp File Reference	196
7.54 SunwayMatrixSolver/src/solvers/MG/MG.cpp File Reference	198
7.55 SunwayMatrixSolver/src/solvers/MG/MG.hpp File Reference	199
7.56 SunwayMatrixSolver/src/solvers/MG/MGSolve.cpp File Reference	200
7.57 SunwayMatrixSolver/src/solvers/MG/scalingFactor.cpp File Reference	201

7.58 SunwayMatrixSolver/src/solvers/MG/solveCoarsestLevel.cpp File Reference	201
7.59 SunwayMatrixSolver/src/solvers/MG/Vcycle.cpp File Reference	202
7.60 SunwayMatrixSolver/src/solvers/PBiCGStab/PBiCGStab.cpp File Reference	203
7.61 SunwayMatrixSolver/src/solvers/PBiCGStab/PBiCGStab.hpp File Reference	204
7.62 SunwayMatrixSolver/src/solvers/PCG/PCG.cpp File Reference	206
7.63 SunwayMatrixSolver/src/solvers/PCG/PCG.hpp File Reference	207
7.64 SunwayMatrixSolver/tools/compass/compassInterface.cpp File Reference	208
7.64.1 Macro Definition Documentation	208
7.64.1.1 PTR2OBJ	209
7.65 SunwayMatrixSolver/tools/compass/compassInterface.hpp File Reference	209
7.66 SunwayMatrixSolver/tools/fortranInterface/fortranInterface.cpp File Reference	210
7.66.1 Macro Definition Documentation	211
7.66.1.1 RETURN_VALUE	212
7.66.2 Function Documentation	212
7.66.2.1 coo2ldumatrixcreat_()	212
7.66.2.2 csr2ldumatrixcreat_()	212
7.66.2.3 ldumatrixcreat_()	213
7.66.2.4 matrixinterfacescreat_()	213
7.66.2.5 mgsolversolve_()	214
7.66.2.6 pbicgstabsolversolve_()	215
7.66.2.7 pcgsolversolve_()	215
7.67 SunwayMatrixSolver/tools/fortranInterface/fortranInterface.h File Reference	216
7.67.1 Function Documentation	217
7.67.1.1 coo2ldumatrixcreat_()	217
7.67.1.2 csr2ldumatrixcreat_()	217
7.67.1.3 ldumatrixcreat_()	218
7.67.1.4 matrixinterfacescreat_()	218
7.67.1.5 mgsolversolve_()	218
7.67.1.6 pbicgstabsolversolve_()	218
7.67.1.7 pcgsolversolve_()	219
7.68 SunwayMatrixSolver/tools/matrixConversion/coo2csr.cpp File Reference	219
7.69 SunwayMatrixSolver/tools/matrixConversion/coo2ldu.cpp File Reference	219
7.70 SunwayMatrixSolver/tools/matrixConversion/csr2coo.cpp File Reference	220
7.71 SunwayMatrixSolver/tools/matrixConversion/matrixConversion.cpp File Reference	220
7.72 SunwayMatrixSolver/tools/matrixConversion/matrixConversion.hpp File Reference	221
7.73 SunwayMatrixSolver/tools/printData/printSMS/printSMS.cpp File Reference	223
7.74 SunwayMatrixSolver/tools/printData/printSMS/printSMS.hpp File Reference	224
7.74.1 Macro Definition Documentation	226
7.74.1.1 FILECLOSE	226
7.74.1.2 FILEOPEN	226
7.75 SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.cpp File Reference	226
7.76 SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.hpp File Reference	227

7.77 SunwayMatrixSolver/tools/timers/Timers.cpp File Reference	229
7.77.1 Function Documentation	230
7.77.1.1 getSystemTime()	230
7.78 SunwayMatrixSolver/tools/timers/Timers.hpp File Reference	230
Index	233

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

SMS	9
---------------	---

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

SMS::matrix::agglomeration	25
SMS::lduAgglomeration	46
SMS::eigenDiagPCG	33
SMS::interfaces	37
SMS::labelPair	42
SMS::matrix	87
SMS::lduMatrix	67
SMS::patch	100
SMS::matrix::preconditioner	114
SMS::lduDiagPrecond	54
SMS::lduDICPrecond	57
SMS::lduDILUPrecond	61
SMS::MGPrecond	90
SMS::PtrList< T >	115
SMS::PtrList< SMS::lduMatrix >	115
SMS::PtrList< SMS::matrix::smoother >	115
SMS::PtrList< SMS::patch >	115
SMS::PtrList< SMS::Vector >	115
SMS::matrix::smoother	119
SMS::chebySmoother	30
SMS::lduGaussSeidelSmoother	65
SMS::smsMPI	120
SMS::matrix::solver	123
SMS::MGSolver	93
SMS::PBiCGStab	107
SMS::PCG	111
SMS::matrix::solverPerformance	128
Timer	133
SMS::Vector< T >	137
SMS::Vector< label >	137
SMS::Vector< scalar >	137

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

SMS::matrix::agglomeration	25
SMS::chebySmoothen	30
SMS::eigenDiagPCG	33
SMS::interfaces	37
SMS::labelPair	42
SMS::lduAgglomeration	46
SMS::lduDiagPrecond	54
SMS::lduDICPrecond	57
SMS::lduDILUPrecond	61
SMS::lduGaussSeidelSmoothen	65
SMS::lduMatrix	67
SMS::matrix	87
SMS::MGPrecond	90
SMS::MGSolver	93
SMS::patch	100
SMS::PBiCGStab	107
SMS::PCG	111
SMS::matrix::preconditioner	114
SMS::PtrList< T >	115
SMS::matrix::smoother	119
SMS::smsMPI	120
SMS::matrix::solver	123
SMS::matrix::solverPerformance	128
Timer	133
SMS::Vector< T >	137

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

SunwayMatrixSolver/src/base/ sms.cpp	151
SunwayMatrixSolver/src/base/ sms.hpp	
Brief information to be added	152
SunwayMatrixSolver/src/base/mpi/ smsMPI.cpp	147
SunwayMatrixSolver/src/base/mpi/ smsMPI.hpp	
Brief information to be added	147
SunwayMatrixSolver/src/base/PtrList/ PtrList.hpp	
Brief information to be added	150
SunwayMatrixSolver/src/base/vectors/ smsVector.hpp	
Brief information to be added	156
SunwayMatrixSolver/src/base/vectors/ smsVectorOperations.cpp	157
SunwayMatrixSolver/src/matrix/ smsMatrix.hpp	176
SunwayMatrixSolver/src/matrix/ smsMatrixAgglomeration.cpp	178
SunwayMatrixSolver/src/matrix/ smsMatrixPrecond.cpp	178
SunwayMatrixSolver/src/matrix/ smsMatrixSolver.cpp	179
SunwayMatrixSolver/src/matrix/ smsMatrixTest.cpp	180
SunwayMatrixSolver/src/matrix/csrMatrix/ csrMatrix.cpp	158
SunwayMatrixSolver/src/matrix/csrMatrix/ csrMatrix.hpp	158
SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/ csrAgglomeration.cpp	158
SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/ csrAgglomeration.hpp	158
SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/ eigenDiagPCG.cpp	159
SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/ eigenDiagPCG.hpp	159
SunwayMatrixSolver/src/matrix/interfaces/ interfaces.cpp	161
SunwayMatrixSolver/src/matrix/interfaces/ interfaces.hpp	162
SunwayMatrixSolver/src/matrix/interfaces/labelPair/ labelPair.cpp	164
SunwayMatrixSolver/src/matrix/interfaces/labelPair/ labelPair.hpp	164
SunwayMatrixSolver/src/matrix/interfaces/patch/ patch.cpp	165
SunwayMatrixSolver/src/matrix/interfaces/patch/ patch.hpp	166
SunwayMatrixSolver/src/matrix/lduMatrix/ lduMatrix.cpp	173
SunwayMatrixSolver/src/matrix/lduMatrix/ lduMatrix.hpp	174
SunwayMatrixSolver/src/matrix/lduMatrix/ lduMatrixSpMV.cpp	176
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/ lduAgglomerate.cpp	167
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/ lduAgglomerateMatrix.cpp	168
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/ lduAgglomeration.cpp	169
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/ lduAgglomeration.hpp	170

SunwayMatrixSolver/src/matrix/lDuMatrix/lDuAgglomeration/ lDuAgglomerationAddressing.cpp	172
SunwayMatrixSolver/src/matrix/lDuMatrix/lDuAgglomeration/ lDuAgglomerationCombineLevels.cpp	172
SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDiagonal/ csrDiagPrecond.cpp	181
SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDiagonal/ csrDiagPrecond.hpp	181
SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDIC/ csrDICPrecond.cpp	181
SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDIC/ csrDICPrecond.hpp	181
SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDILU/ csrDILUPrecond.cpp	182
SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDILU/ csrDILUPrecond.hpp	182
SunwayMatrixSolver/src/preconditioners/lDuPreconditioners/lDuDiagonal/ lDuDiagPrecond.cpp	182
SunwayMatrixSolver/src/preconditioners/lDuPreconditioners/lDuDiagonal/ lDuDiagPrecond.hpp	183
SunwayMatrixSolver/src/preconditioners/lDuPreconditioners/lDuDIC/ lDuDICPrecond.cpp	185
SunwayMatrixSolver/src/preconditioners/lDuPreconditioners/lDuDIC/ lDuDICPrecond.hpp	186
SunwayMatrixSolver/src/preconditioners/lDuPreconditioners/lDuDILU/ lDuDILUPrecond.cpp	187
SunwayMatrixSolver/src/preconditioners/lDuPreconditioners/lDuDILU/ lDuDILUPrecond.hpp	188
SunwayMatrixSolver/src/preconditioners/MGPreconditioner/ MGPreconditioner.cpp	190
SunwayMatrixSolver/src/preconditioners/MGPreconditioner/ MGPreconditioner.hpp	190
SunwayMatrixSolver/src/smoothers/chebySmoothers/ chebySmoothers.cpp	192
SunwayMatrixSolver/src/smoothers/chebySmoothers/ chebySmoothers.hpp	193
SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmoothers/ csrGaussSeidelSmoothers.cpp	195
SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmoothers/ csrGaussSeidelSmoothers.hpp	195
SunwayMatrixSolver/src/smoothers/lDuSmoothers/lDuGaussSeidelSmoothers/ lDuGaussSeidelSmoothers.cpp	195
SunwayMatrixSolver/src/smoothers/lDuSmoothers/lDuGaussSeidelSmoothers/ lDuGaussSeidelSmoothers.hpp	196
SunwayMatrixSolver/src/solvers/MG/ MG.cpp	198
SunwayMatrixSolver/src/solvers/MG/ MG.hpp	199
SunwayMatrixSolver/src/solvers/MG/ MGsolve.cpp	200
SunwayMatrixSolver/src/solvers/MG/ scalingFactor.cpp	201
SunwayMatrixSolver/src/solvers/MG/ solveCoarsestLevel.cpp	201
SunwayMatrixSolver/src/solvers/MG/ Vcycle.cpp	202
SunwayMatrixSolver/src/solvers/PBiCGStab/ PBiCGStab.cpp	203
SunwayMatrixSolver/src/solvers/PBiCGStab/ PBiCGStab.hpp	204
SunwayMatrixSolver/src/solvers/PCG/ PCG.cpp	206
SunwayMatrixSolver/src/solvers/PCG/ PCG.hpp	207
SunwayMatrixSolver/tools/compass/ compassInterface.cpp	208
SunwayMatrixSolver/tools/compass/ compassInterface.hpp	209
SunwayMatrixSolver/tools/fortranInterface/ fortranInterface.cpp	210
SunwayMatrixSolver/tools/fortranInterface/ fortranInterface.h	216
SunwayMatrixSolver/tools/matrixConversion/ coo2csr.cpp	219
SunwayMatrixSolver/tools/matrixConversion/ coo2lDu.cpp	219
SunwayMatrixSolver/tools/matrixConversion/ csr2coo.cpp	220
SunwayMatrixSolver/tools/matrixConversion/ matrixConversion.cpp	220
SunwayMatrixSolver/tools/matrixConversion/ matrixConversion.hpp	221
SunwayMatrixSolver/tools/printData/printSMS/ printSMS.cpp	223
SunwayMatrixSolver/tools/printData/printSMS/ printSMS.hpp	224
SunwayMatrixSolver/tools/readData/readFromOpenFOAM/ readFromOpenFOAM.cpp	226
SunwayMatrixSolver/tools/readData/readFromOpenFOAM/ readFromOpenFOAM.hpp	227
SunwayMatrixSolver/tools/timers/ Timers.cpp	229
SunwayMatrixSolver/tools/timers/ Timers.hpp	230

Chapter 5

Namespace Documentation

5.1 SMS Namespace Reference

Classes

- class [chebySmoother](#)
- class [eigenDiagPCG](#)
- class [interfaces](#)
- class [labelPair](#)
- class [lDUAgglomeration](#)
- class [lDUDiagPrecond](#)
- class [lDUDICPrecond](#)
- class [lDUDILUPrecond](#)
- class [lDUGaussSeidelSmoother](#)
- class [lDUMatrix](#)
- class [matrix](#)
- class [MGPrecond](#)
- class [MGSolver](#)
- class [patch](#)
- class [PBiCGStab](#)
- class [PCG](#)
- class [PtrList](#)
- class [smsMPI](#)
- class [Vector](#)

TypeDefs

- typedef [Vector< label > labelField](#)
- typedef [Vector< scalar > scalarField](#)

Functions

- template<typename T >
void **reduce** (T &v)
- unsigned int **sleep** (const unsigned int)
- template<typename T >
T **max** (T a, T b)
- template<typename T >
T **min** (T a, T b)
- template<typename T >
T **mag** (T v)
- scalar **dot** (const scalarField &v1, const scalarField &v2)
- void **contract_sw_matrix_** (long int *APtrPtr, const label *nCellsPtr, const label *rowsPtr, const label *colsPtr, const label *sizePtr)
- void **contract_sw_matrix_interfaces_** (long int *APtrPtr, const label *nNeiProcsPtr, const label *destRankPtr, const label *offDiagRowsPtr, const label *offDiagStartsPtr)
- void **fill_sw_matrix_coefficients_** (long int *APtrPtr, const scalar *diagPtr, const scalar *upperPtr, const scalar *lowerPtr)
- void **fill_sw_matrix_interfaces_coefficients_** (long int *APtrPtr, const label *offDiagStartsPtr, const scalar *offDiagCoeffs)
- void **contract_solver_mg_** (long int *mgPtrPtr, long int *APtrPtr, const scalar *faceAreaPtr, const label *smTypePtr, const label *maxLevelsPtr, const label *nCellsCoarsestPtr)
- void **sw_solver_mg_set_maxiter_** (long int *solverPtrPtr, const label *maxIterPtr)
- void **sw_solver_mg_set_miniter_** (long int *solverPtrPtr, const label *minIterPtr)
- void **sw_solver_mg_set_tol_** (long int *solverPtrPtr, const scalar *tolPtr)
- void **sw_solver_mg_set_reltol_** (long int *solverPtrPtr, const scalar *reltolPtr)
- void **sw_solver_mg_set_npresweeps_** (long int *solverPtrPtr, const label *numPtr)
- void **sw_solver_mg_set_npostsweeps_** (long int *solverPtrPtr, const label *numPtr)
- void **sw_solver_mg_set_nfinestsweeps_** (long int *solverPtrPtr, const label *numPtr)
- void **sw_solve_mg_** (long int *mgPtrPtr, long int *APtrPtr, scalar *xPtr, scalar *bPtr)
- void **contract_solver_pbicgstab_** (long int *solverPtrPtr)
- void **sw_solver_pbicgstab_set_maxiter_** (long int *solverPtrPtr, const label *maxIterPtr)
- void **sw_solver_pbicgstab_set_miniter_** (long int *solverPtrPtr, const label *minIterPtr)
- void **sw_solver_pbicgstab_set_tol_** (long int *solverPtrPtr, const scalar *tolPtr)
- void **sw_solver_pbicgstab_set_reltol_** (long int *solverPtrPtr, const scalar *reltolPtr)
- void **sw_solver_pbicgstab_set_precond_** (long int *solverPtrPtr, long int *APtrPtr, const label *precondTypePtr)
- void **sw_solve_pbicgstab_** (long int *solverPtrPtr, long int *APtrPtr, scalar *xPtr, scalar *bPtr)
- void **sw_matrix_destroy_** (long int *APtrPtr)
- void **sw_solver_destroy_mg_** (long int *solverPtrPtr)
- void **sw_solver_destroy_pbicgstab_** (long int *solverPtrPtr)
- void **sortData** (scalarField &data, const labelField &order, const labelField &cellFaces)
- **lduMatrix & coo2ldu** (const scalar *dataPtr, const label *rowsPtr, const label *columnPtr, const label nCells, const label size, const bool symm)
- **lduMatrix & csr2ldu** (const scalar *dataPtr, const label *compRowsPtr, const label *columnPtr, const label nCells, const label size, const bool symm)
- void **printLDUMatrix** (const **lduMatrix** &A, const char *name)
- template<typename T >
void **printVector** (const T &b, const char *name)
- void **printInterfaces** (const **lduMatrix** &A, const char *name)
- void **constructMatrixFromOpenFOAM** (**lduMatrix** &lduA, const char *fileName)
- void **constructVectorFromOpenFOAM** (scalarField &b, const char *fileName)
- void **constructInterfacesFromOpenFOAM** (**lduMatrix** &lduA, const char *fileName)

5.1.1 Typedef Documentation

5.1.1.1 labelField

```
typedef Vector<label> SMS::labelField
```

5.1.1.2 scalarField

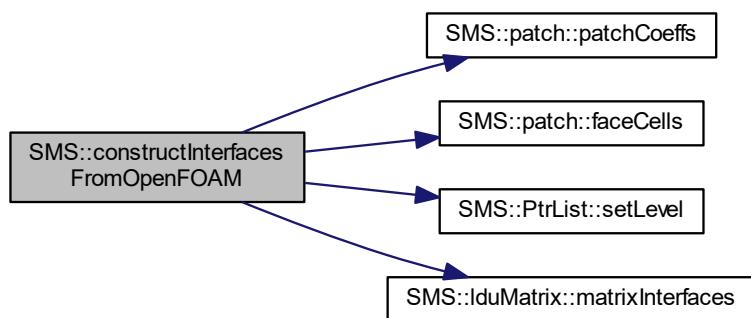
```
typedef Vector<scalar> SMS::scalarField
```

5.1.2 Function Documentation

5.1.2.1 constructInterfacesFromOpenFOAM()

```
void SMS::constructInterfacesFromOpenFOAM (
    lduMatrix & lduA,
    const char * fileName )
```

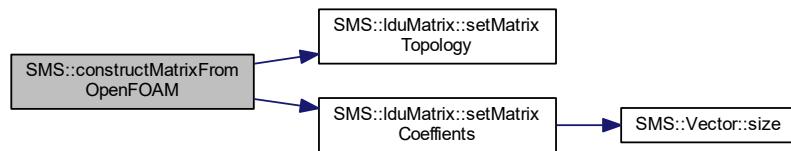
Here is the call graph for this function:



5.1.2.2 constructMatrixFromOpenFOAM()

```
void SMS::constructMatrixFromOpenFOAM (
    lduMatrix & lduA,
    const char * fileName )
```

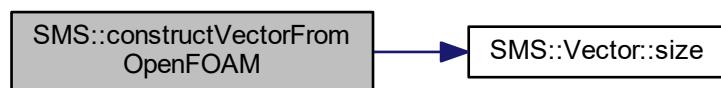
Here is the call graph for this function:



5.1.2.3 constructVectorFromOpenFOAM()

```
void SMS::constructVectorFromOpenFOAM (
    scalarField & b,
    const char * fileName )
```

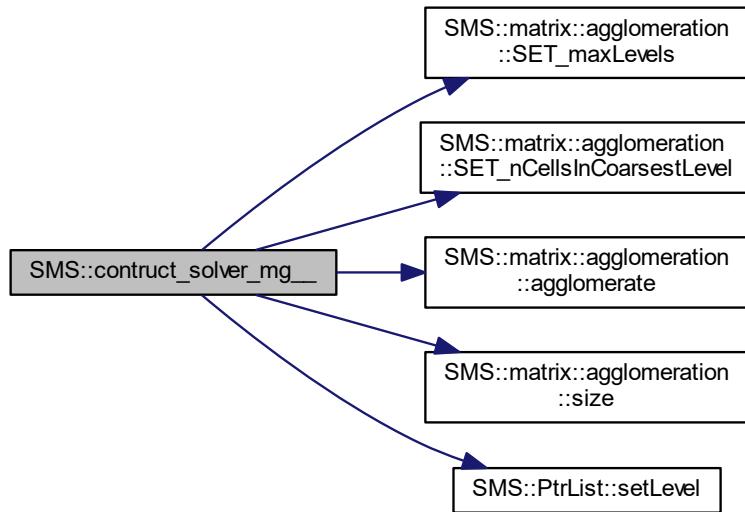
Here is the call graph for this function:



5.1.2.4 construct_solver_mg_()

```
void SMS::construct_solver_mg_ (
    long int * mgPtrPtr,
    long int * APtrPtr,
    const scalar * faceAreaPtr,
    const label * smTypePtr,
    const label * maxLeveisPtr,
    const label * nCellsCoarsestPtr )
```

Here is the call graph for this function:



5.1.2.5 `construct_solver_pbicgstab_()`

```
void SMS::construct_solver_pbicgstab_ (
    long int * solverPtrPtr )
```

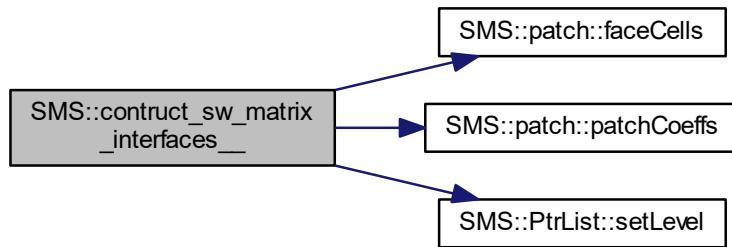
5.1.2.6 `construct_sw_matrix_()`

```
void SMS::construct_sw_matrix_ (
    long int * APtrPtr,
    const label * nCellsPtr,
    const label * rowsPtr,
    const label * colsPtr,
    const label * sizePtr )
```

5.1.2.7 `construct_sw_matrix_interfaces_()`

```
void SMS::construct_sw_matrix_interfaces_ (
    long int * APtrPtr,
    const label * nNeiProcsPtr,
    const label * destRankPtr,
    const label * offDiagRowsPtr,
    const label * offDiagStartsPtr )
```

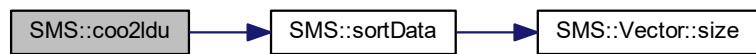
Here is the call graph for this function:



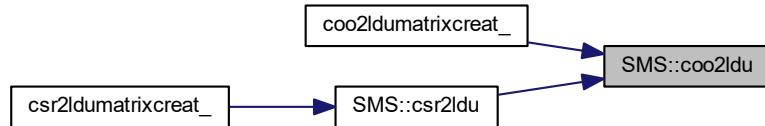
5.1.2.8 `coo2ldu()`

```
SMS::lduMatrix & SMS::coo2ldu (
    const scalar * dataPtr,
    const label * rowsPtr,
    const label * columnPtr,
    const label nCells,
    const label size,
    const bool symm )
```

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.9 csr2ldu()

```
SMS::lduMatrix & SMS::csr2ldu (
    const scalar * dataPtr,
    const label * compRowsPtr,
    const label * columnPtr,
    const label nCells,
    const label size,
    const bool symm )
```

Here is the call graph for this function:



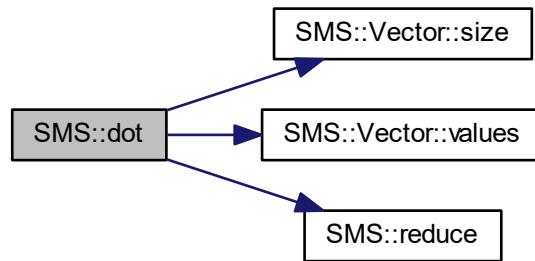
Here is the caller graph for this function:



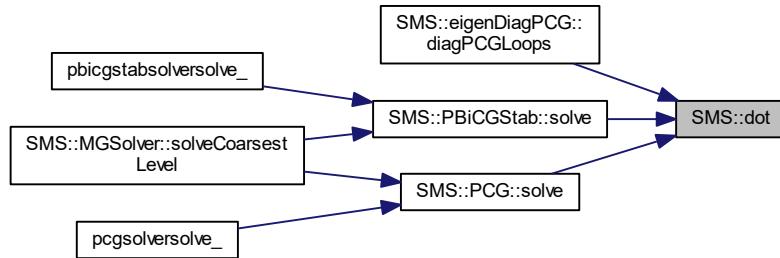
5.1.2.10 dot()

```
scalar SMS::dot (
    const scalarField & v1,
    const scalarField & v2 )
```

Here is the call graph for this function:



Here is the caller graph for this function:



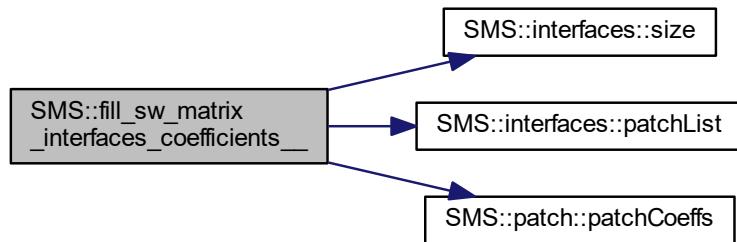
5.1.2.11 fill_sw_matrix_coefficients__()

```
void SMS::fill_sw_matrix_coefficients__ (
    long int * APtrPtr,
    const scalar * diagPtr,
    const scalar * upperPtr,
    const scalar * lowerPtr )
```

5.1.2.12 `fill_sw_matrix_interfaces_coefficients_()`

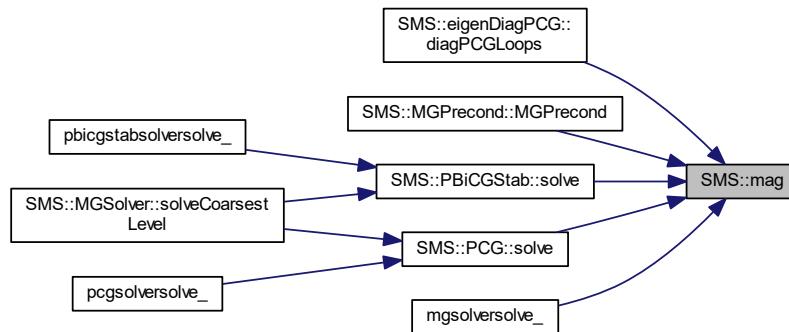
```
void SMS::fill_sw_matrix_interfaces_coefficients_ (
    long int * APtrPtr,
    const label * offDiagStartsPtr,
    const scalar * offDiagCoeffs )
```

Here is the call graph for this function:

5.1.2.13 `mag()`

```
template<typename T >
T SMS::mag (
    T v )
```

Here is the caller graph for this function:



5.1.2.14 max()

```
template<typename T >
T SMS::max (
    T a,
    T b )
```

Here is the caller graph for this function:



5.1.2.15 min()

```
template<typename T >
T SMS::min (
    T a,
    T b )
```

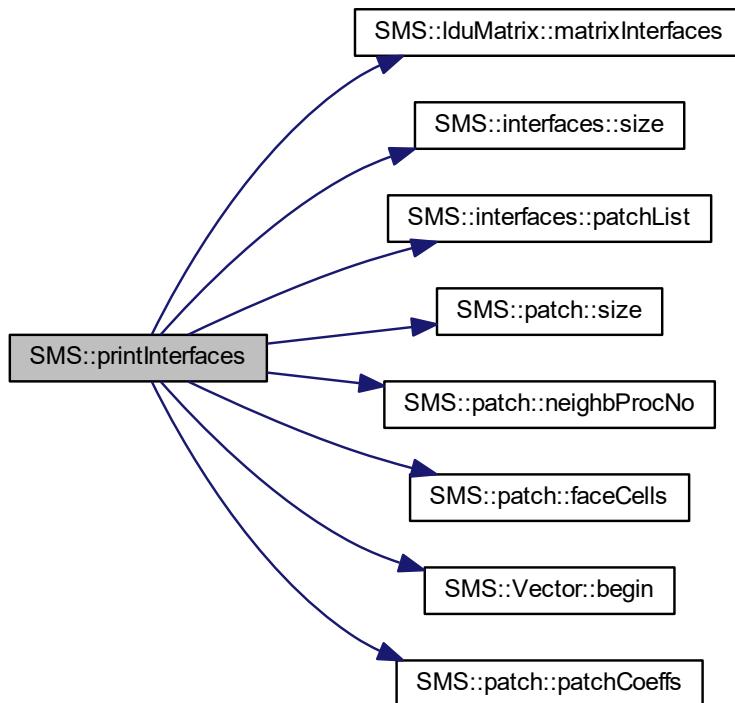
Here is the caller graph for this function:



5.1.2.16 printInterfaces()

```
void SMS::printInterfaces (
    const lduMatrix & A,
    const char * name )
```

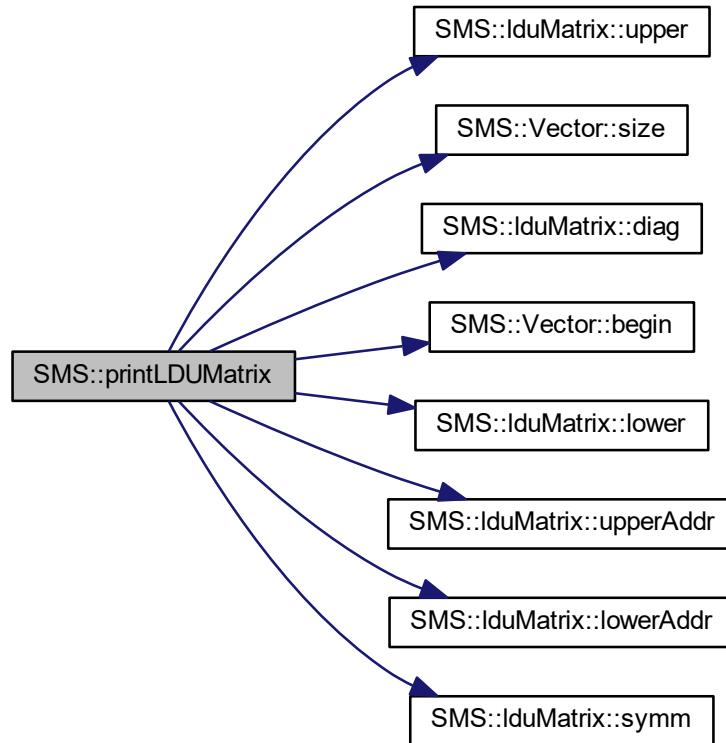
Here is the call graph for this function:



5.1.2.17 printLDUMatrix()

```
void SMS::printLDUMatrix (
    const lduMatrix & A,
    const char * name )
```

Here is the call graph for this function:



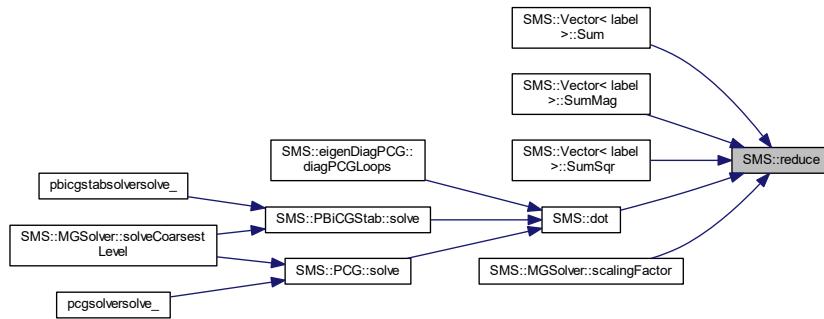
5.1.2.18 printVector()

```
template<typename T >
void SMS::printVector (
    const T & b,
    const char * name )
```

5.1.2.19 reduce()

```
template<typename T >
void SMS::reduce (
    T & v )
```

Here is the caller graph for this function:



5.1.2.20 `sleep()`

```
unsigned int SMS::sleep (
    const unsigned int )
```

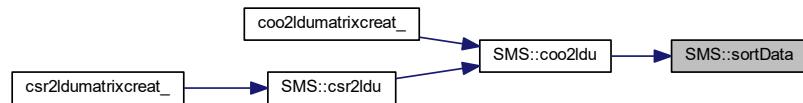
5.1.2.21 `sortData()`

```
void SMS::sortData (
    scalarField & data,
    const labelField & order,
    const labelField & cellFaces )
```

Here is the call graph for this function:



Here is the caller graph for this function:



5.1.2.22 `sw_matrix_destroy_()`

```
void SMS::sw_matrix_destroy_ (
    long int * APtrPtr )
```

5.1.2.23 `sw_solve_mg_()`

```
void SMS::sw_solve_mg_ (
    long int * mgPtrPtr,
    long int * APtrPtr,
    scalar * xPtr,
    scalar * bPtr )
```

5.1.2.24 `sw_solve_pbicgstab_()`

```
void SMS::sw_solve_pbicgstab_ (
    long int * solverPtrPtr,
    long int * APtrPtr,
    scalar * xPtr,
    scalar * bPtr )
```

5.1.2.25 `sw_solver_destroy_mg_()`

```
void SMS::sw_solver_destroy_mg_ (
    long int * solverPtrPtr )
```

5.1.2.26 `sw_solver_destroy_pbicgstab_()`

```
void SMS::sw_solver_destroy_pbicgstab_ (
    long int * solverPtrPtr )
```

5.1.2.27 `sw_solver_mg_set_maxiter_()`

```
void SMS::sw_solver_mg_set_maxiter_ (
    long int * solverPtrPtr,
    const label * maxIterPtr )
```

5.1.2.28 `sw_solver_mg_set_miniter_()`

```
void SMS::sw_solver_mg_set_miniter__ (
    long int * solverPtrPtr,
    const label * minIterPtr )
```

5.1.2.29 `sw_solver_mg_set_nfinestsweeps_()`

```
void SMS::sw_solver_mg_set_nfinestsweeps__ (
    long int * solverPtrPtr,
    const label * numPtr )
```

5.1.2.30 `sw_solver_mg_set_npostsweeps_()`

```
void SMS::sw_solver_mg_set_npostsweeps__ (
    long int * solverPtrPtr,
    const label * numPtr )
```

5.1.2.31 `sw_solver_mg_set_npresweeps_()`

```
void SMS::sw_solver_mg_set_npresweeps__ (
    long int * solverPtrPtr,
    const label * numPtr )
```

5.1.2.32 `sw_solver_mg_set_reltol_()`

```
void SMS::sw_solver_mg_set_reltol__ (
    long int * solverPtrPtr,
    const scalar * reltolPtr )
```

5.1.2.33 `sw_solver_mg_set_tol_()`

```
void SMS::sw_solver_mg_set_tol__ (
    long int * solverPtrPtr,
    const scalar * tolPtr )
```

5.1.2.34 `sw_solver_pbicgstab_set_maxiter_()`

```
void SMS::sw_solver_pbicgstab_set_maxiter_ (
    long int * solverPtrPtr,
    const label * maxIterPtr )
```

5.1.2.35 `sw_solver_pbicgstab_set_miniter_()`

```
void SMS::sw_solver_pbicgstab_set_miniter_ (
    long int * solverPtrPtr,
    const label * minIterPtr )
```

5.1.2.36 `sw_solver_pbicgstab_set_precond_()`

```
void SMS::sw_solver_pbicgstab_set_precond_ (
    long int * solverPtrPtr,
    long int * APtrPtr,
    const label * precondTypePtr )
```

5.1.2.37 `sw_solver_pbicgstab_set_reltol_()`

```
void SMS::sw_solver_pbicgstab_set_reltol_ (
    long int * solverPtrPtr,
    const scalar * reltolPtr )
```

5.1.2.38 `sw_solver_pbicgstab_set_tol_()`

```
void SMS::sw_solver_pbicgstab_set_tol_ (
    long int * solverPtrPtr,
    const scalar * tolPtr )
```

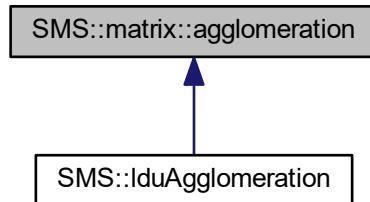
Chapter 6

Class Documentation

6.1 SMS::matrix::agglomeration Class Reference

```
#include <smsMatrix.hpp>
```

Inheritance diagram for SMS::matrix::agglomeration:



Public Member Functions

- `agglomeration ()`
- `virtual ~agglomeration ()`
- `virtual label size () const =0`
- `virtual const matrix & coarseMatrixLevels (const label leveli) const =0`
- `virtual const labelField & restrictAddressing (const label leveli) const =0`
- `virtual const labelField & faceRestrictAddressing (const label leveli) const =0`
- `virtual void agglomerateMatrix (const label fineLevelIndex)=0`
- `virtual void agglomerate (const scalarField &weights)=0`
- `virtual void SET_nCellsInCoarsestLevel (const label i)=0`
- `virtual void SET_maxLevels (const label i)=0`
- `template<typename T >`
 `void restrictField (Vector< T > &cf, const Vector< T > &ff, const label fineLevelIndex) const`
- `template<typename T >`
 `void prolongField (Vector< T > &ff, const Vector< T > &cf, const label coarseLevelIndex) const`
- `template<typename T >`
 `void restrictFaceField (Vector< T > &cf, const Vector< T > &ff, const label fineLevelIndex) const`

6.1.1 Constructor & Destructor Documentation

6.1.1.1 agglomeration()

```
SMS::matrix::agglomeration::agglomeration ( )
```

6.1.1.2 ~agglomeration()

```
virtual SMS::matrix::agglomeration::~agglomeration ( ) [inline], [virtual]
```

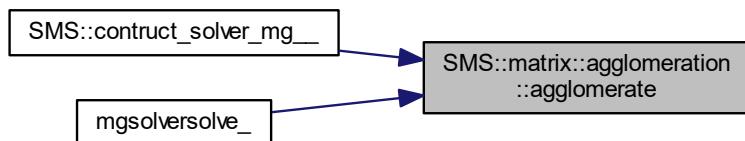
6.1.2 Member Function Documentation

6.1.2.1 agglomerate()

```
virtual void SMS::matrix::agglomeration::agglomerate (
    const scalarField & weights ) [pure virtual]
```

Implemented in [SMS::lduAgglomeration](#).

Here is the caller graph for this function:



6.1.2.2 agglomerateMatrix()

```
virtual void SMS::matrix::agglomeration::agglomerateMatrix (
    const label fineLevelIndex ) [pure virtual]
```

Implemented in [SMS::lduAgglomeration](#).

6.1.2.3 coarseMatrixLevels()

```
virtual const matrix& SMS::matrix::agglomeration::coarseMatrixLevels (
    const label leveli ) const [pure virtual]
```

Implemented in [SMS::lDUAgglomeration](#).

6.1.2.4 faceRestrictAddressing()

```
virtual const labelField& SMS::matrix::agglomeration::faceRestrictAddressing (
    const label leveli ) const [pure virtual]
```

Implemented in [SMS::lDUAgglomeration](#).

6.1.2.5 prolongField()

```
template<typename T >
void SMS::matrix::agglomeration::prolongField (
    Vector< T > & ff,
    const Vector< T > & cf,
    const label coarseLevelIndex ) const
```

Here is the call graph for this function:



6.1.2.6 restrictAddressing()

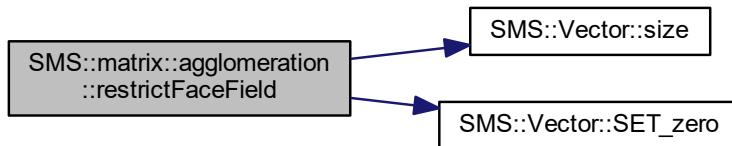
```
virtual const labelField& SMS::matrix::agglomeration::restrictAddressing (
    const label leveli ) const [pure virtual]
```

Implemented in [SMS::lDUAgglomeration](#).

6.1.2.7 restrictFaceField()

```
template<typename T >
void SMS::matrix::agglomeration::restrictFaceField (
    Vector< T > & cf,
    const Vector< T > & ff,
    const label fineLevelIndex ) const
```

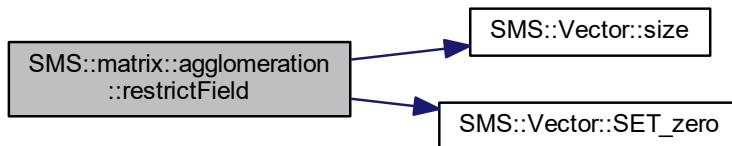
Here is the call graph for this function:



6.1.2.8 restrictField()

```
template<typename T >
void SMS::matrix::agglomeration::restrictField (
    Vector< T > & cf,
    const Vector< T > & ff,
    const label fineLevelIndex ) const
```

Here is the call graph for this function:

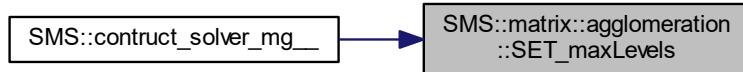


6.1.2.9 SET_maxLevels()

```
virtual void SMS::matrix::agglomeration::SET_maxLevels (
    const label i ) [pure virtual]
```

Implemented in [SMS::lduAgglomeration](#).

Here is the caller graph for this function:

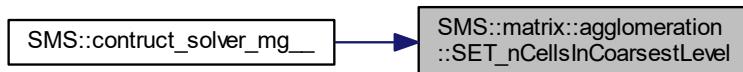


6.1.2.10 SET_nCellsInCoarsestLevel()

```
virtual void SMS::matrix::agglomeration::SET_nCellsInCoarsestLevel (
    const label i ) [pure virtual]
```

Implemented in [SMS::lduAgglomeration](#).

Here is the caller graph for this function:

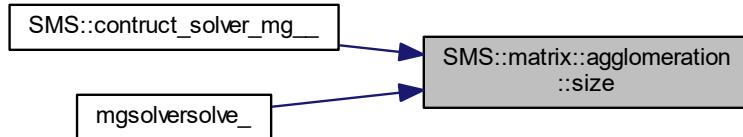


6.1.2.11 size()

```
virtual label SMS::matrix::agglomeration::size () const [pure virtual]
```

Implemented in [SMS::lduAgglomeration](#).

Here is the caller graph for this function:



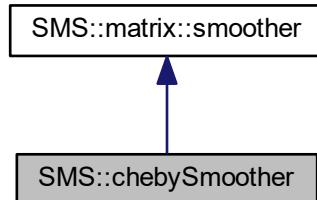
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/[smsMatrix.hpp](#)
- SunwayMatrixSolver/src/matrix/[smsMatrixAgglomeration.cpp](#)

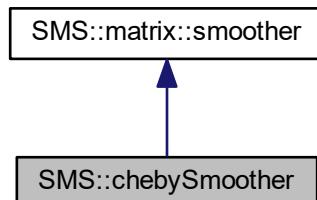
6.2 SMS::chebySmoothen Class Reference

```
#include <chebySmoothen.hpp>
```

Inheritance diagram for SMS::chebySmoothen:



Collaboration diagram for SMS::chebySmoothen:



Public Member Functions

- `chebySmoothen ()`
- `virtual void smooth (scalarField &x, const matrix &A, const scalarField &b, const label nSweeps) const`
- `void smooth (scalarField &x, const lduMatrix &A, const scalarField &b, const label nSweeps) const`
- `virtual void init () const`

Private Attributes

- label `nDiagPCGs_`
- scalar `maxEigPCG_`
- bool `eigFirstTimeComputed_`
- scalar `eigRatioCheby_`
- scalar `boostFactorCheby_`
- scalar `eigRatioCoarest_`
- label `preSmoothUsing_`

6.2.1 Constructor & Destructor Documentation

6.2.1.1 `chebySmoothen()`

```
SMS::chebySmoothen::chebySmoothen ( ) [inline]
```

6.2.2 Member Function Documentation

6.2.2.1 `init()`

```
virtual void SMS::chebySmoothen::init ( ) const [inline], [virtual]
```

Implements [SMS::matrix::smoother](#).

6.2.2.2 `smooth()` [1/2]

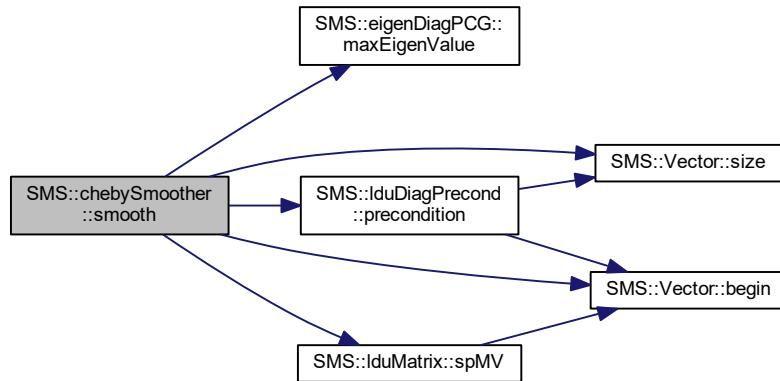
```
void SMS::chebySmoothen::smooth (
    scalarField & x,
    const matrix & A,
    const scalarField & b,
    const label nSweeps ) const [virtual]
```

Implements [SMS::matrix::smoother](#).

6.2.2.3 smooth() [2/2]

```
void SMS::chebySmoothen::smooth (
    scalarField & x,
    const lduMatrix & A,
    const scalarField & b,
    const label nSweeps ) const
```

Here is the call graph for this function:



6.2.3 Member Data Documentation

6.2.3.1 boostFactorCheby_

```
scalar SMS::chebySmoothen::boostFactorCheby_ [private]
```

6.2.3.2 eigFirstTimeComputed_

```
bool SMS::chebySmoothen::eigFirstTimeComputed_ [mutable], [private]
```

6.2.3.3 eigRatioCheby_

```
scalar SMS::chebySmoothen::eigRatioCheby_ [private]
```

6.2.3.4 eigRatioCoarest_

```
scalar SMS::chebySmoothen::eigRatioCoarest_ [private]
```

6.2.3.5 maxEigPCG_

```
scalar SMS::chebySmoothen::maxEigPCG_ [mutable], [private]
```

6.2.3.6 nDiagPCGs_

```
label SMS::chebySmoothen::nDiagPCGs_ [private]
```

6.2.3.7 preSmoothUsing_

```
label SMS::chebySmoothen::preSmoothUsing_ [private]
```

The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/smoothers/chebySmoothen/[chebySmoothen.hpp](#)
- SunwayMatrixSolver/src/smoothers/chebySmoothen/[chebySmoothen.cpp](#)

6.3 SMS::eigenDiagPCG Class Reference

```
#include <eigenDiagPCG.hpp>
```

Public Member Functions

- [eigenDiagPCG](#) (const [matrix](#) &A, [scalarField](#) &x, const [scalarField](#) &b, const [matrix::preconditioner](#) &precond, const [label](#) nDiagPCGs)
- scalar [maxEigenValue](#) () const

Private Member Functions

- void [h14Sturm](#) (scalar **TriMatrix, const scalar lamb, std::vector< scalar > &p, [label](#) &s, const [label](#) nPCGs) const
- void [computeValueForMatrix](#) (const [scalarField](#) &alphas, const [scalarField](#) &betas, scalar **TriMatrix, const [label](#) nPCGs) const
- void [determineEigRange](#) (scalar **TriMatrix, scalar &xBegin, scalar &xEnd, const [label](#) nPCGs) const
- void [computeMaxEig](#) (const [scalarField](#) &alphas, const [scalarField](#) &betas, const [label](#) nPCGs, const [label](#) k) const
- void [diagPCGLoops](#) (const [matrix](#) &A, [scalarField](#) &x, const [scalarField](#) &b, const [matrix::preconditioner](#) &precond, const [label](#) nDiagPCGs, [scalarField](#) &alphas, [scalarField](#) &betas) const
- template<typename T >
T ** [allocateSym2D](#) ([label](#) nPCGs) const
- template<typename T >
void [deleteSym2D](#) (T **arr, [label](#) nPCGs) const

Private Attributes

- scalar `maxEigenValue_`

6.3.1 Constructor & Destructor Documentation

6.3.1.1 `eigenDiagPCG()`

```
SMS::eigenDiagPCG::eigenDiagPCG (
    const matrix & A,
    scalarField & x,
    const scalarField & b,
    const matrix::preconditioner & precond,
    const label nDiagPCGs )
```

6.3.2 Member Function Documentation

6.3.2.1 `allocateSym2D()`

```
template<typename T >
T ** SMS::eigenDiagPCG::allocateSym2D (
    label nPCGs ) const [private]
```

6.3.2.2 `computeMaxEig()`

```
void SMS::eigenDiagPCG::computeMaxEig (
    const scalarField & alphas,
    const scalarField & betas,
    const label nPCGs,
    const label k ) const [private]
```

Parameters

<code>betas</code>	alphas from nPCGs times of <code>PCG</code>
<code>nPCGs</code>	betas from nPCGs times of <code>PCG</code>
<code>k</code>	nDiagPCGs_kth largest eigenvalue, default k=1 for the maximum eigenvalue

6.3.2.3 computeValueForMatrix()

```
void SMS::eigenDiagPCG::computeValueForMatrix (
    const scalarField & alphas,
    const scalarField & betas,
    scalar ** TriMatrix,
    const label nPCGs ) const [private]
```

Parameters

<i>betas</i>	alphas from nPCGs times of PCG
<i>TriMatrix</i>	betas from nPCGs times of PCG
<i>nPCGs</i>	tridiagonal matrix nDiagPCGs_

6.3.2.4 deleteSym2D()

```
template<typename T >
void SMS::eigenDiagPCG::deleteSym2D (
    T ** arr,
    label nPCGs ) const [private]
```

6.3.2.5 determineEigRange()

```
void SMS::eigenDiagPCG::determineEigRange (
    scalar ** TriMatrix,
    scalar & xBegin,
    scalar & xEnd,
    const label nPCGs ) const [private]
```

Parameters

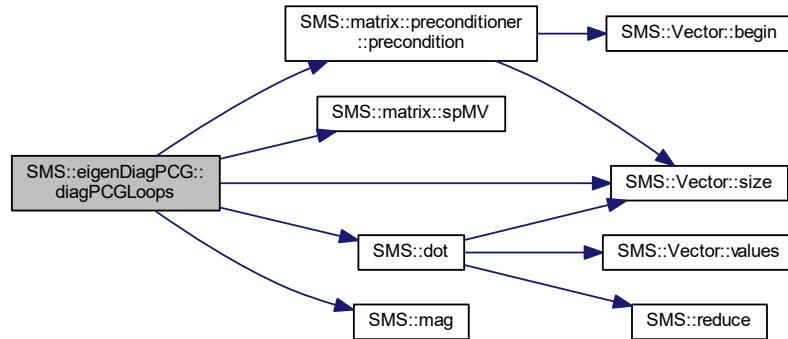
<i>xBegin</i>	tridiagonal matrix
<i>xEnd</i>	left bound of eigenvalues
<i>nPCGs</i>	right bound of eigenvalues nDiagPCGs_

6.3.2.6 diagPCGLoops()

```
void SMS::eigenDiagPCG::diagPCGLoops (
    const matrix & A,
    scalarField & x,
    const scalarField & b,
```

```
const matrix::preconditioner & precond,
const label nDiagPCGs,
scalarField & alphas,
scalarField & betas ) const [private]
```

Here is the call graph for this function:



6.3.2.7 h14Sturm()

```
void SMS::eigenDiagPCG::h14Sturm (
    scalar ** TriMatrix,
    const scalar lamb,
    std::vector< scalar > & p,
    label & s,
    const label nPCGs ) const [private]
```

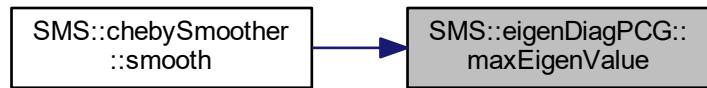
Parameters

<i>lamb</i>	tridiagonal matrix
<i>p</i>	guess eigen value (used in bisection method)
<i>s</i>	a pvector polynomials (<i>pi_1</i> to <i>pi_n</i>)
<i>nPCGs</i>	the number of sign change nDiagPCGs_

6.3.2.8 maxEigenValue()

```
scalar SMS::eigenDiagPCG::maxEigenValue ( ) const [inline]
```

Here is the caller graph for this function:



6.3.3 Member Data Documentation

6.3.3.1 maxEigenValue_

```
scalar SMS::eigenDiagPCG::maxEigenValue_ [mutable], [private]
```

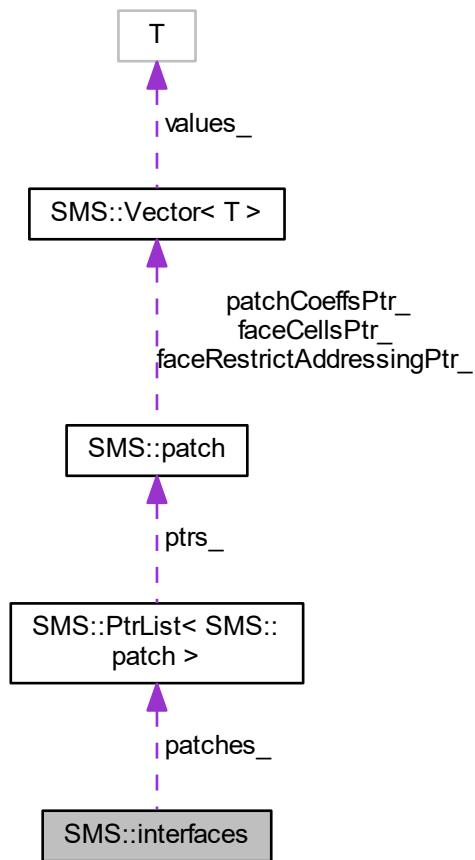
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/[eigenDiagPCG.hpp](#)
- SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/[eigenDiagPCG.cpp](#)

6.4 SMS::interfaces Class Reference

```
#include <interfaces.hpp>
```

Collaboration diagram for SMS::interfaces:



Public Member Functions

- `interfaces (const label size)`
- `interfaces (PtrList< patch > &patches)`
- `virtual ~interfaces ()`
- `virtual void initMatrixInterfaces (const scalarField &psi) const`
- `virtual void updateMatrixInterfaces (scalarField &result) const`
- `virtual patch & patchList (const label i) const`
- `PtrList< patch > & patchList ()`
- `virtual label size () const`

Private Attributes

- `PtrList< patch > & patches_`
- `scalar * sendBuffer_`
- `scalar * recvBuffer_`
- `MPI_Request * sendRecvRequests_`
- `label * locPosition_`
- `label * destRank_`

6.4.1 Constructor & Destructor Documentation

6.4.1.1 interfaces() [1/2]

```
SMS::interfaces::interfaces (
    const label size )
```

6.4.1.2 interfaces() [2/2]

```
SMS::interfaces::interfaces (
    PtrList< patch > & patches )
```

6.4.1.3 ~interfaces()

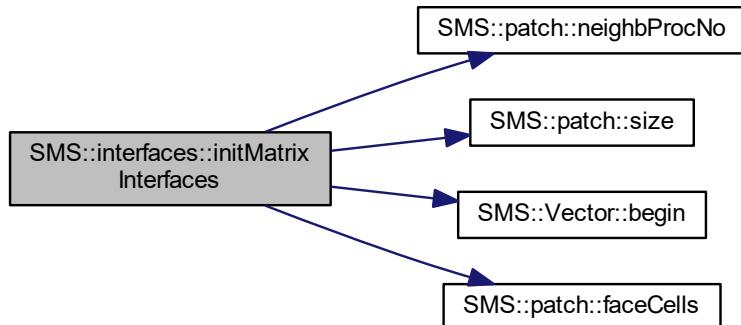
```
SMS::interfaces::~interfaces ( ) [virtual]
```

6.4.2 Member Function Documentation

6.4.2.1 initMatrixInterfaces()

```
void SMS::interfaces::initMatrixInterfaces (
    const scalarField & psi ) const [virtual]
```

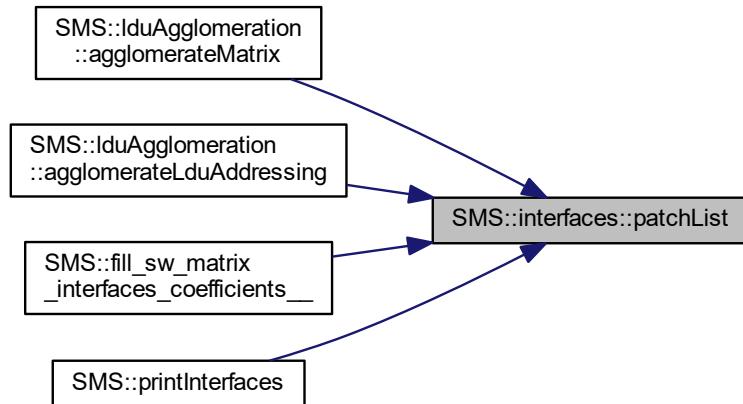
Here is the call graph for this function:



6.4.2.2 patchList() [1/2]

```
virtual patch& SMS::interfaces::patchList (
    const label i ) const [inline], [virtual]
```

Here is the caller graph for this function:



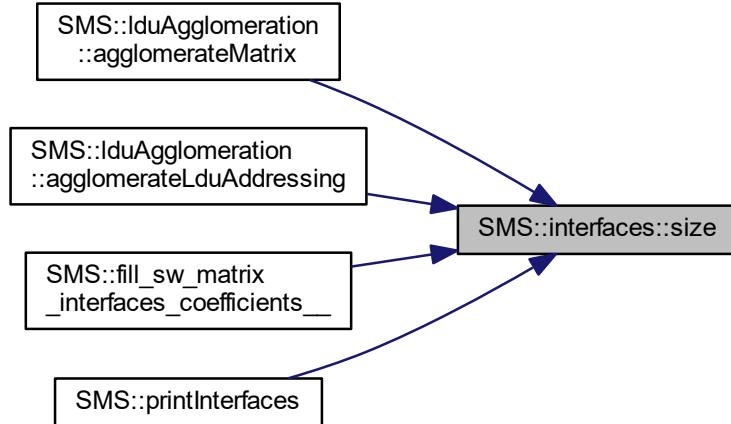
6.4.2.3 patchList() [2/2]

```
PtrList<patch>& SMS::interfaces::patchList ( ) [inline]
```

6.4.2.4 size()

```
virtual label SMS::interfaces::size ( ) const [inline], [virtual]
```

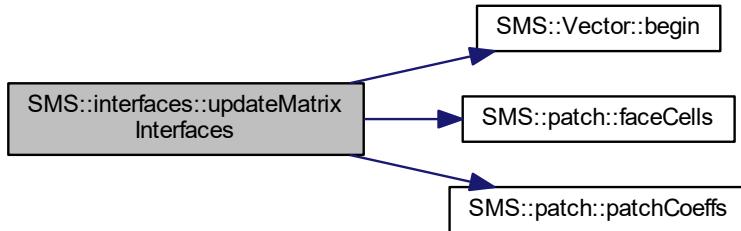
Here is the caller graph for this function:



6.4.2.5 updateMatrixInterfaces()

```
void SMS::interfaces::updateMatrixInterfaces (
    scalarField & result ) const [virtual]
```

Here is the call graph for this function:



6.4.3 Member Data Documentation

6.4.3.1 destRank_

```
label* SMS::interfaces::destRank_ [mutable], [private]
```

6.4.3.2 locPosition_

```
label* SMS::interfaces::locPosition_ [mutable], [private]
```

6.4.3.3 patches_

```
PtrList<patch>& SMS::interfaces::patches_ [private]
```

6.4.3.4 recvBuffer_

```
scalar* SMS::interfaces::recvBuffer_ [mutable], [private]
```

6.4.3.5 sendBuffer_

```
scalar* SMS::interfaces::sendBuffer_ [mutable], [private]
```

6.4.3.6 sendRecvRequests_

```
MPI_Request* SMS::interfaces::sendRecvRequests_ [mutable], [private]
```

The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/interfaces/[interfaces.hpp](#)
- SunwayMatrixSolver/src/matrix/interfaces/[interfaces.cpp](#)

6.5 SMS::labelPair Class Reference

```
#include <labelPair.hpp>
```

Public Member Functions

- `labelPair (label i, label j)`
- `label first () const`
- `label second () const`
- `label facel () const`
- `void first (const label i)`
- `void second (const label i)`
- `void facel (const label i)`
- `bool operator== (const labelPair &a) const`

Private Attributes

- `label first_`
- `label second_`
- `label facel_`

6.5.1 Constructor & Destructor Documentation

6.5.1.1 `labelPair()`

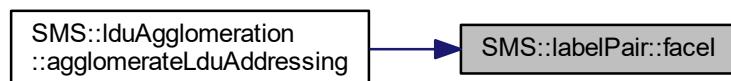
```
SMS::labelPair::labelPair (
    label i,
    label j )
```

6.5.2 Member Function Documentation

6.5.2.1 `facel()` [1/2]

```
label SMS::labelPair::faceI ( ) const [inline]
```

Here is the caller graph for this function:



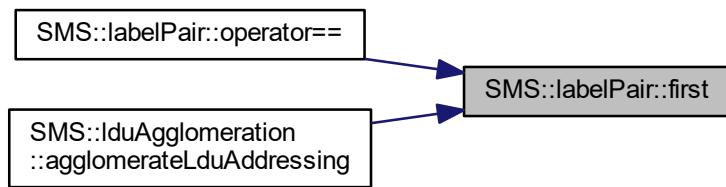
6.5.2.2 faceI() [2/2]

```
void SMS::labelPair::faceI (
    const label i )  [inline]
```

6.5.2.3 first() [1/2]

```
label SMS::labelPair::first ( ) const  [inline]
```

Here is the caller graph for this function:



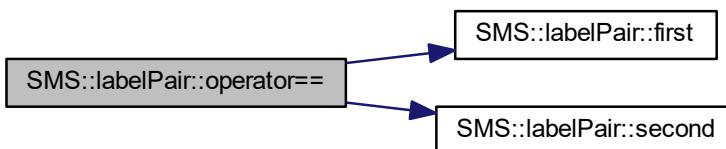
6.5.2.4 first() [2/2]

```
void SMS::labelPair::first (
    const label i )  [inline]
```

6.5.2.5 operator==()

```
bool SMS::labelPair::operator== (
    const labelPair & a ) const  [inline]
```

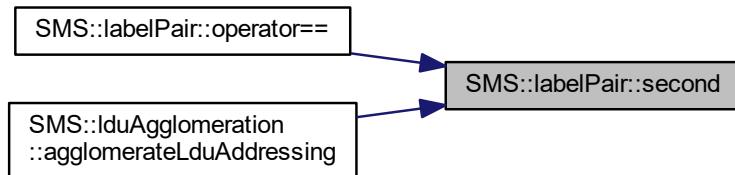
Here is the call graph for this function:



6.5.2.6 second() [1/2]

```
label SMS::labelPair::second ( ) const [inline]
```

Here is the caller graph for this function:



6.5.2.7 second() [2/2]

```
void SMS::labelPair::second ( const label i ) [inline]
```

6.5.3 Member Data Documentation

6.5.3.1 faceI_

```
label SMS::labelPair::faceI_ [private]
```

6.5.3.2 first_

```
label SMS::labelPair::first_ [private]
```

6.5.3.3 second_

```
label SMS::labelPair::second_ [private]
```

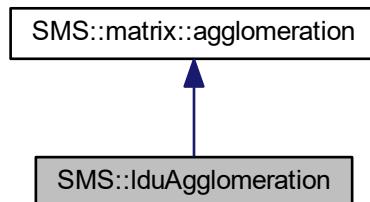
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/interfaces/labelPair/[labelPair.hpp](#)
- SunwayMatrixSolver/src/matrix/interfaces/labelPair/[labelPair.cpp](#)

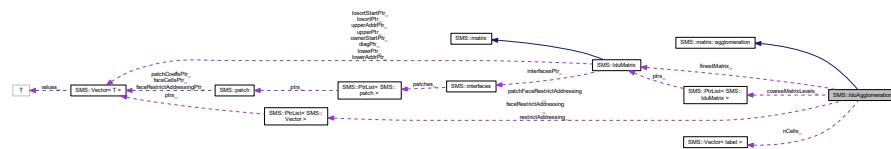
6.6 SMS::lduAgglomeration Class Reference

```
#include <lduAgglomeration.hpp>
```

Inheritance diagram for SMS::lduAgglomeration:



Collaboration diagram for SMS::lduAgglomeration:



Public Member Functions

- `lduAgglomeration (const lduMatrix &A)`
- `virtual ~lduAgglomeration ()`
- `virtual void agglomerate (const scalarField &weights)`
- `virtual label size () const`
- `virtual const lduMatrix & coarseMatrixLevels (const label leveli) const`
- `virtual const labelField & restrictAddressing (const label leveli) const`
- `virtual const labelField & faceRestrictAddressing (const label leveli) const`
- `virtual void SET_nCellsInCoarsestLevel (const label i)`
- `virtual void SET_maxLevels (const label i)`

Private Member Functions

- `void agglomerateLduAddressing (const label fineLevelIndex)`
- `void compactLevels (const label nCreatedLevels)`
- `bool continueAgglomerating (const label nCoarseCells) const`
- `void combineLevels (const label curLevel)`
- `labelField & agglomerate (label &nCoarseCells, const lduMatrix &fineA, const scalarField &weights)`
- `virtual void agglomerateMatrix (const label fineLevelIndex)`
- `const lduMatrix & matrixLevel (const label leveli) const`

Private Attributes

- label `maxLevels_`
- label `nCellsInCoarsestLevel_`
- `labelField nCells_`
- label `nCreatedLevels_`
- label `mergeLevels_`
- const `lduMatrix & finestMatrix_`
- `PtrList< lduMatrix > coarseMatrixLevels_`
- `PtrList< labelField > restrictAddressing_`
- `PtrList< labelField > faceRestrictAddressing_`
- `PtrList< labelField > patchFaceRestrictAddressing_`

Static Private Attributes

- static bool `forward_`

6.6.1 Constructor & Destructor Documentation

6.6.1.1 `lduAgglomeration()`

```
SMS::lduAgglomeration::lduAgglomeration (
    const lduMatrix & A )
```

6.6.1.2 `~lduAgglomeration()`

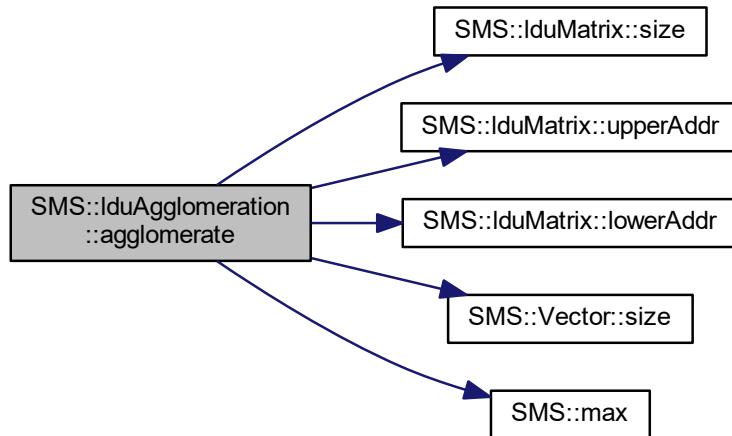
```
SMS::lduAgglomeration::~lduAgglomeration ( ) [virtual]
```

6.6.2 Member Function Documentation

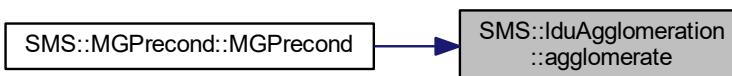
6.6.2.1 `agglomerate()` [1/2]

```
SMS::labelField & SMS::lduAgglomeration::agglomerate (
    label & nCoarseCells,
    const lduMatrix & fineA,
    const scalarField & weights ) [private]
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.6.2.2 `agglomerate()` [2/2]

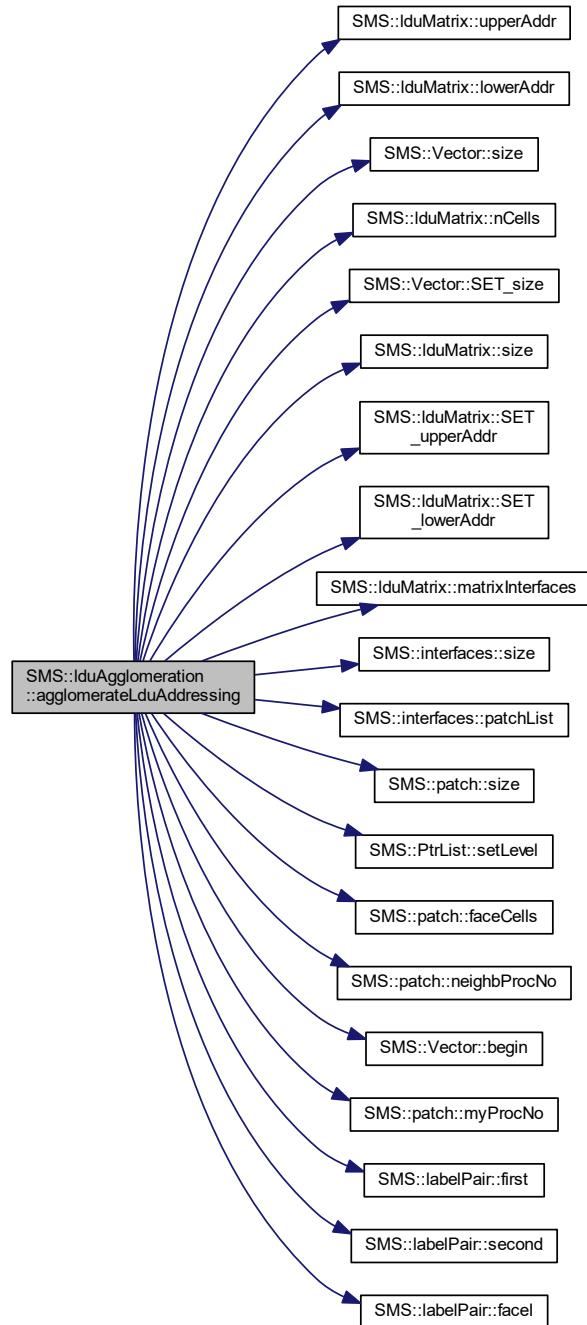
```
void SMS::lduAgglomeration::agglomerate (
    const scalarField & weights ) [virtual]
```

Implements [SMS::matrix::agglomeration](#).

6.6.2.3 agglomerateLduAddressing()

```
void SMS::lduAgglomeration::agglomerateLduAddressing (
    const label fineLevelIndex ) [private]
```

Here is the call graph for this function:

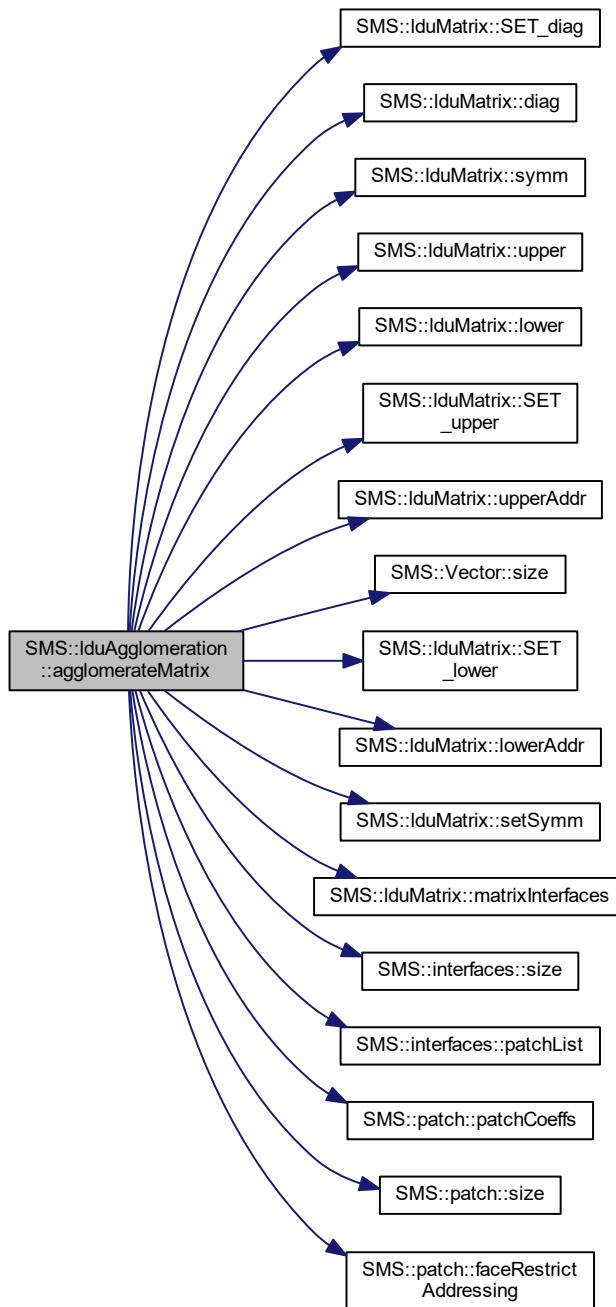


6.6.2.4 agglomerateMatrix()

```
void SMS::lduAgglomeration::agglomerateMatrix (
    const label fineLevelIndex ) [private], [virtual]
```

Implements [SMS::matrix::agglomeration](#).

Here is the call graph for this function:



6.6.2.5 coarseMatrixLevels()

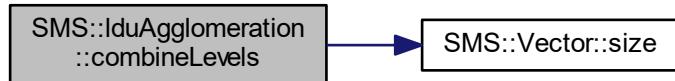
```
virtual const lduMatrix& SMS::lduAgglomeration::coarseMatrixLevels (
    const label leveli ) const [inline], [virtual]
```

Implements [SMS::matrix::agglomeration](#).

6.6.2.6 combineLevels()

```
void SMS::lduAgglomeration::combineLevels (
    const label curLevel ) [private]
```

Here is the call graph for this function:



6.6.2.7 compactLevels()

```
void SMS::lduAgglomeration::compactLevels (
    const label nCreatedLevels ) [private]
```

6.6.2.8 continueAgglomerating()

```
bool SMS::lduAgglomeration::continueAgglomerating (
    const label nCoarseCells ) const [private]
```

6.6.2.9 faceRestrictAddressing()

```
virtual const labelField& SMS::lduAgglomeration::faceRestrictAddressing (
    const label leveli ) const [inline], [virtual]
```

Implements [SMS::matrix::agglomeration](#).

6.6.2.10 matrixLevel()

```
const SMS::lduMatrix & SMS::lduAgglomeration::matrixLevel (
    const label leveli ) const [private]
```

6.6.2.11 restrictAddressing()

```
virtual const labelField& SMS::lduAgglomeration::restrictAddressing (
    const label leveli ) const [inline], [virtual]
```

Implements [SMS::matrix::agglomeration](#).

6.6.2.12 SET_maxLevels()

```
void SMS::lduAgglomeration::SET_maxLevels (
    const label i ) [virtual]
```

Implements [SMS::matrix::agglomeration](#).

6.6.2.13 SET_nCellsInCoarsestLevel()

```
void SMS::lduAgglomeration::SET_nCellsInCoarsestLevel (
    const label i ) [virtual]
```

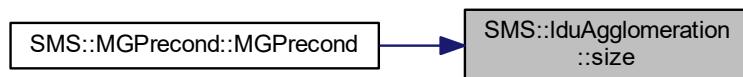
Implements [SMS::matrix::agglomeration](#).

6.6.2.14 size()

```
virtual label SMS::lduAgglomeration::size () const [inline], [virtual]
```

Implements [SMS::matrix::agglomeration](#).

Here is the caller graph for this function:



6.6.3 Member Data Documentation

6.6.3.1 coarseMatrixLevels_

```
PtrList<lduMatrix> SMS::lduAgglomeration::coarseMatrixLevels_ [private]
```

6.6.3.2 faceRestrictAddressing_

```
PtrList<labelField> SMS::lduAgglomeration::faceRestrictAddressing_ [private]
```

6.6.3.3 finestMatrix_

```
const lduMatrix& SMS::lduAgglomeration::finestMatrix_ [private]
```

6.6.3.4 forward_

```
bool SMS::lduAgglomeration::forward_ [static], [private]
```

6.6.3.5 maxLevels_

```
label SMS::lduAgglomeration::maxLevels_ [private]
```

6.6.3.6 mergeLevels_

```
label SMS::lduAgglomeration::mergeLevels_ [private]
```

6.6.3.7 nCells_

```
labelField SMS::lduAgglomeration::nCells_ [private]
```

6.6.3.8 `nCellsInCoarsestLevel_`

```
label SMS::lduAgglomeration::nCellsInCoarsestLevel_ [private]
```

6.6.3.9 `nCreatedLevels_`

```
label SMS::lduAgglomeration::nCreatedLevels_ [private]
```

6.6.3.10 `patchFaceRestrictAddressing_`

```
PtrList<labelField> SMS::lduAgglomeration::patchFaceRestrictAddressing_ [private]
```

6.6.3.11 `restrictAddressing_`

```
PtrList<labelField> SMS::lduAgglomeration::restrictAddressing_ [private]
```

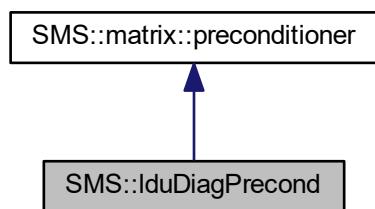
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomeration.hpp
- SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerate.cpp
- SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerateMatrix.cpp
- SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomeration.cpp
- SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerationAddressing.cpp
- SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerationCombineLevels.cpp

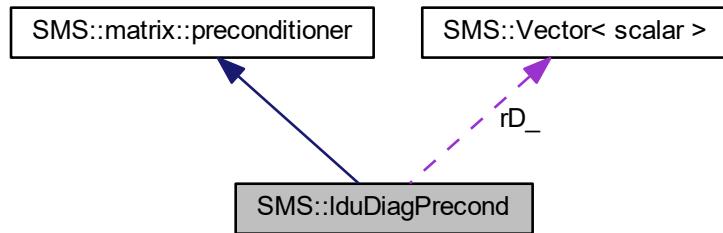
6.7 SMS::lduDiagPrecond Class Reference

```
#include <lduDiagPrecond.hpp>
```

Inheritance diagram for SMS::lduDiagPrecond:



Collaboration diagram for SMS::lduDiagPrecond:



Public Member Functions

- `lduDiagPrecond (const lduMatrix &A)`
- `virtual ~lduDiagPrecond ()`
- `virtual void precondition (scalarField &w, const scalarField &r) const`

Private Attributes

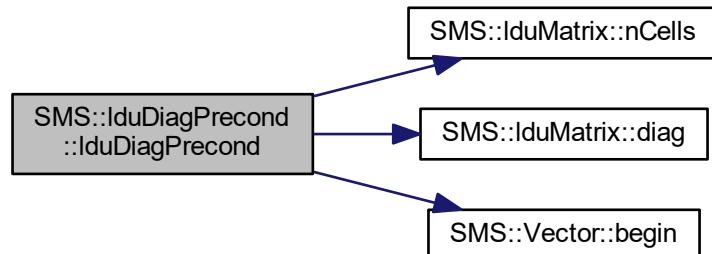
- `scalarField rD_`

6.7.1 Constructor & Destructor Documentation

6.7.1.1 `lduDiagPrecond()`

```
SMS::lduDiagPrecond::lduDiagPrecond (
    const lduMatrix & A )
```

Here is the call graph for this function:



6.7.1.2 ~lduDiagPrecond()

```
virtual SMS::lduDiagPrecond::~lduDiagPrecond ( ) [inline], [virtual]
```

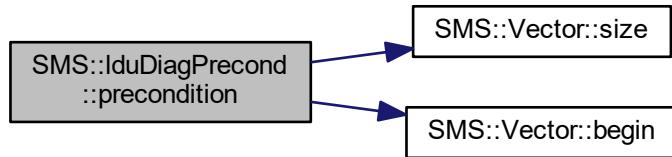
6.7.2 Member Function Documentation

6.7.2.1 precondition()

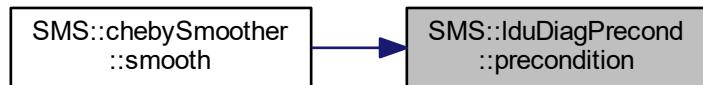
```
void SMS::lduDiagPrecond::precondition (
    scalarField & w,
    const scalarField & r ) const [virtual]
```

Reimplemented from [SMS::matrix::preconditioner](#).

Here is the call graph for this function:



Here is the caller graph for this function:



6.7.3 Member Data Documentation

6.7.3.1 rD_

```
scalarField SMS::lduDiagPrecond::rD_ [private]
```

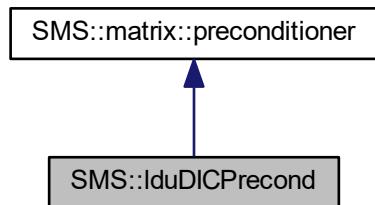
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDiagonal/[lduDiagPrecond.hpp](#)
- SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDiagonal/[lduDiagPrecond.cpp](#)

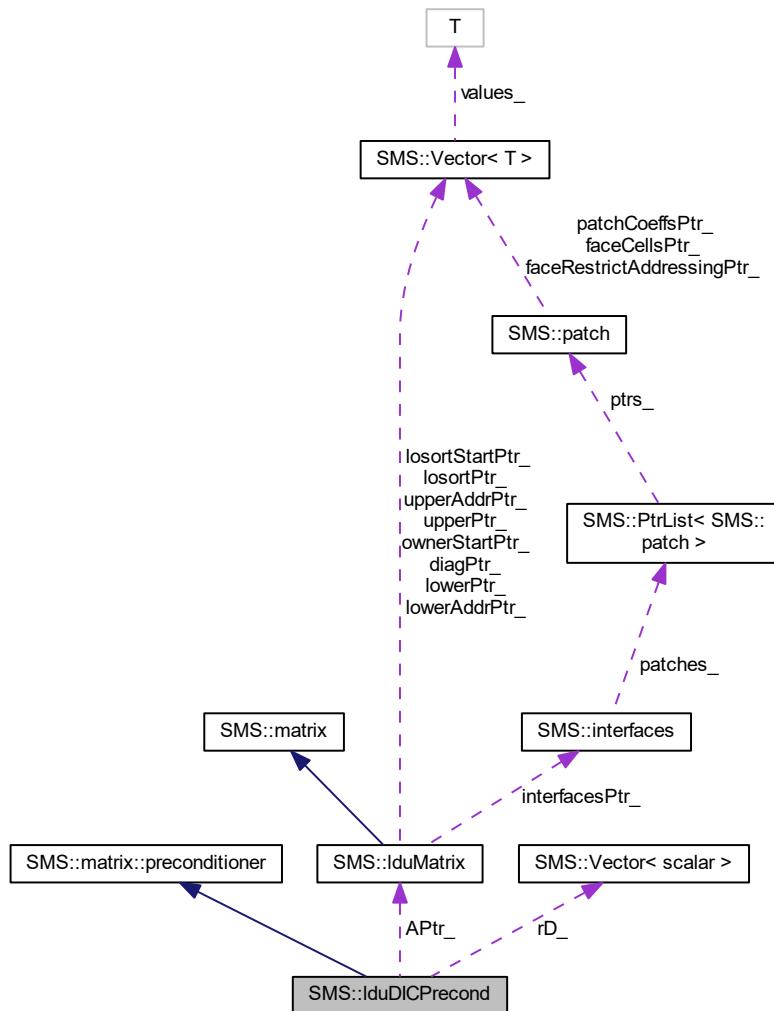
6.8 SMS::lduDICPrecond Class Reference

```
#include <lduDICPrecond.hpp>
```

Inheritance diagram for SMS::lduDICPrecond:



Collaboration diagram for SMS::IduDICPrecond:



Public Member Functions

- `IduDICPrecond (const IduMatrix &A)`
- `virtual ~IduDICPrecond ()`
- `virtual void precondition (scalarField &w, const scalarField &r) const`

Static Public Member Functions

- `static void calcReciprocalID (scalarField &rD, const IduMatrix &A)`

Private Attributes

- `scalarField rD_`
- `const IduMatrix * APtr_`

6.8.1 Constructor & Destructor Documentation

6.8.1.1 lduDICPrecond()

```
SMS::lduDICPrecond::lduDICPrecond (
    const lduMatrix & A )
```

6.8.1.2 ~lduDICPrecond()

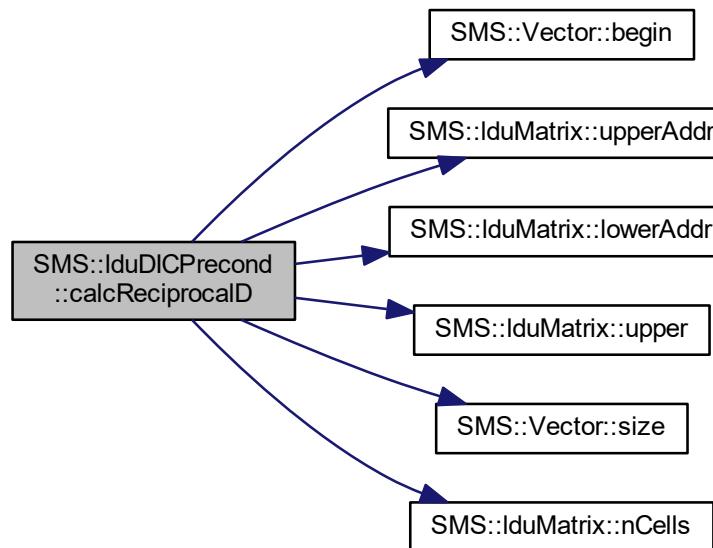
```
virtual SMS::lduDICPrecond::~lduDICPrecond () [inline], [virtual]
```

6.8.2 Member Function Documentation

6.8.2.1 calcReciprocalD()

```
void SMS::lduDICPrecond::calcReciprocalD (
    scalarField & rD,
    const lduMatrix & A ) [static]
```

Here is the call graph for this function:

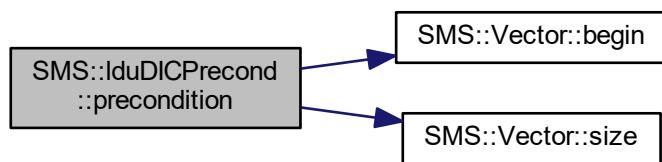


6.8.2.2 precondition()

```
void SMS::lduDICPrecond::precondition (
    scalarField & w,
    const scalarField & r ) const [virtual]
```

Reimplemented from [SMS::matrix::preconditioner](#).

Here is the call graph for this function:



6.8.3 Member Data Documentation

6.8.3.1 APtr_

```
const lduMatrix* SMS::lduDICPrecond::APtr_ [private]
```

6.8.3.2 rD_

```
scalarField SMS::lduDICPrecond::rD_ [private]
```

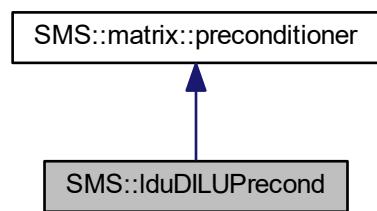
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDIC/[lduDICPrecond.hpp](#)
- SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDIC/[lduDICPrecond.cpp](#)

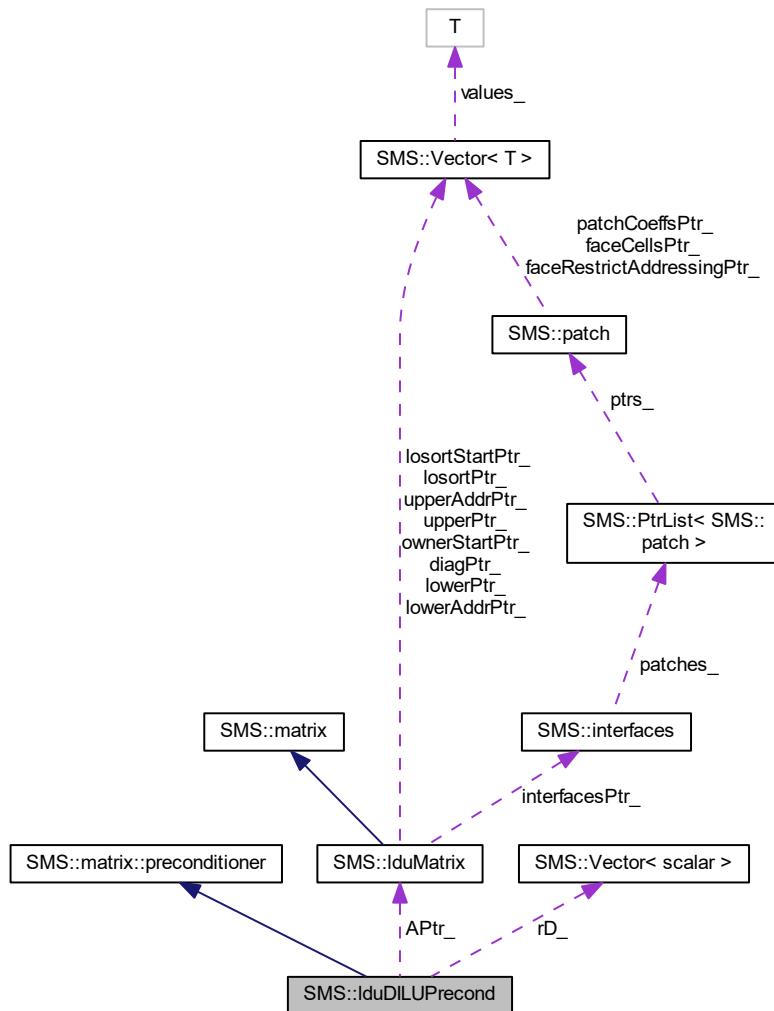
6.9 SMS::IduDILUPrecond Class Reference

```
#include <lduDILUPrecond.hpp>
```

Inheritance diagram for SMS::IduDILUPrecond:



Collaboration diagram for SMS::IduDILUPrecond:



Public Member Functions

- `IduDILUPrecond (const IduMatrix &A)`
- `virtual ~IduDILUPrecond ()`
- `virtual void precondition (scalarField &w, const scalarField &r) const`

Static Public Member Functions

- `static void calcReciprocalID (scalarField &rD, const IduMatrix &A)`

Private Attributes

- `scalarField rD_`
- `const IduMatrix * APtr_`

6.9.1 Constructor & Destructor Documentation

6.9.1.1 lduDILUPrecond()

```
SMS::lduDILUPrecond::lduDILUPrecond (
    const lduMatrix & A )
```

6.9.1.2 ~lduDILUPrecond()

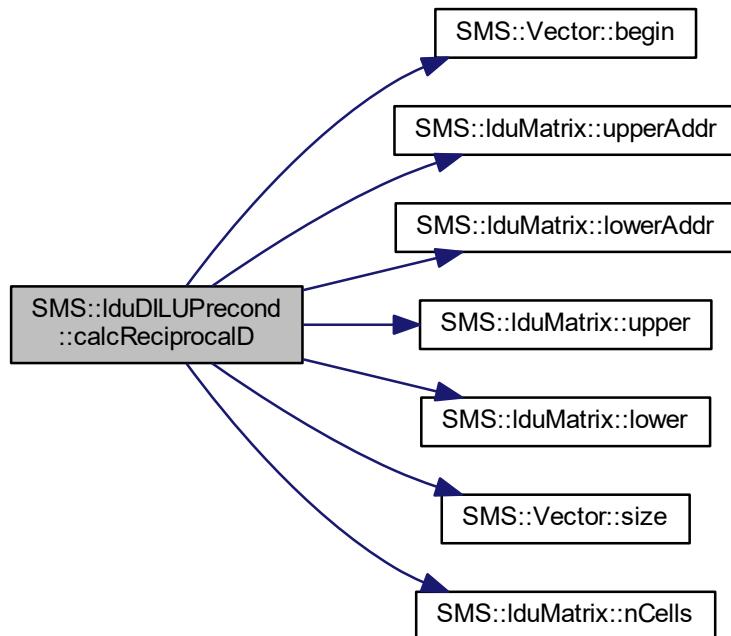
```
virtual SMS::lduDILUPrecond::~lduDILUPrecond ( ) [inline], [virtual]
```

6.9.2 Member Function Documentation

6.9.2.1 calcReciprocalD()

```
void SMS::lduDILUPrecond::calcReciprocalD (
    scalarField & rD,
    const lduMatrix & A ) [static]
```

Here is the call graph for this function:

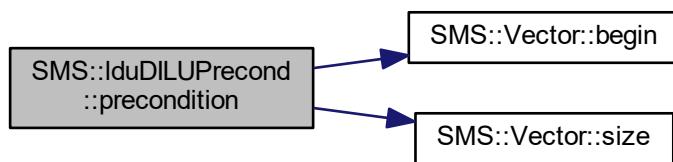


6.9.2.2 precondition()

```
void SMS::lduDILUPrecond::precondition (
    scalarField & w,
    const scalarField & r ) const [virtual]
```

Reimplemented from [SMS::matrix::preconditioner](#).

Here is the call graph for this function:



6.9.3 Member Data Documentation

6.9.3.1 APtr_

```
const lduMatrix* SMS::lduDILUPrecond::APtr_ [private]
```

6.9.3.2 rD_

```
scalarField SMS::lduDILUPrecond::rD_ [private]
```

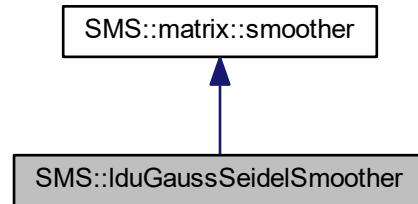
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDILU/lduDILUPrecond.hpp
- SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDILU/lduDILUPrecond.cpp

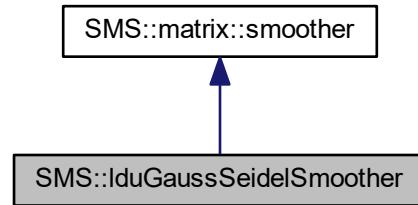
6.10 SMS::IduGaussSeidelSmoothen Class Reference

```
#include <lduGaussSeidelSmoothen.hpp>
```

Inheritance diagram for SMS::IduGaussSeidelSmoothen:



Collaboration diagram for SMS::IduGaussSeidelSmoothen:



Public Member Functions

- `IduGaussSeidelSmoothen ()`
- virtual `~IduGaussSeidelSmoothen ()`
- virtual void `smooth (scalarField &x, const matrix &A, const scalarField &b, const label nSweeps) const`
- void `smooth (scalarField &x, const IduMatrix &A, const scalarField &b, const label nSweeps) const`
- virtual void `init () const`

6.10.1 Constructor & Destructor Documentation

6.10.1.1 `lduGaussSeidelSmoothen()`

```
SMS::lduGaussSeidelSmoothen::lduGaussSeidelSmoothen ( ) [inline]
```

6.10.1.2 `~lduGaussSeidelSmoothen()`

```
virtual SMS::lduGaussSeidelSmoothen::~lduGaussSeidelSmoothen ( ) [inline], [virtual]
```

6.10.2 Member Function Documentation

6.10.2.1 `init()`

```
virtual void SMS::lduGaussSeidelSmoothen::init ( ) const [inline], [virtual]
```

Implements [SMS::matrix::smoother](#).

6.10.2.2 `smooth()` [1/2]

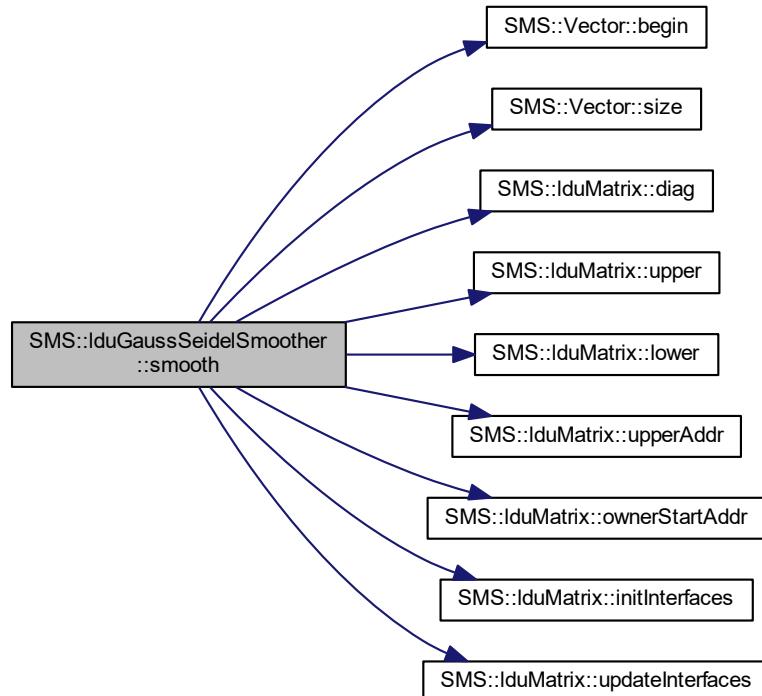
```
void SMS::lduGaussSeidelSmoothen::smooth (
    scalarField & x,
    const matrix & A,
    const scalarField & b,
    const label nSweeps ) const [virtual]
```

Implements [SMS::matrix::smoother](#).

6.10.2.3 `smooth()` [2/2]

```
void SMS::lduGaussSeidelSmoothes::smooth (
    scalarField & x,
    const lduMatrix & A,
    const scalarField & b,
    const label nSweeps ) const
```

Here is the call graph for this function:



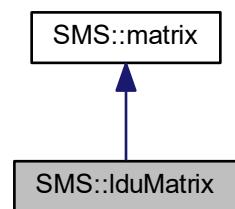
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/smoothers/lduSmoothers/lduGaussSeidelSmoothes/lduGaussSeidelSmoothes.hpp
- SunwayMatrixSolver/src/smoothers/lduSmoothers/lduGaussSeidelSmoothes/lduGaussSeidelSmoothes.cpp

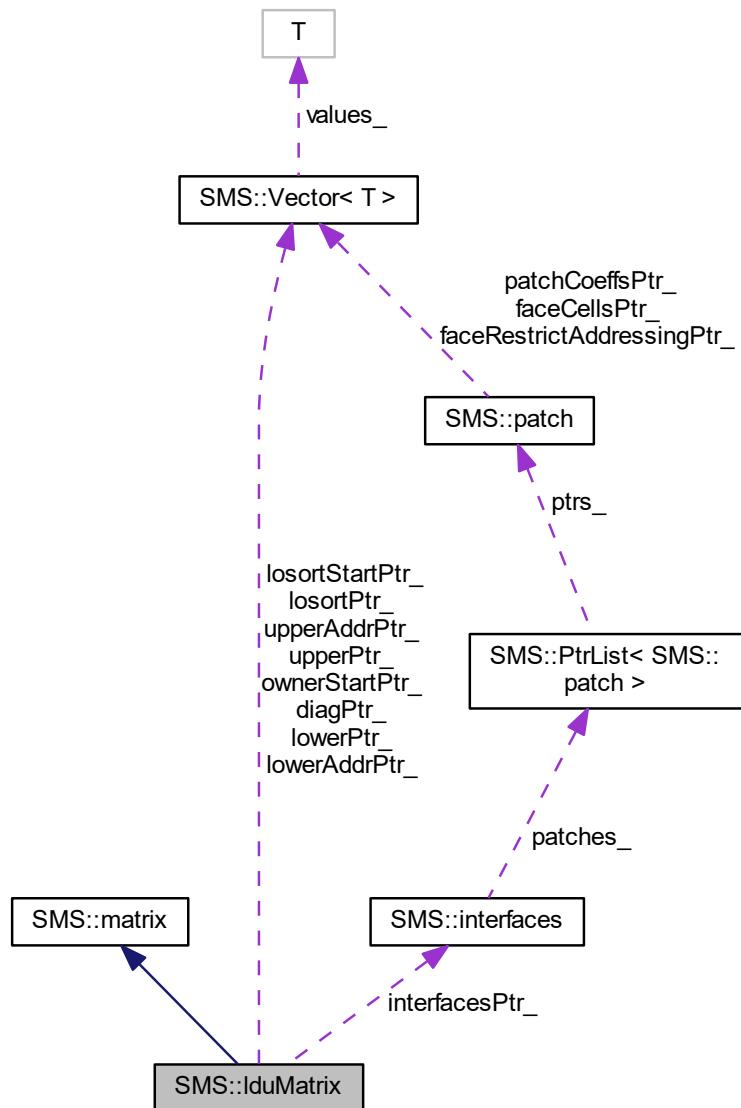
6.11 SMS::lduMatrix Class Reference

```
#include <lduMatrix.hpp>
```

Inheritance diagram for SMS::IduMatrix:



Collaboration diagram for SMS::IduMatrix:



Public Member Functions

- `lDUMatrix ()`
 - `lDUMatrix (const label &nCells, const labelField &lowerAddr, const labelField &upperAddr, const scalarField &lower, const scalarField &diag, const scalarField &upper)`
 - `lDUMatrix (const label &nCells, const labelField &lowerAddr, const labelField &upperAddr, const scalarField &lower, const scalarField &diag, const scalarField &upper, const bool reUse)`
 - `lDUMatrix (const label &nCells, const labelField &lowerAddr, const labelField &upperAddr)`
 - `lDUMatrix (const label &nCells, const labelField &lowerAddr, const labelField &upperAddr, const bool reUse)`
 - `~lDUMatrix ()`
 - virtual `labelField & lowerAddr () const`
 - virtual `labelField & upperAddr () const`

- virtual `scalarField & lower () const`
- virtual `scalarField & diag () const`
- virtual `scalarField & upper () const`
- void `SET_lowerAddr (labelField &newLowerAddr)`
- void `SET_upperAddr (labelField &newUpperAddr)`
- void `SET_lower (scalarField &newLower)`
- void `SET_lower (label newSize)`
- void `SET_upper (scalarField &newUpper)`
- void `SET_upper (label newSize)`
- void `SET_diag (scalarField &newDiag)`
- void `SET_diag (label newSize)`
- `scalarField & diag ()`
- void `setSymm ()`
- virtual bool `symm () const`
- virtual label `size () const`
- virtual label `nCells () const`
- void `size (label size)`
- virtual void `spMV (scalarField &Apsi, const scalarField &psi) const`
- const `labelField & losortAddr () const`
- const `labelField & ownerStartAddr () const`
- const `labelField & losortStartAddr () const`
- virtual `interfaces & matrixInterfaces () const`
- virtual void `matrixInterfaces (interfaces &a)`
- void `matrixInterfaces (const label size)`
- virtual void `initInterfaces (const scalarField &psi) const`
- virtual void `updateInterfaces (scalarField &Apsi) const`
- void `setMatrixCoefficients (const scalarField &diag, const scalarField &upper, const scalarField &lower)`
- void `setMatrixCoefficients (const scalarField &diag, const scalarField &upper, const scalarField &lower, const bool reuse)`
- void `setMatrixTopology (const labelField &upperAddr, const labelField &lowerAddr, const bool reUse)`
- void `setMatrixTopology (const labelField &upperAddr, const labelField &lowerAddr)`

Private Member Functions

- void `calcLosort () const`
- void `calcOwnerStart () const`
- void `calcLosortStart () const`

Private Attributes

- label `nCells_`
- `labelField * lowerAddrPtr_`
- `labelField * upperAddrPtr_`
- `scalarField * lowerPtr_`
- `scalarField * diagPtr_`
- `scalarField * upperPtr_`
- `interfaces * interfacesPtr_`
- `labelField * losortPtr_`
- `labelField * ownerStartPtr_`
- `labelField * losortStartPtr_`

6.11.1 Constructor & Destructor Documentation

6.11.1.1 `lduMatrix()` [1/5]

```
SMS::lduMatrix::lduMatrix ( )
```

6.11.1.2 `lduMatrix()` [2/5]

```
SMS::lduMatrix::lduMatrix (
    const label & nCells,
    const labelField & lowerAddr,
    const labelField & upperAddr,
    const scalarField & lower,
    const scalarField & diag,
    const scalarField & upper )
```

6.11.1.3 `lduMatrix()` [3/5]

```
SMS::lduMatrix::lduMatrix (
    const label & nCells,
    const labelField & lowerAddr,
    const labelField & upperAddr,
    const scalarField & lower,
    const scalarField & diag,
    const scalarField & upper,
    const bool reUse )
```

6.11.1.4 `lduMatrix()` [4/5]

```
SMS::lduMatrix::lduMatrix (
    const label & nCells,
    const labelField & lowerAddr,
    const labelField & upperAddr )
```

6.11.1.5 `lduMatrix()` [5/5]

```
SMS::lduMatrix::lduMatrix (
    const label & nCells,
    const labelField & lowerAddr,
    const labelField & upperAddr,
    const bool reUse )
```

6.11.1.6 ~lduMatrix()

```
SMS::lduMatrix::~lduMatrix ( )
```

6.11.2 Member Function Documentation

6.11.2.1 calcLosort()

```
void SMS::lduMatrix::calcLosort ( ) const [private]
```

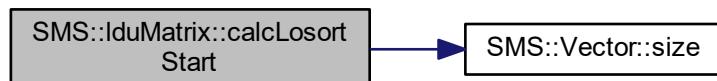
Here is the call graph for this function:



6.11.2.2 calcLosortStart()

```
void SMS::lduMatrix::calcLosortStart ( ) const [private]
```

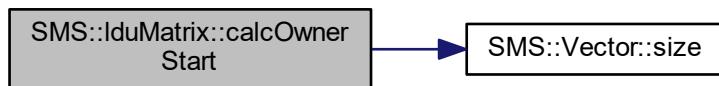
Here is the call graph for this function:



6.11.2.3 calcOwnerStart()

```
void SMS::lduMatrix::calcOwnerStart ( ) const [private]
```

Here is the call graph for this function:

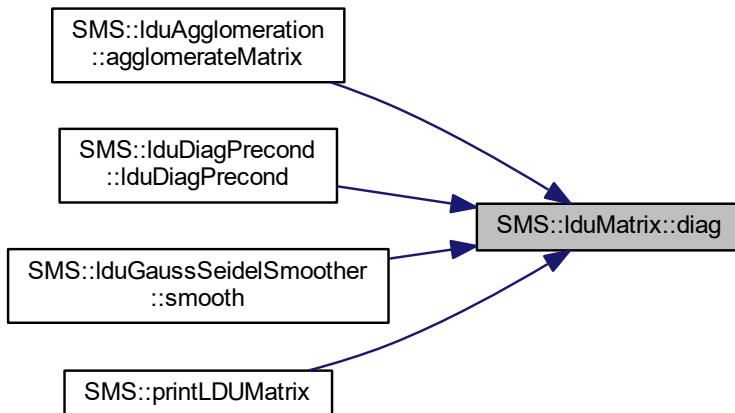


6.11.2.4 diag() [1/2]

```
SMS::scalarField & SMS::lduMatrix::diag ( ) const [virtual]
```

Implements [SMS::matrix](#).

Here is the caller graph for this function:



6.11.2.5 diag() [2/2]

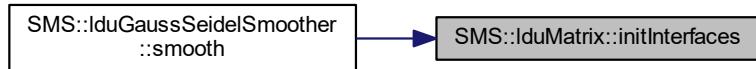
```
scalarField& SMS::lduMatrix::diag ( ) [inline]
```

6.11.2.6 initInterfaces()

```
void SMS::lduMatrix::initInterfaces (
    const scalarField & psi ) const [virtual]
```

Implements [SMS::matrix](#).

Here is the caller graph for this function:



6.11.2.7 losortAddr()

```
const SMS::labelField & SMS::lduMatrix::losortAddr ( ) const
```

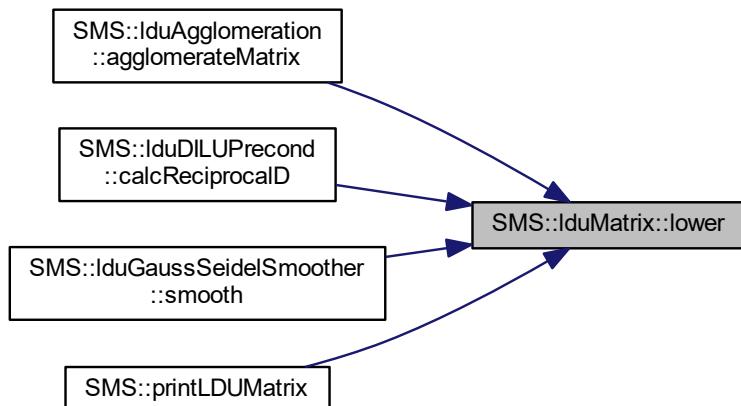
6.11.2.8 losortStartAddr()

```
const SMS::labelField & SMS::lduMatrix::losortStartAddr ( ) const
```

6.11.2.9 lower()

```
SMS::scalarField & SMS::lduMatrix::lower ( ) const [virtual]
```

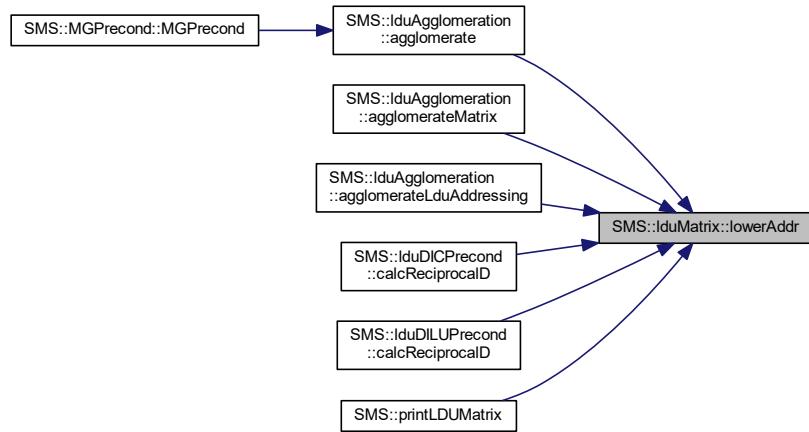
Here is the caller graph for this function:



6.11.2.10 lowerAddr()

```
SMS::labelField & SMS::lduMatrix::lowerAddr ( ) const [virtual]
```

Here is the caller graph for this function:

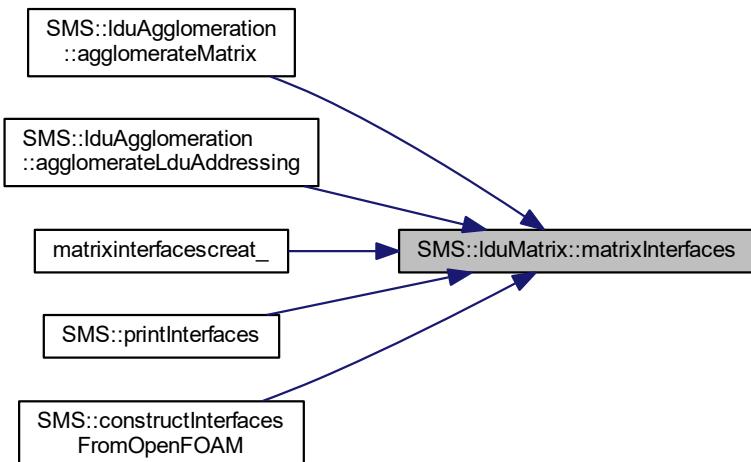


6.11.2.11 matrixInterfaces() [1/3]

```
virtual interfaces& SMS::lduMatrix::matrixInterfaces ( ) const [inline], [virtual]
```

Implements [SMS::matrix](#).

Here is the caller graph for this function:



6.11.2.12 matrixInterfaces() [2/3]

```
virtual void SMS::lduMatrix::matrixInterfaces (
    interfaces & a ) [inline], [virtual]
```

6.11.2.13 matrixInterfaces() [3/3]

```
void SMS::lduMatrix::matrixInterfaces (
    const label size ) [inline]
```

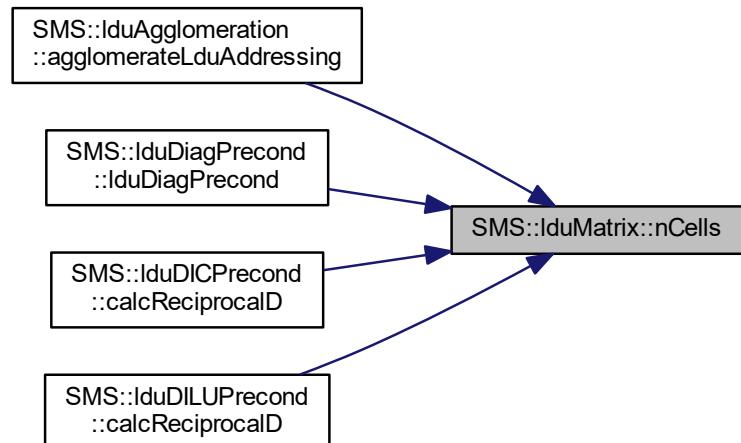
Here is the call graph for this function:



6.11.2.14 nCells()

```
virtual label SMS::lduMatrix::nCells ( ) const [inline], [virtual]
```

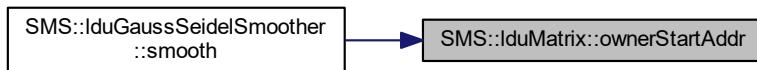
Here is the caller graph for this function:



6.11.2.15 ownerStartAddr()

```
const SMS::labelField & SMS::lduMatrix::ownerStartAddr () const
```

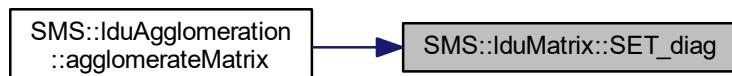
Here is the caller graph for this function:



6.11.2.16 SET_diag() [1/2]

```
void SMS::lduMatrix::SET_diag (
    scalarField & newDiag ) [inline]
```

Here is the caller graph for this function:



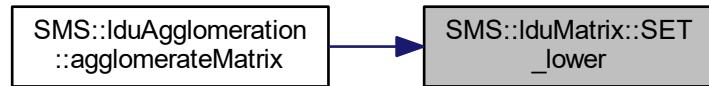
6.11.2.17 SET_diag() [2/2]

```
void SMS::lduMatrix::SET_diag (
    label newSize ) [inline]
```

6.11.2.18 SET_lower() [1/2]

```
void SMS::lduMatrix::SET_lower (
    scalarField & newLower ) [inline]
```

Here is the caller graph for this function:

**6.11.2.19 SET_lower() [2/2]**

```
void SMS::lduMatrix::SET_lower (
    label newSize ) [inline]
```

6.11.2.20 SET_lowerAddr()

```
void SMS::lduMatrix::SET_lowerAddr (
    labelField & newLowerAddr ) [inline]
```

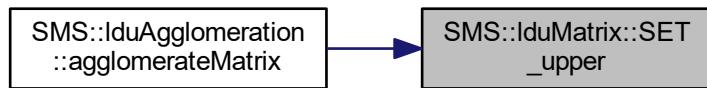
Here is the caller graph for this function:



6.11.2.21 SET_upper() [1/2]

```
void SMS::lduMatrix::SET_upper (
    scalarField & newUpper ) [inline]
```

Here is the caller graph for this function:

**6.11.2.22 SET_upper() [2/2]**

```
void SMS::lduMatrix::SET_upper (
    label newSize ) [inline]
```

6.11.2.23 SET_upperAddr()

```
void SMS::lduMatrix::SET_upperAddr (
    labelField & newUpperAddr ) [inline]
```

Here is the caller graph for this function:



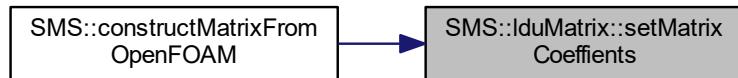
6.11.2.24 setMatrixCoeffients() [1/2]

```
void SMS::lduMatrix::setMatrixCoeffients (
    const scalarField & diag,
    const scalarField & upper,
    const scalarField & lower )
```

Here is the call graph for this function:



Here is the caller graph for this function:

**6.11.2.25 setMatrixCoeffients() [2/2]**

```
void SMS::lduMatrix::setMatrixCoeffients (
    const scalarField & diag,
    const scalarField & upper,
    const scalarField & lower,
    const bool reuse )
```

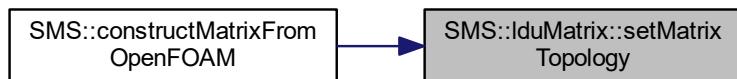
Here is the call graph for this function:



6.11.2.26 setMatrixTopology() [1/2]

```
void SMS::lduMatrix::setMatrixTopology (
    const labelField & upperAddr,
    const labelField & lowerAddr,
    const bool reUse )
```

Here is the caller graph for this function:

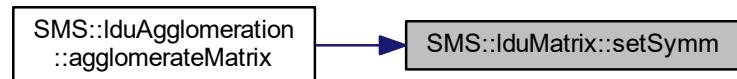
**6.11.2.27 setMatrixTopology() [2/2]**

```
void SMS::lduMatrix::setMatrixTopology (
    const labelField & upperAddr,
    const labelField & lowerAddr )
```

6.11.2.28 setSymm()

```
void SMS::lduMatrix::setSymm () [inline]
```

Here is the caller graph for this function:

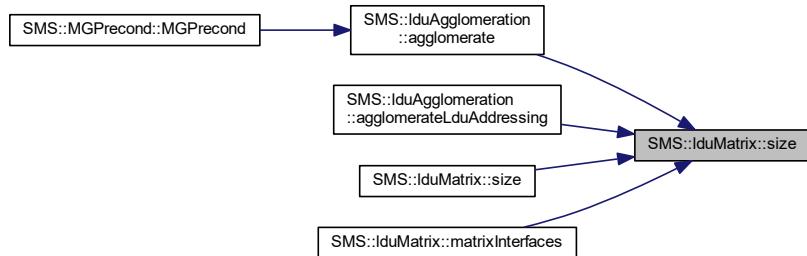


6.11.2.29 size() [1/2]

```
virtual label SMS::lduMatrix::size () const [inline], [virtual]
```

Implements [SMS::matrix](#).

Here is the caller graph for this function:



6.11.2.30 size() [2/2]

```
void SMS::lduMatrix::size (
    label size ) [inline]
```

Here is the call graph for this function:



6.11.2.31 spMV()

```
void SMS::lduMatrix::spMV (
    scalarField & Apsi,
    const scalarField & psi ) const [virtual]
```

Implements [SMS::matrix](#).

Here is the call graph for this function:



Here is the caller graph for this function:

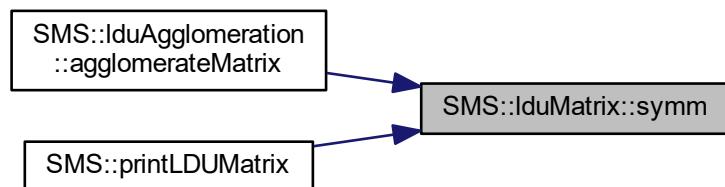


6.11.2.32 symm()

```
virtual bool SMS::lduMatrix::symm ( ) const [inline], [virtual]
```

Implements [SMS::matrix](#).

Here is the caller graph for this function:



6.11.2.33 updateInterfaces()

```
void SMS::lduMatrix::updateInterfaces (
    scalarField & Apsi ) const [virtual]
```

Implements [SMS::matrix](#).

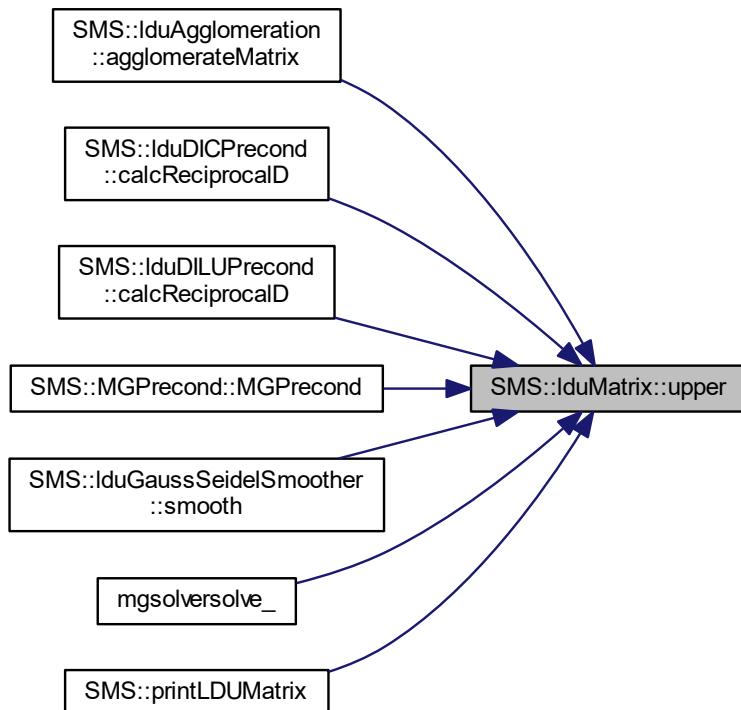
Here is the caller graph for this function:



6.11.2.34 upper()

```
SMS::scalarField & SMS::lduMatrix::upper () const [virtual]
```

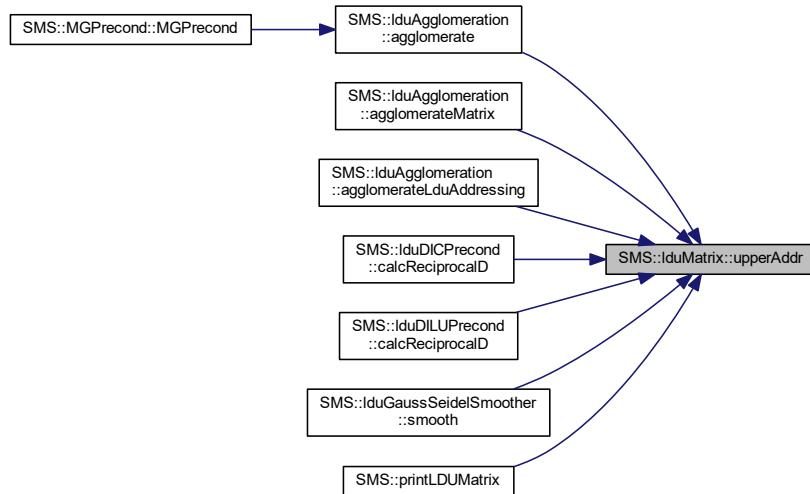
Here is the caller graph for this function:



6.11.2.35 upperAddr()

```
SMS::labelField & SMS::lduMatrix::upperAddr () const [virtual]
```

Here is the caller graph for this function:



6.11.3 Member Data Documentation

6.11.3.1 diagPtr_

```
scalarField * SMS::lduMatrix::diagPtr_ [private]
```

6.11.3.2 interfacesPtr_

```
interfaces* SMS::lduMatrix::interfacesPtr_ [private]
```

6.11.3.3 losortPtr_

```
labelField* SMS::lduMatrix::losortPtr_ [mutable], [private]
```

6.11.3.4 losortStartPtr_

```
labelField* SMS::lduMatrix::losortStartPtr_ [mutable], [private]
```

6.11.3.5 lowerAddrPtr_

```
labelField* SMS::lduMatrix::lowerAddrPtr_ [private]
```

6.11.3.6 lowerPtr_

```
scalarField* SMS::lduMatrix::lowerPtr_ [private]
```

6.11.3.7 nCells_

```
label SMS::lduMatrix::nCells_ [private]
```

6.11.3.8 ownerStartPtr_

```
labelField* SMS::lduMatrix::ownerStartPtr_ [mutable], [private]
```

6.11.3.9 upperAddrPtr_

```
labelField * SMS::lduMatrix::upperAddrPtr_ [private]
```

6.11.3.10 upperPtr_

```
scalarField * SMS::lduMatrix::upperPtr_ [private]
```

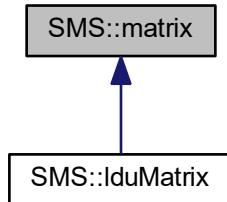
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/LduMatrix/[LduMatrix.hpp](#)
- SunwayMatrixSolver/src/matrix/LduMatrix/[LduMatrix.cpp](#)
- SunwayMatrixSolver/src/matrix/LduMatrix/[LduMatrixSpMV.cpp](#)

6.12 SMS::matrix Class Reference

```
#include <smsMatrix.hpp>
```

Inheritance diagram for SMS::matrix:



Classes

- class [agglomeration](#)
- class [preconditioner](#)
- class [smoother](#)
- class [solver](#)
- class [solverPerformance](#)

Public Member Functions

- virtual [~matrix \(\)](#)
- virtual void [spMV \(scalarField &Apsi, const scalarField &psi\) const =0](#)
- virtual label [size \(\) const =0](#)
- virtual scalarField & [diag \(\) const =0](#)
- virtual bool [symm \(\) const =0](#)
- virtual interfaces & [matrixInterfaces \(\) const =0](#)
- virtual void [initInterfaces \(const scalarField &psi\) const =0](#)
- virtual void [updateInterfaces \(scalarField &Apsi\) const =0](#)

6.12.1 Constructor & Destructor Documentation

6.12.1.1 [~matrix\(\)](#)

```
virtual SMS::matrix::~matrix ( ) [inline], [virtual]
```

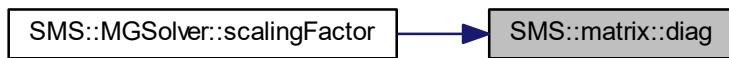
6.12.2 Member Function Documentation

6.12.2.1 diag()

```
virtual scalarField& SMS::matrix::diag ( ) const [pure virtual]
```

Implemented in [SMS::IduMatrix](#).

Here is the caller graph for this function:



6.12.2.2 initInterfaces()

```
virtual void SMS::matrix::initInterfaces (
    const scalarField & psi ) const [pure virtual]
```

Implemented in [SMS::IduMatrix](#).

6.12.2.3 matrixInterfaces()

```
virtual interfaces& SMS::matrix::matrixInterfaces ( ) const [pure virtual]
```

Implemented in [SMS::IduMatrix](#).

6.12.2.4 size()

```
virtual label SMS::matrix::size ( ) const [pure virtual]
```

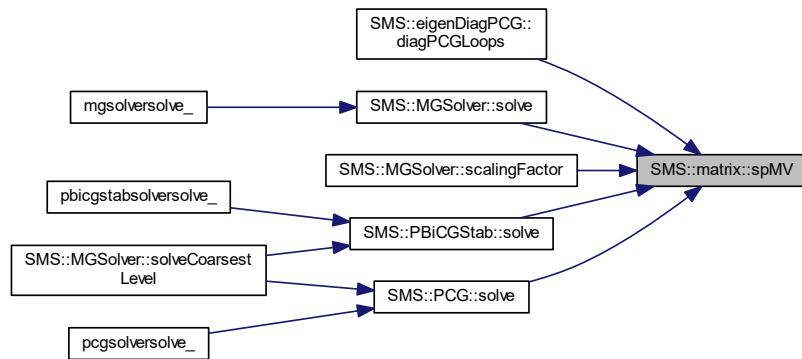
Implemented in [SMS::IduMatrix](#).

6.12.2.5 spMV()

```
virtual void SMS::matrix::spMV (
    scalarField & Apsi,
    const scalarField & psi ) const [pure virtual]
```

Implemented in [SMS::lDUMatrix](#).

Here is the caller graph for this function:



6.12.2.6 symm()

```
virtual bool SMS::matrix::symm ( ) const [pure virtual]
```

Implemented in [SMS::lDUMatrix](#).

Here is the caller graph for this function:



6.12.2.7 updateInterfaces()

```
virtual void SMS::matrix::updateInterfaces (
    scalarField & Apsi ) const [pure virtual]
```

Implemented in [SMS::IduMatrix](#).

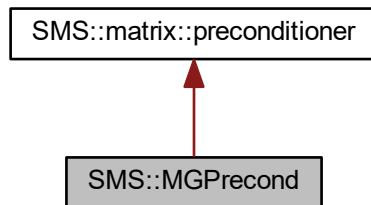
The documentation for this class was generated from the following file:

- SunwayMatrixSolver/src/matrix/[smsMatrix.hpp](#)

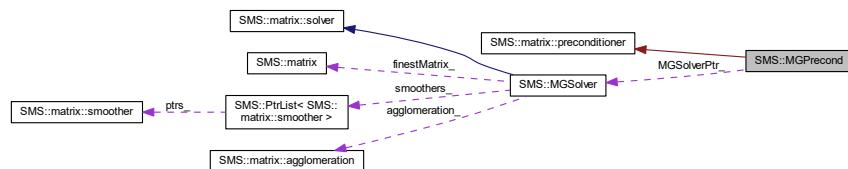
6.13 SMS::MGPrecond Class Reference

```
#include <MGPreconditioner.hpp>
```

Inheritance diagram for SMS::MGPrecond:



Collaboration diagram for SMS::MGPrecond:



Public Member Functions

- [MGPrecond](#) (const [IduMatrix](#) &A)
- virtual [~MGPrecond](#) ()
- virtual void [precondition](#) ([scalarField](#) &wA, const [scalarField](#) &rA) const
- void [set_nVcycles](#) (const label n)

Private Attributes

- label `nVcycles_`
- const `IduMatrix * APtr_`
- const `MGSolver * MGSolverPtr_`

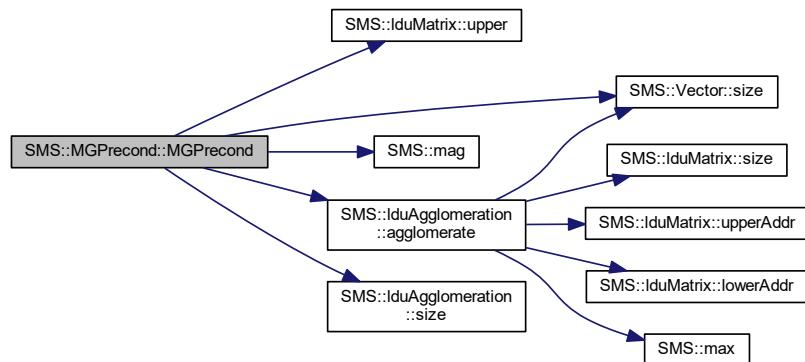
Additional Inherited Members

6.13.1 Constructor & Destructor Documentation

6.13.1.1 MGPrecond()

```
SMS::MGPrecond::MGPrecond (
    const IduMatrix & A )
```

Here is the call graph for this function:



6.13.1.2 ~MGPrecond()

```
virtual SMS::MGPrecond::~MGPrecond ( ) [inline], [virtual]
```

6.13.2 Member Function Documentation

6.13.2.1 precondition()

```
void SMS::MGPrecond::precondition (
    scalarField & wA,
    const scalarField & rA ) const [virtual]
```

Reimplemented from [SMS::matrix::preconditioner](#).

6.13.2.2 set_nVcycles()

```
void SMS::MGPrecond::set_nVcycles (
    const label n ) [inline]
```

6.13.3 Member Data Documentation

6.13.3.1 APtr_

```
const lduMatrix* SMS::MGPrecond::APtr_ [private]
```

6.13.3.2 MGsolverPtr_

```
const MGsolver* SMS::MGPrecond::MGsolverPtr_ [private]
```

6.13.3.3 nVcycles_

```
label SMS::MGPrecond::nVcycles_ [private]
```

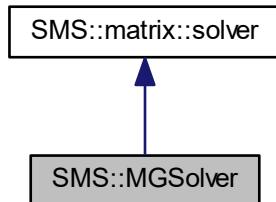
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/preconditioners/MGPreconditioner/[MGPreconditioner.hpp](#)
- SunwayMatrixSolver/src/preconditioners/MGPreconditioner/[MGPreconditioner.cpp](#)

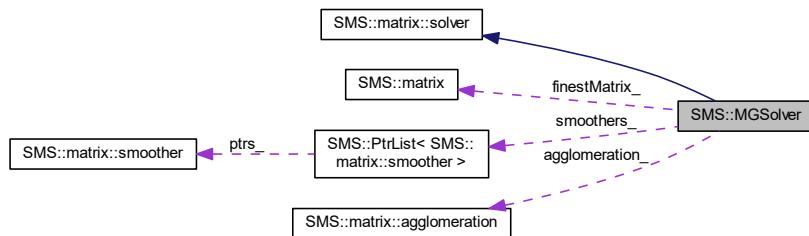
6.14 SMS::MGSolver Class Reference

```
#include <MG.hpp>
```

Inheritance diagram for SMS::MGSolver:



Collaboration diagram for SMS::MGSolver:



Public Member Functions

- `MGSolver (const matrix &A, matrix::agglomeration &agglomerator, const PtrList< matrix::smoother > &smoothers)`
- `virtual ~MGSolver ()`
- `virtual matrix::solverPerformance solve (scalarField &x, const matrix &A, const scalarField &b) const`
- `void SET_nPreSweeps (const label nPreSweeps)`
- `void SET_nPostSweeps (const label nPostSweeps)`
- `void SET_nFinestSweeps (const label nFinestSweeps)`
- `void SET_cacheAgglomeration (bool cacheAgglomeration)`
- `void SET_scaleCorrection (bool scaleCorrection)`
- `void initSmoothers ()`

Protected Member Functions

- `scalar scalingFactor (scalarField &Acf, scalarField &field, const scalarField &source, const matrix &A) const`
- `void initVcycle (PtrList< scalarField > &coarseCorrFields, PtrList< scalarField > &coarseSources) const`
- `void Vcycle (scalarField &psi, const scalarField &source, scalarField &Apsi, scalarField &finestCorrection, scalarField &finestResidual, PtrList< scalarField > &coarseCorrFields, PtrList< scalarField > &coarseSources) const`
- `void solveCoarsestLevel (scalarField &coarsestCorrField, const scalarField &coarsestSource) const`

Protected Attributes

- const `matrix & finestMatrix_`
- bool `cacheAgglomeration_`
- label `nPreSweeps_`
- label `nPostSweeps_`
- label `nFinestSweeps_`
- bool `scaleCorrection_`
- `matrix::agglomeration & agglomeration_`
- const `PtrList< matrix::smoother > & smoothers_`

6.14.1 Constructor & Destructor Documentation

6.14.1.1 `MGSolver()`

```
SMS::MGSolver::MGSolver (
    const matrix & A,
    matrix::agglomeration & agglomerator,
    const PtrList< matrix::smoother > & smoothers )
```

6.14.1.2 `~MGSolver()`

```
SMS::MGSolver::~MGSolver ( ) [virtual]
```

6.14.2 Member Function Documentation

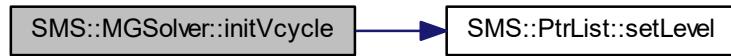
6.14.2.1 `initSmoothers()`

```
void SMS::MGSolver::initSmoothers ( )
```

6.14.2.2 initVcycle()

```
void SMS::MGSolver::initVcycle (
    PtrList< scalarField > & coarseCorrFields,
    PtrList< scalarField > & coarseSources ) const [protected]
```

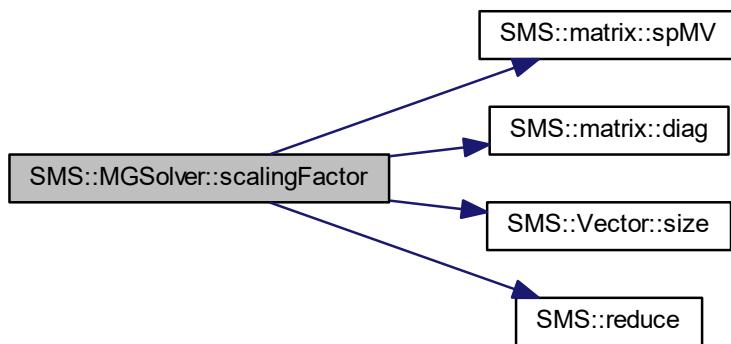
Here is the call graph for this function:



6.14.2.3 scalingFactor()

```
SMS::scalar SMS::MGSolver::scalingFactor (
    scalarField & Acf,
    scalarField & field,
    const scalarField & source,
    const matrix & A ) const [protected]
```

Here is the call graph for this function:



6.14.2.4 SET_cacheAgglomeration()

```
void SMS::MGSolver::SET_cacheAgglomeration (
    bool cacheAgglomeration )
```

6.14.2.5 SET_nFinestSweeps()

```
void SMS::MGSolver::SET_nFinestSweeps (
    const label nFinestSweeps )
```

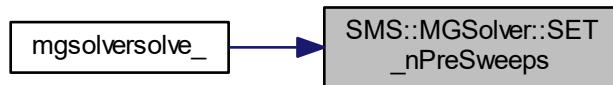
6.14.2.6 SET_nPostSweeps()

```
void SMS::MGSolver::SET_nPostSweeps (
    const label nPostSweeps )
```

6.14.2.7 SET_nPreSweeps()

```
void SMS::MGSolver::SET_nPreSweeps (
    const label nPreSweeps )
```

Here is the caller graph for this function:



6.14.2.8 SET_scaleCorrection()

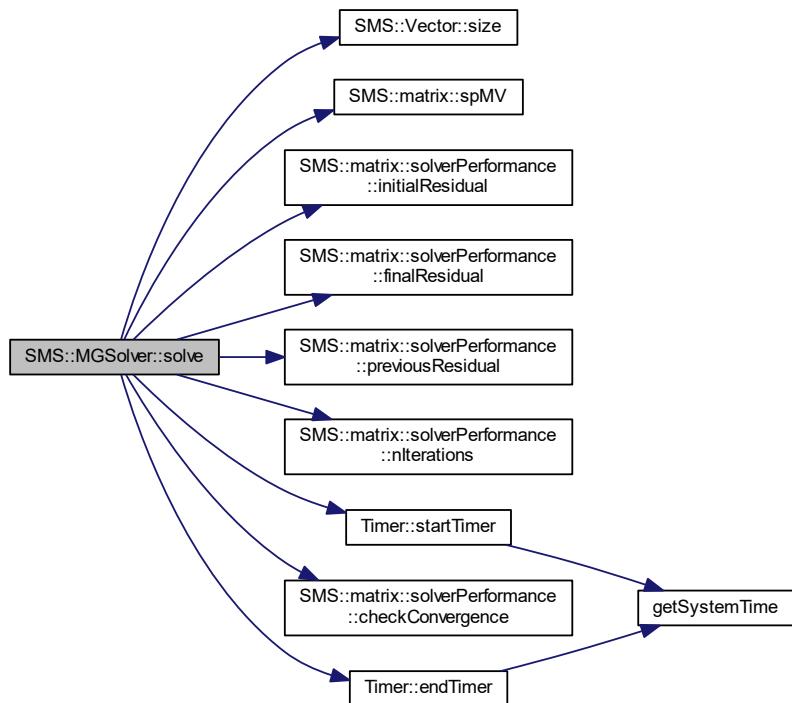
```
void SMS::MGSolver::SET_scaleCorrection (
    bool scaleCorrection )
```

6.14.2.9 solve()

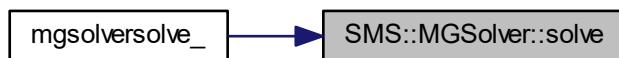
```
SMS::matrix::solverPerformance SMS::MGSolver::solve (
    scalarField & x,
    const matrix & A,
    const scalarField & b ) const [virtual]
```

Implements [SMS::matrix::solver](#).

Here is the call graph for this function:



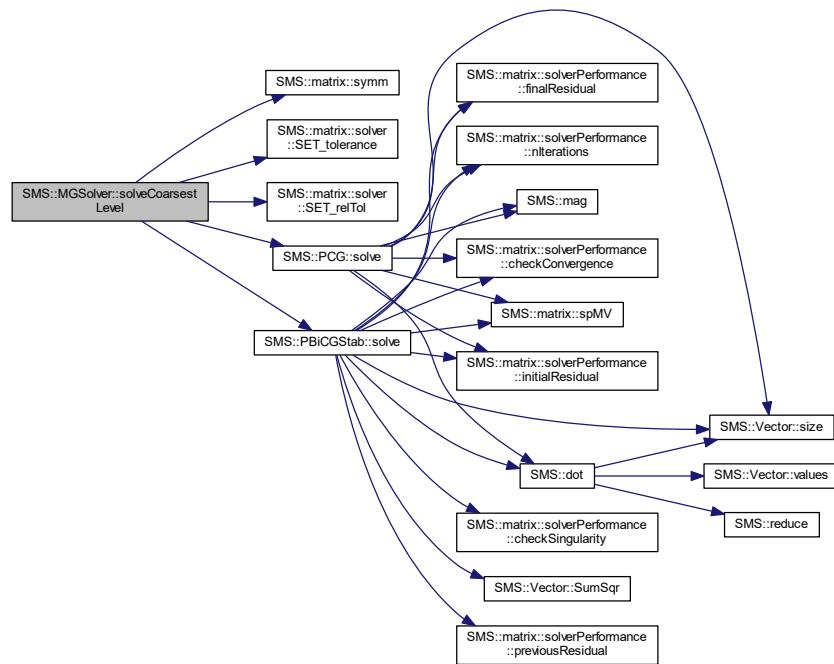
Here is the caller graph for this function:



6.14.2.10 solveCoarsestLevel()

```
void SMS::MGSolver::solveCoarsestLevel (
    scalarField & coarsestCorrField,
    const scalarField & coarsestSource ) const [protected]
```

Here is the call graph for this function:



6.14.2.11 Vcycle()

```
void SMS::MGSolver::Vcycle (
    scalarField & psi,
    const scalarField & source,
    scalarField & Apsi,
    scalarField & finestCorrection,
    scalarField & finestResidual,
    PtrList< scalarField > & coarseCorrFields,
    PtrList< scalarField > & coarseSources ) const [protected]
```

Here is the call graph for this function:



6.14.3 Member Data Documentation

6.14.3.1 `agglomeration_`

```
matrix::agglomeration& SMS::MGSolver::agglomeration_ [protected]
```

6.14.3.2 `cacheAgglomeration_`

```
bool SMS::MGSolver::cacheAgglomeration_ [protected]
```

6.14.3.3 `finestMatrix_`

```
const matrix& SMS::MGSolver::finestMatrix_ [protected]
```

6.14.3.4 `nFinestSweeps_`

```
label SMS::MGSolver::nFinestSweeps_ [protected]
```

6.14.3.5 `nPostSweeps_`

```
label SMS::MGSolver::nPostSweeps_ [protected]
```

6.14.3.6 `nPreSweeps_`

```
label SMS::MGSolver::nPreSweeps_ [protected]
```

6.14.3.7 `scaleCorrection_`

```
bool SMS::MGSolver::scaleCorrection_ [protected]
```

6.14.3.8 smoothers_

```
const PtrList<matrix::smoother>& SMS::MGSolver::smoothers_ [protected]
```

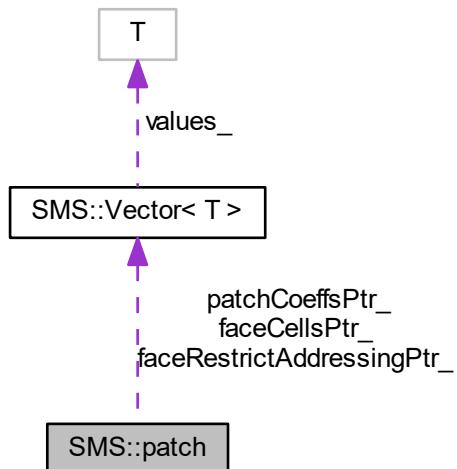
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/solvers/MG/[MG.hpp](#)
- SunwayMatrixSolver/src/solvers/MG/[MG.cpp](#)
- SunwayMatrixSolver/src/solvers/MG/[MGSoLve.cpp](#)
- SunwayMatrixSolver/src/solvers/MG/[scalingFactor.cpp](#)
- SunwayMatrixSolver/src/solvers/MG/[solveCoarsestLevel.cpp](#)
- SunwayMatrixSolver/src/solvers/MG/[Vcycle.cpp](#)

6.15 SMS::patch Class Reference

```
#include <patch.hpp>
```

Collaboration diagram for SMS::patch:



Public Member Functions

- `patch (label size, label myProcNo, label neighbProcNo)`
- virtual `~patch ()`
- label `neighbProcNo () const`
- void `neighbProcNo (const label i)`
- label `myProcNo () const`
- void `myProcNo (const label i)`
- `labelField & faceCells () const`
- void `faceCells (labelField &a) const`

- scalar `patchCoeffs` (const label `facel`) const
- `scalarField & patchCoeffs` () const
- void `patchCoeffs` (`scalarField &a`)
- label `size` () const
- void `size` (const label `i`)
- void `faceRestrictAddressing` (`labelField &a`) const
- `labelField & faceRestrictAddressing` () const

Private Attributes

- label `size_`
- label `myProcNo_`
- label `neighbProcNo_`
- `labelField * faceCellsPtr_`
- `scalarField * patchCoeffsPtr_`
- `labelField * faceRestrictAddressingPtr_`

6.15.1 Constructor & Destructor Documentation

6.15.1.1 `patch()`

```
SMS::patch::patch (
    label size,
    label myProcNo,
    label neighbProcNo )
```

6.15.1.2 `~patch()`

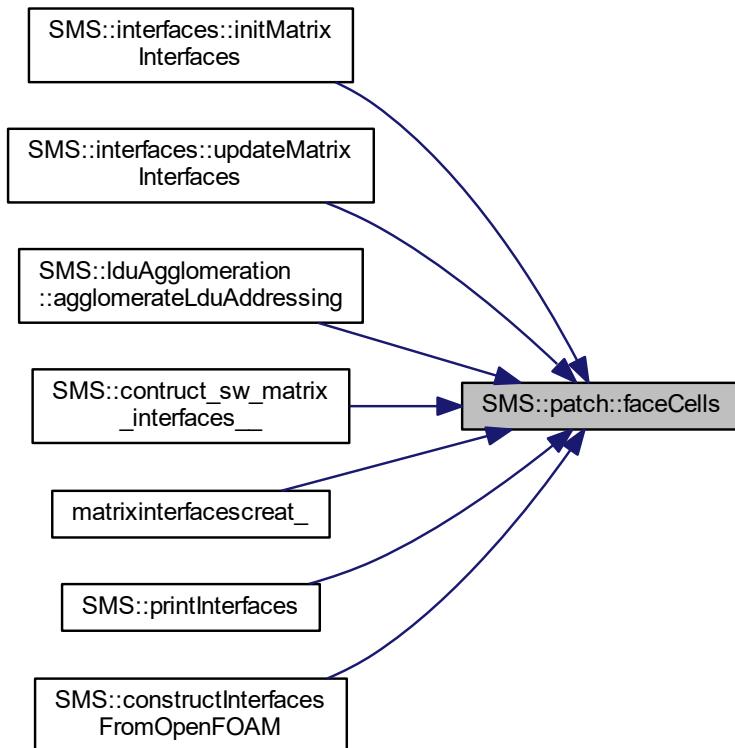
```
SMS::patch::~patch ( ) [virtual]
```

6.15.2 Member Function Documentation

6.15.2.1 faceCells() [1/2]

```
labelField& SMS::patch::faceCells ( ) const [inline]
```

Here is the caller graph for this function:



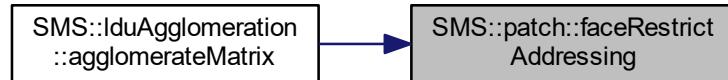
6.15.2.2 faceCells() [2/2]

```
void SMS::patch::faceCells (
    labelField & a ) const [inline]
```

6.15.2.3 faceRestrictAddressing() [1/2]

```
void SMS::patch::faceRestrictAddressing (
    labelField & a ) const [inline]
```

Here is the caller graph for this function:



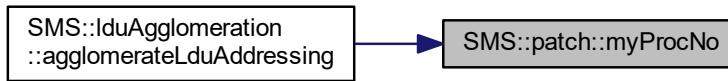
6.15.2.4 faceRestrictAddressing() [2/2]

```
labelField& SMS::patch::faceRestrictAddressing( ) const [inline]
```

6.15.2.5 myProcNo() [1/2]

```
label SMS::patch::myProcNo( ) const [inline]
```

Here is the caller graph for this function:



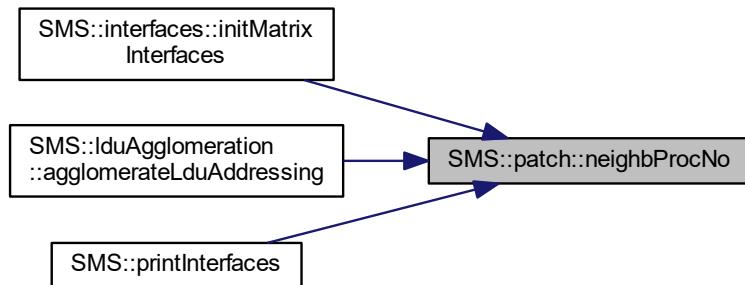
6.15.2.6 myProcNo() [2/2]

```
void SMS::patch::myProcNo( <br>const label i ) [inline]
```

6.15.2.7 `neighbProcNo()` [1/2]

```
label SMS::patch::neighbProcNo ( ) const [inline]
```

Here is the caller graph for this function:

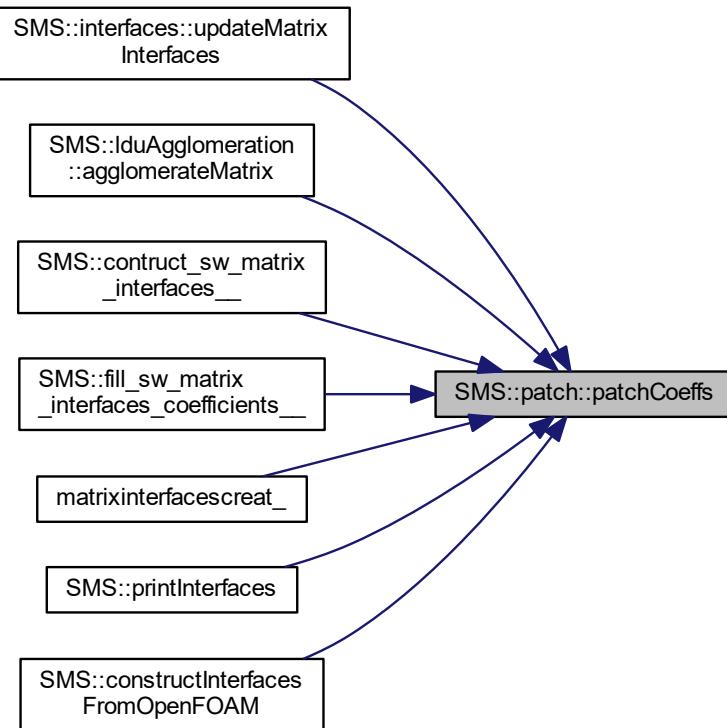
**6.15.2.8 `neighbProcNo()` [2/2]**

```
void SMS::patch::neighbProcNo (
    const label i ) [inline]
```

6.15.2.9 `patchCoeffs()` [1/3]

```
scalar SMS::patch::patchCoeffs (
    const label faceI ) const [inline]
```

Here is the caller graph for this function:



6.15.2.10 `patchCoeffs()` [2/3]

```
scalarField& SMS::patch::patchCoeffs ( ) const [inline]
```

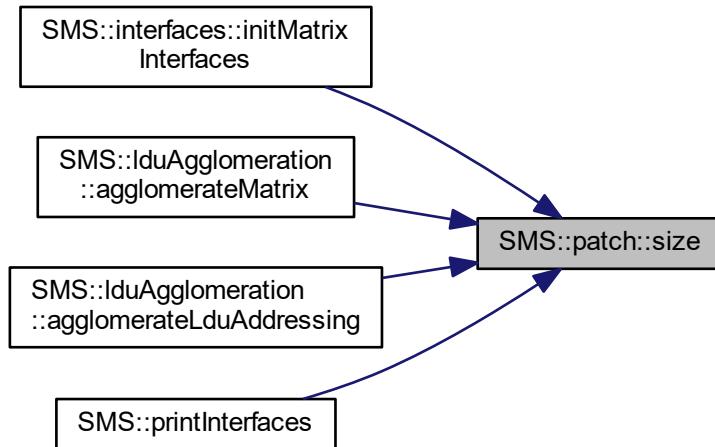
6.15.2.11 `patchCoeffs()` [3/3]

```
void SMS::patch::patchCoeffs (
    scalarField & a ) [inline]
```

6.15.2.12 size() [1/2]

```
label SMS::patch::size ( ) const [inline]
```

Here is the caller graph for this function:



6.15.2.13 size() [2/2]

```
void SMS::patch::size (
    const label i ) [inline]
```

6.15.3 Member Data Documentation

6.15.3.1 faceCellsPtr_

```
labelField* SMS::patch::faceCellsPtr_ [mutable], [private]
```

6.15.3.2 faceRestrictAddressingPtr_

```
labelField* SMS::patch::faceRestrictAddressingPtr_ [mutable], [private]
```

6.15.3.3 myProcNo_

```
label SMS::patch::myProcNo_ [private]
```

6.15.3.4 neighbProcNo_

```
label SMS::patch::neighbProcNo_ [private]
```

6.15.3.5 patchCoeffsPtr_

```
scalarField* SMS::patch::patchCoeffsPtr_ [mutable], [private]
```

6.15.3.6 size_

```
label SMS::patch::size_ [private]
```

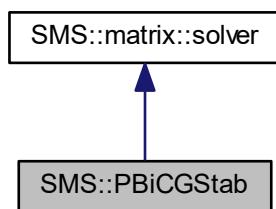
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/interfaces/patch/[patch.hpp](#)
- SunwayMatrixSolver/src/matrix/interfaces/patch/[patch.cpp](#)

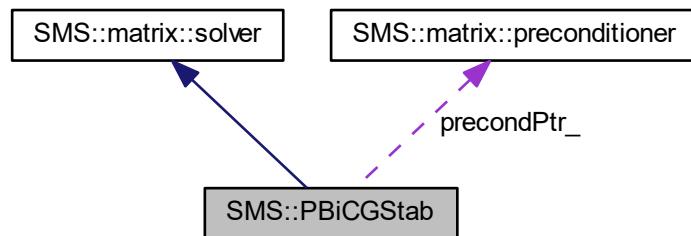
6.16 SMS::PBiCGStab Class Reference

```
#include <PBiCGStab.hpp>
```

Inheritance diagram for SMS::PBiCGStab:



Collaboration diagram for SMS::PBiCGStab:



Public Member Functions

- `PBiCGStab()`
- `PBiCGStab(const label i)`
- `PBiCGStab(matrix::preconditioner &precond)`
- `virtual ~PBiCGStab()`
- `virtual matrix::solverPerformance solve (scalarField &x, const matrix &A, const scalarField &b) const`
- `void SET_preconditioner (matrix::preconditioner &precond)`

Private Attributes

- `bool deletePrecondPtr_`
- `matrix::preconditioner * precondPtr_`

Additional Inherited Members

6.16.1 Constructor & Destructor Documentation

6.16.1.1 `PBiCGStab()` [1/3]

`SMS::PBiCGStab::PBiCGStab()`

6.16.1.2 `PBiCGStab()` [2/3]

`SMS::PBiCGStab::PBiCGStab(`
`const label i)`

6.16.1.3 PBiCGStab() [3/3]

```
SMS::PBiCGStab::PBiCGStab (
    matrix::preconditioner & precond )
```

6.16.1.4 ~PBiCGStab()

```
virtual SMS::PBiCGStab::~PBiCGStab ( ) [inline], [virtual]
```

6.16.2 Member Function Documentation

6.16.2.1 SET_preconditioner()

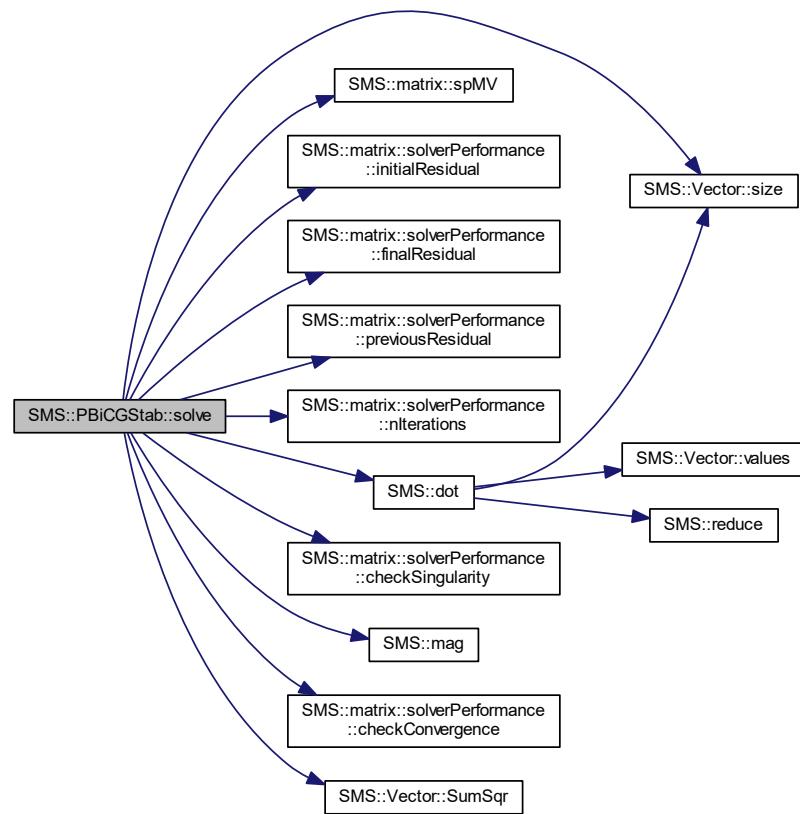
```
void SMS::PBiCGStab::SET_preconditioner (
    matrix::preconditioner & precond ) [inline]
```

6.16.2.2 solve()

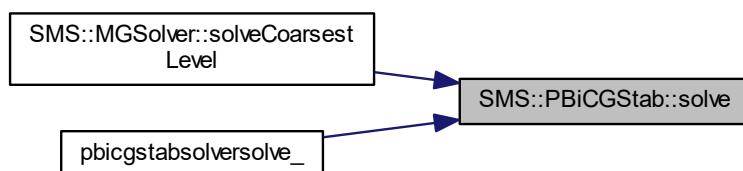
```
SMS::matrix::solverPerformance SMS::PBiCGStab::solve (
    scalarField & x,
    const matrix & A,
    const scalarField & b ) const [virtual]
```

Implements [SMS::matrix::solver](#).

Here is the call graph for this function:



Here is the caller graph for this function:



6.16.3 Member Data Documentation

6.16.3.1 deletePrecondPtr_

```
bool SMS::PBiCGStab::deletePrecondPtr_ [private]
```

6.16.3.2 precondPtr_

```
matrix::preconditioner* SMS::PBiCGStab::precondPtr_ [private]
```

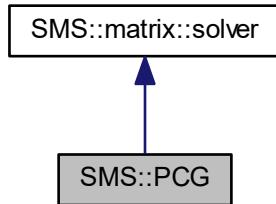
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/solvers/PBiCGStab/[PBiCGStab.hpp](#)
- SunwayMatrixSolver/src/solvers/PBiCGStab/[PBiCGStab.cpp](#)

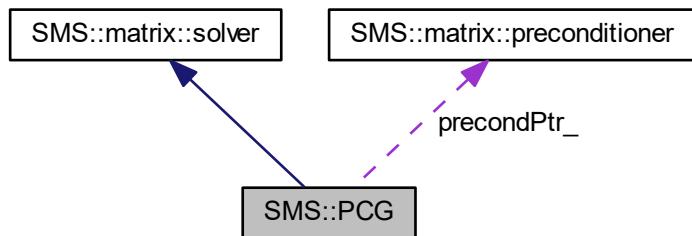
6.17 SMS::PCG Class Reference

```
#include <PCG.hpp>
```

Inheritance diagram for SMS::PCG:



Collaboration diagram for SMS::PCG:



Public Member Functions

- [PCG \(\)](#)
- [PCG \(matrix::preconditioner &precond\)](#)
- [virtual ~PCG \(\)](#)
- [virtual matrix::solverPerformance solve \(scalarField &x, const matrix &A, const scalarField &b\) const](#)

Private Attributes

- bool `deletePrecondPtr_`
- `matrix::preconditioner * precondPtr_`

Additional Inherited Members

6.17.1 Constructor & Destructor Documentation

6.17.1.1 `PCG()` [1/2]

```
SMS::PCG::PCG ( )
```

6.17.1.2 `PCG()` [2/2]

```
SMS::PCG::PCG (
    matrix::preconditioner & precond )
```

6.17.1.3 `~PCG()`

```
virtual SMS::PCG::~PCG ( ) [inline], [virtual]
```

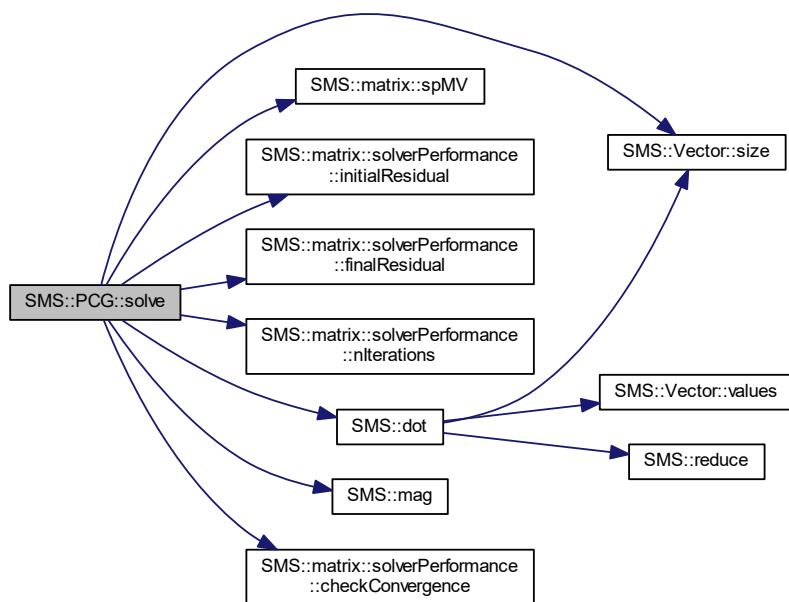
6.17.2 Member Function Documentation

6.17.2.1 solve()

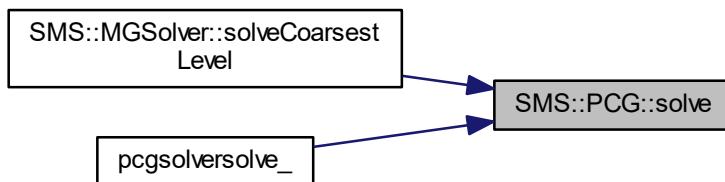
```
SMS::matrix::solverPerformance SMS::PCG::solve (
    scalarField & x,
    const matrix & A,
    const scalarField & b ) const [virtual]
```

Implements [SMS::matrix::solver](#).

Here is the call graph for this function:



Here is the caller graph for this function:



6.17.3 Member Data Documentation

6.17.3.1 deletePrecondPtr_

```
bool SMS::PCG::deletePrecondPtr_ [private]
```

6.17.3.2 precondPtr_

```
matrix::preconditioner* SMS::PCG::precondPtr_ [private]
```

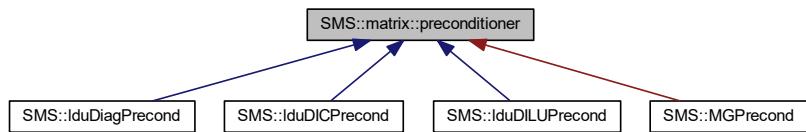
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/solvers/PCG/PCG.hpp
- SunwayMatrixSolver/src/solvers/PCG/PCG.cpp

6.18 SMS::matrix::preconditioner Class Reference

```
#include <smsMatrix.hpp>
```

Inheritance diagram for SMS::matrix::preconditioner:



Public Member Functions

- virtual ~preconditioner ()
- virtual void precondition (scalarField &wA, const scalarField &rA) const

6.18.1 Constructor & Destructor Documentation

6.18.1.1 ~preconditioner()

```
virtual SMS::matrix::preconditioner::~preconditioner () [inline], [virtual]
```

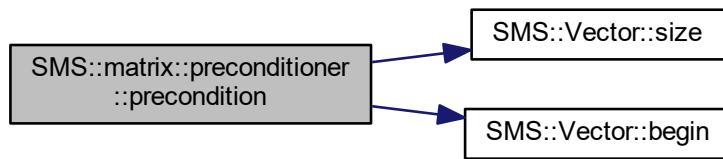
6.18.2 Member Function Documentation

6.18.2.1 precondition()

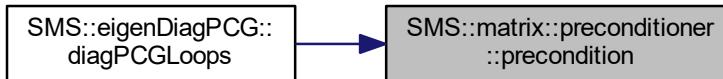
```
void SMS::matrix::preconditioner::precondition (
    scalarField & wA,
    const scalarField & rA ) const [virtual]
```

Reimplemented in [SMS::lDU DICPrecond](#), [SMS::lDU DILUPrecond](#), [SMS::MGPrecond](#), and [SMS::lDU DiagPrecond](#).

Here is the call graph for this function:



Here is the caller graph for this function:



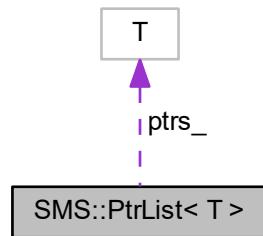
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/[smsMatrix.hpp](#)
- SunwayMatrixSolver/src/matrix/[smsMatrixPrecond.cpp](#)

6.19 SMS::PtrList< T > Class Template Reference

```
#include <PtrList.hpp>
```

Collaboration diagram for SMS::PtrList< T >:



Public Member Functions

- `PtrList ()`
- `PtrList (const label)`
- `~PtrList ()`
- label `size () const`
- void `SET_size (const label i)`
- bool `isEmpty () const`
- `T & operator[] (const label i) const`
- void `setLevel (const label i, T &obj)`
- void `removeLevel (const label i)`

Private Attributes

- label `size_`
- `T ** ptrs_`

6.19.1 Constructor & Destructor Documentation

6.19.1.1 `PtrList()` [1/2]

```
template<class T >
SMS::PtrList< T >::PtrList ( )
```

6.19.1.2 `PtrList()` [2/2]

```
template<class T >
SMS::PtrList< T >::PtrList (
    const label size )
```

6.19.1.3 ~PtrList()

```
template<typename T>
SMS::PtrList< T >::~PtrList ( ) [inline]
```

6.19.2 Member Function Documentation

6.19.2.1 isEmpty()

```
template<class T >
bool SMS::PtrList< T >::isEmpty ( ) const [inline]
```

6.19.2.2 operator[]()

```
template<typename T>
T& SMS::PtrList< T >::operator[ ] (
    const label i ) const [inline]
```

6.19.2.3 removeLevel()

```
template<class T >
void SMS::PtrList< T >::removeLevel (
    const label i )
```

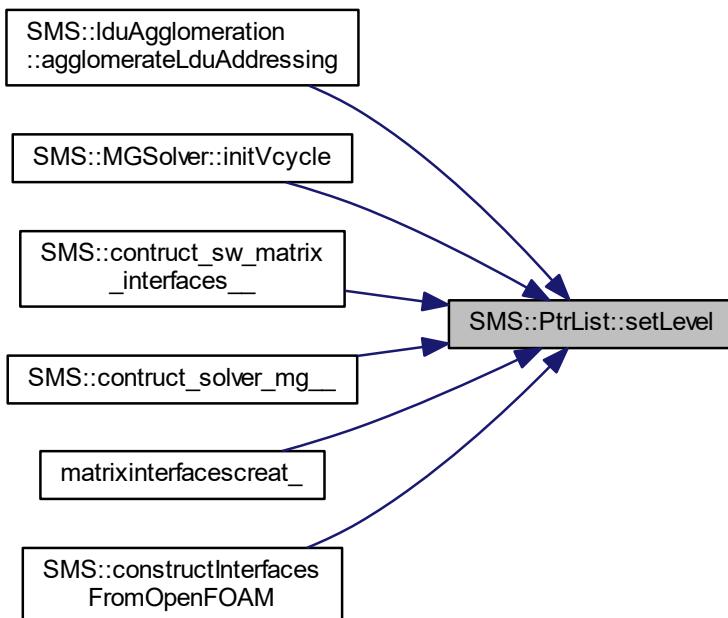
6.19.2.4 SET_size()

```
template<class T >
void SMS::PtrList< T >::SET_size (
    const label i ) [inline]
```

6.19.2.5 setLevel()

```
template<typename T>
void SMS::PtrList< T >::setLevel (
    const label i,
    T & obj )
```

Here is the caller graph for this function:



6.19.2.6 size()

```
template<class T >
label SMS::PtrList< T >::size ( ) const [inline]
```

6.19.3 Member Data Documentation

6.19.3.1 ptrs_

```
template<typename T>
T** SMS::PtrList< T >::ptrs_ [private]
```

6.19.3.2 size_

```
template<typename T>
label SMS::PtrList< T >::size\_ [private]
```

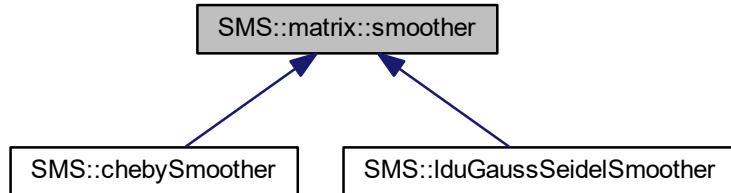
The documentation for this class was generated from the following file:

- SunwayMatrixSolver/src/base/PtrList/[PtrList.hpp](#)

6.20 SMS::matrix::smoother Class Reference

```
#include <smsMatrix.hpp>
```

Inheritance diagram for SMS::matrix::smoother:



Public Member Functions

- [smoother \(\)](#)
- virtual [~smoother \(\)](#)
- virtual void [smooth \(scalarField &x, const matrix &A, const scalarField &b, const label nSweeps\) const =0](#)
- virtual void [init \(\) const =0](#)

6.20.1 Constructor & Destructor Documentation

6.20.1.1 smoother()

```
SMS::matrix::smoother::smoother ( ) [inline]
```

6.20.1.2 ~smoother()

```
virtual SMS::matrix::smoother::~smoother ( ) [inline], [virtual]
```

6.20.2 Member Function Documentation

6.20.2.1 init()

```
virtual void SMS::matrix::smoother::init ( ) const [pure virtual]
```

Implemented in [SMS::chebySmoothen](#), and [SMS::lduGaussSeidelSmoothen](#).

6.20.2.2 smooth()

```
virtual void SMS::matrix::smoother::smooth (
    scalarField & x,
    const matrix & A,
    const scalarField & b,
    const label nSweeps ) const [pure virtual]
```

Implemented in [SMS::chebySmoothen](#), and [SMS::lduGaussSeidelSmoothen](#).

The documentation for this class was generated from the following file:

- SunwayMatrixSolver/src/matrix/[smsMatrix.hpp](#)

6.21 SMS::smsMPI Class Reference

```
#include <smsMPI.hpp>
```

Static Public Member Functions

- static bool [init](#) ()
- static bool & [parRun](#) ()
- static label [nProcs](#) ()
- static label [myProcNo](#) ()
- static void [exitMPI](#) ()
- static MPI_Datatype & [smsLabel](#) ()
- static MPI_Datatype & [smsScalar](#) ()

Static Private Attributes

- static label `myProcNo_` = 0
- static label `nProcs_` = 1
- static bool `parRun_` = false
- static MPI_Datatype `smsLabel_` = MPI_INT
- static MPI_Datatype `smsScalar_` = MPI_DOUBLE

6.21.1 Member Function Documentation

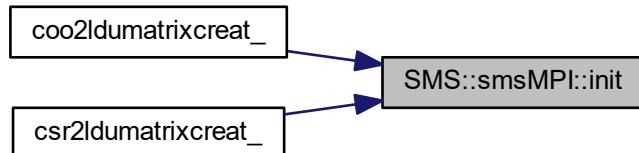
6.21.1.1 `exitMPI()`

```
void SMS::smsMPI::exitMPI ( ) [static]
```

6.21.1.2 `init()`

```
bool SMS::smsMPI::init ( ) [static]
```

Here is the caller graph for this function:



6.21.1.3 `myProcNo()`

```
static label SMS::smsMPI::myProcNo ( ) [inline], [static]
```

6.21.1.4 `nProcs()`

```
static label SMS::smsMPI::nProcs ( ) [inline], [static]
```

6.21.1.5 `parRun()`

```
static bool& SMS::smsMPI::parRun ( ) [inline], [static]
```

6.21.1.6 `smsLabel()`

```
static MPI_Datatype& SMS::smsMPI::smsLabel ( ) [inline], [static]
```

6.21.1.7 `smsScalar()`

```
static MPI_Datatype& SMS::smsMPI::smsScalar ( ) [inline], [static]
```

6.21.2 Member Data Documentation

6.21.2.1 `myProcNo_`

```
SMS::label SMS::smsMPI::myProcNo_ = 0 [static], [private]
```

6.21.2.2 `nProcs_`

```
SMS::label SMS::smsMPI::nProcs_ = 1 [static], [private]
```

6.21.2.3 `parRun_`

```
bool SMS::smsMPI::parRun_ = false [static], [private]
```

6.21.2.4 `smsLabel_`

```
MPI_Datatype SMS::smsMPI::smsLabel_ = MPI_INT [static], [private]
```

6.21.2.5 smsScalar_

```
MPI_Datatype SMS::smsMPI::smsScalar_ = MPI_DOUBLE [static], [private]
```

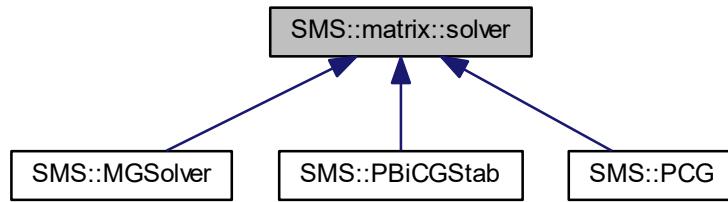
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/base/mpi/[smsMPI.hpp](#)
- SunwayMatrixSolver/src/base/mpi/[smsMPI.cpp](#)

6.22 SMS::matrix::solver Class Reference

```
#include <smsMatrix.hpp>
```

Inheritance diagram for SMS::matrix::solver:



Public Member Functions

- [solver \(\)](#)
- virtual [~solver \(\)](#)
- virtual [solverPerformance solve \(scalarField &x, const matrix &A, const scalarField &b\) const =0](#)
- scalar [normFactor \(const scalarField &source\) const](#)
- label [maxIter \(\) const](#)
- label [minIter \(\) const](#)
- scalar [relTol \(\) const](#)
- scalar [tolerance \(\) const](#)
- void [SET_maxIter \(label maxIter\)](#)
- void [SET_minIter \(label minIter\)](#)
- void [SET_relTol \(scalar relTol\)](#)
- void [SET_tolerance \(scalar tolerance\)](#)

Protected Attributes

- label [maxIter_](#)
- label [minIter_](#)
- scalar [relTol_](#)
- scalar [tolerance_](#)

6.22.1 Constructor & Destructor Documentation

6.22.1.1 solver()

```
SMS::matrix::solver::solver ( )
```

6.22.1.2 ~solver()

```
virtual SMS::matrix::solver::~solver ( ) [inline], [virtual]
```

6.22.2 Member Function Documentation

6.22.2.1 maxIter()

```
label SMS::matrix::solver::maxIter ( ) const [inline]
```

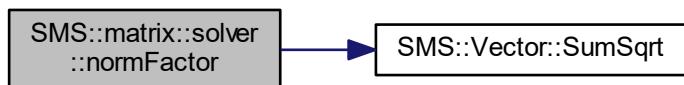
6.22.2.2 minIter()

```
label SMS::matrix::solver::minIter ( ) const [inline]
```

6.22.2.3 normFactor()

```
SMS::scalar SMS::matrix::solver::normFactor (   
     const scalarField & source ) const
```

Here is the call graph for this function:



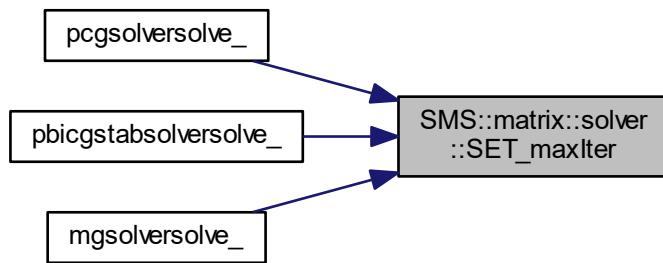
6.22.2.4 relTol()

```
scalar SMS::matrix::solver::relTol ( ) const [inline]
```

6.22.2.5 SET_maxIter()

```
void SMS::matrix::solver::SET_maxIter (
    label maxIter )
```

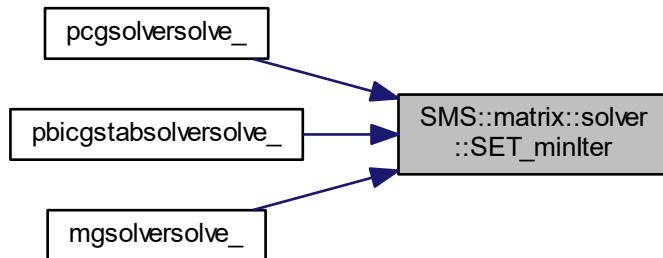
Here is the caller graph for this function:



6.22.2.6 SET_minIter()

```
void SMS::matrix::solver::SET_minIter (
    label minIter )
```

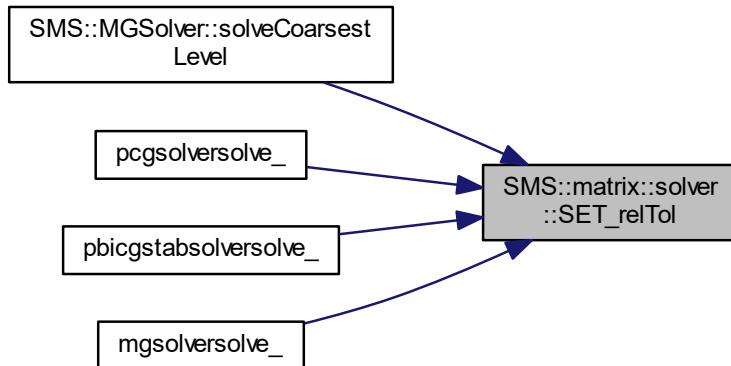
Here is the caller graph for this function:



6.22.2.7 SET_relTol()

```
void SMS::matrix::solver::SET_relTol (
    scalar relTol )
```

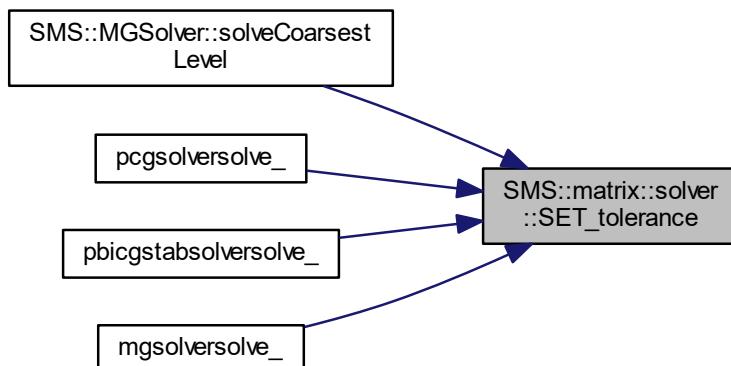
Here is the caller graph for this function:



6.22.2.8 SET_tolerance()

```
void SMS::matrix::solver::SET_tolerance (
    scalar tolerance )
```

Here is the caller graph for this function:



6.22.2.9 solve()

```
virtual solverPerformance SMS::matrix::solver::solve (
    scalarField & x,
    const matrix & A,
    const scalarField & b ) const [pure virtual]
```

Implemented in [SMS::MGSolver](#), [SMS::PBiCGStab](#), and [SMS::PCG](#).

6.22.2.10 tolerance()

```
scalar SMS::matrix::solver::tolerance () const [inline]
```

6.22.3 Member Data Documentation

6.22.3.1 maxIter_

```
label SMS::matrix::solver::maxIter_ [protected]
```

6.22.3.2 minIter_

```
label SMS::matrix::solver::minIter_ [protected]
```

6.22.3.3 relTol_

```
scalar SMS::matrix::solver::relTol_ [protected]
```

6.22.3.4 tolerance_

```
scalar SMS::matrix::solver::tolerance_ [protected]
```

The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/[smsMatrix.hpp](#)
- SunwayMatrixSolver/src/matrix/[smsMatrixSolver.cpp](#)

6.23 SMS::matrix::solverPerformance Class Reference

```
#include <smsMatrix.hpp>
```

Public Member Functions

- `solverPerformance ()`
- `solverPerformance (const scalar iRes, const scalar fRes, const label nIter, const bool converged, const bool singular)`
- scalar `initialResidual () const`
- scalar & `initialResidual ()`
- scalar `finalResidual () const`
- scalar & `finalResidual ()`
- scalar `previousResidual () const`
- scalar & `previousResidual ()`
- label `nIterations () const`
- label & `nIterations ()`
- bool `converged () const`
- bool `singular () const`
- bool `checkConvergence (const scalar tolerance, const scalar relTolerance, const label iter, const label minIter)`
- bool `checkConvergence (const scalar tolerance, const label iter, const label minIter)`
- bool `checkConvergence (const scalar tolerance, const scalar relTolerance)`
- bool `checkSingularity (const scalar residual)`

Private Attributes

- scalar `initialResidual_`
- scalar `finalResidual_`
- scalar `previousResidual_`
- label `nIterations_`
- bool `converged_`
- bool `singular_`

6.23.1 Constructor & Destructor Documentation

6.23.1.1 solverPerformance() [1/2]

```
SMS::matrix::solverPerformance::solverPerformance ( ) [inline]
```

6.23.1.2 solverPerformance() [2/2]

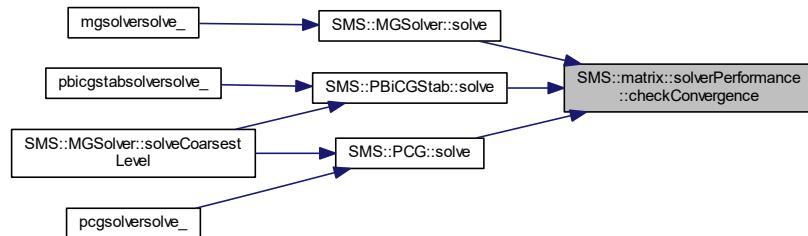
```
SMS::matrix::solverPerformance::solverPerformance (
    const scalar iRes,
    const scalar fRes,
    const label nIter,
    const bool converged,
    const bool singular ) [inline]
```

6.23.2 Member Function Documentation

6.23.2.1 checkConvergence() [1/3]

```
bool SMS::matrix::solverPerformance::checkConvergence (
    const scalar tolerance,
    const scalar relTolerance,
    const label iter,
    const label minIter )
```

Here is the caller graph for this function:



6.23.2.2 checkConvergence() [2/3]

```
bool SMS::matrix::solverPerformance::checkConvergence (
    const scalar tolerance,
    const label iter,
    const label minIter )
```

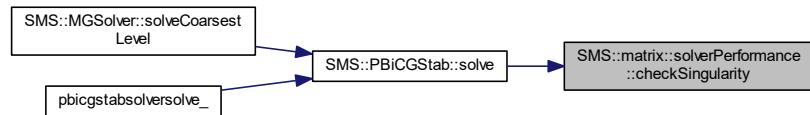
6.23.2.3 checkConvergence() [3/3]

```
bool SMS::matrix::solverPerformance::checkConvergence (
    const scalar tolerance,
    const scalar relTolerance )
```

6.23.2.4 checkSingularity()

```
bool SMS::matrix::solverPerformance::checkSingularity (
    const scalar residual )
```

Here is the caller graph for this function:



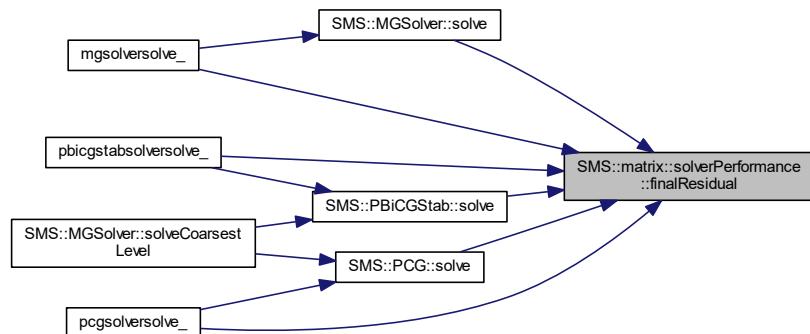
6.23.2.5 converged()

```
bool SMS::matrix::solverPerformance::converged ( ) const [inline]
```

6.23.2.6 finalResidual() [1/2]

```
scalar SMS::matrix::solverPerformance::finalResidual ( ) const [inline]
```

Here is the caller graph for this function:



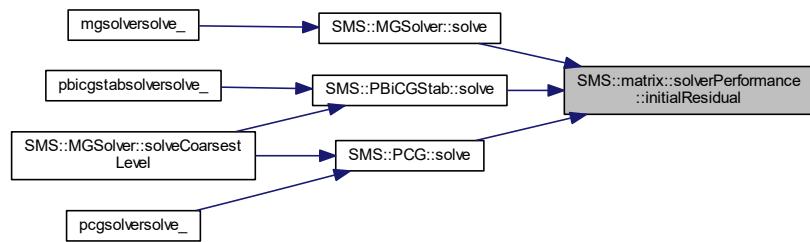
6.23.2.7 finalResidual() [2/2]

```
scalar& SMS::matrix::solverPerformance::finalResidual ( ) [inline]
```

6.23.2.8 initialResidual() [1/2]

```
scalar SMS::matrix::solverPerformance::initialResidual ( ) const [inline]
```

Here is the caller graph for this function:



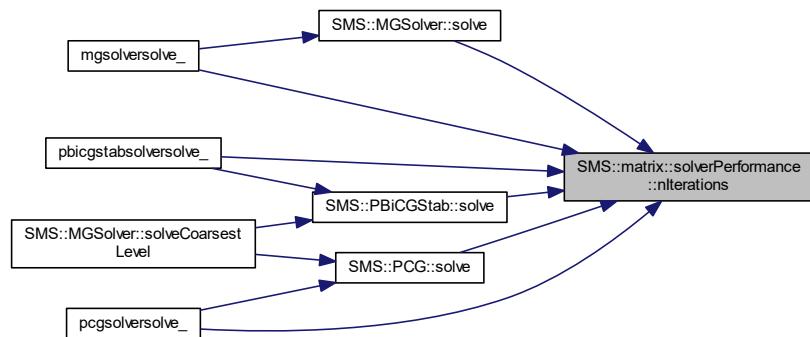
6.23.2.9 initialResidual() [2/2]

```
scalar& SMS::matrix::solverPerformance::initialResidual ( ) [inline]
```

6.23.2.10 nIterations() [1/2]

```
label SMS::matrix::solverPerformance::nIterations ( ) const [inline]
```

Here is the caller graph for this function:



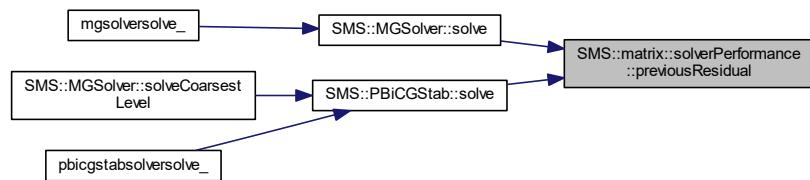
6.23.2.11 nIterations() [2/2]

```
label& SMS::matrix::solverPerformance::nIterations ( ) [inline]
```

6.23.2.12 previousResidual() [1/2]

```
scalar SMS::matrix::solverPerformance::previousResidual ( ) const [inline]
```

Here is the caller graph for this function:



6.23.2.13 previousResidual() [2/2]

```
scalar& SMS::matrix::solverPerformance::previousResidual ( ) [inline]
```

6.23.2.14 singular()

```
bool SMS::matrix::solverPerformance::singular ( ) const [inline]
```

6.23.3 Member Data Documentation

6.23.3.1 converged_

```
bool SMS::matrix::solverPerformance::converged_ [private]
```

6.23.3.2 finalResidual_

```
scalar SMS::matrix::solverPerformance::finalResidual_ [private]
```

6.23.3.3 initialResidual_

```
scalar SMS::matrix::solverPerformance::initialResidual_ [private]
```

6.23.3.4 nIterations_

```
label SMS::matrix::solverPerformance::noIterations_ [private]
```

6.23.3.5 previousResidual_

```
scalar SMS::matrix::solverPerformance::previousResidual_ [private]
```

6.23.3.6 singular_

```
bool SMS::matrix::solverPerformance::singular_ [private]
```

The documentation for this class was generated from the following files:

- SunwayMatrixSolver/src/matrix/[smsMatrix.hpp](#)
- SunwayMatrixSolver/src/matrix/[smsMatrixTest.cpp](#)

6.24 Timer Class Reference

```
#include <Timers.hpp>
```

Public Member Functions

- [~Timer \(\)](#)

Static Public Member Functions

- static void `startTimer` (std::string in)
- static void `endTimer` (std::string in)
- static void `printTimer` ()
- static void `printTimer` (std::string in)
- static void `openTimer` ()
- static void `closeTimer` ()
- template<typename DTYPE>
 static void `Vector_Variance` (DTYPE *vector, DTYPE *vector_compare, int size)
- static void `maxTimeSum` ()

Static Public Attributes

- static int `_OpenTimer` = 1
- static std::map< std::string, double > `timeSum`
- static std::map< std::string, double > `timeStart`
- static std::map< std::string, int > `count`

6.24.1 Constructor & Destructor Documentation

6.24.1.1 `~Timer()`

```
Timer::~Timer ( )
```

6.24.2 Member Function Documentation

6.24.2.1 `closeTimer()`

```
void Timer::closeTimer ( ) [static]
```

6.24.2.2 endTimer()

```
void Timer::endTimer (
    std::string in ) [static]
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.24.2.3 maxTimeSum()

```
void Timer::maxTimeSum ( ) [static]
```

6.24.2.4 openTimer()

```
void Timer::openTimer ( ) [static]
```

6.24.2.5 printTimer() [1/2]

```
void Timer::printTimer ( ) [static]
```

6.24.2.6 printTimer() [2/2]

```
void Timer::printTimer (
    std::string in ) [static]
```

6.24.2.7 startTimer()

```
void Timer::startTimer (
    std::string in ) [static]
```

Here is the call graph for this function:



Here is the caller graph for this function:



6.24.2.8 Vector_Variance()

```
template<typename DTYP>
static void Timer::Vector_Variance (
    DTYP * vector,
    DTYP * vector_compare,
    int size ) [inline], [static]
```

6.24.3 Member Data Documentation

6.24.3.1 _OpenTimer

```
int Timer::_OpenTimer = 1 [static]
```

6.24.3.2 count

```
std::map< std::string, int > Timer::count [static]
```

6.24.3.3 timeStart

```
std::map< std::string, double > Timer::timeStart [static]
```

6.24.3.4 timeSum

```
std::map< std::string, double > Timer::timeSum [static]
```

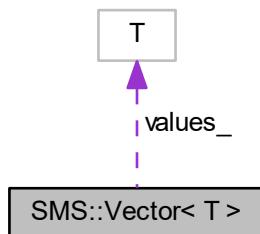
The documentation for this class was generated from the following files:

- SunwayMatrixSolver/tools/timers/[Timers.hpp](#)
- SunwayMatrixSolver/tools/timers/[Timers.cpp](#)

6.25 SMS::Vector< T > Class Template Reference

```
#include <smsVector.hpp>
```

Collaboration diagram for SMS::Vector< T >:



Public Member Functions

- `Vector ()`
- `Vector (const label length)`
- `Vector (const Vector< T > &v)`
- `Vector (const T *val, const label &length)`
- `Vector (const T *val, const label &length, const bool reUse)`
- `Vector (const label length, const T value)`
- `virtual ~Vector ()`
- `T & operator[] (const label i)`
- `const T operator[] (const label i) const`
- `Vector & operator= (const Vector< T > &v)`
- `Vector & operator= (const T &a)`
- `Vector & operator+= (const Vector< T > &v)`
- `Vector & operator-= (const Vector< T > &v)`
- `Vector & operator+= (const T &a)`
- `Vector & operator-= (const T &a)`
- `Vector & operator *= (const T &a)`
- `const Vector operator+ (const Vector< T > &v) const`
- `const Vector operator- (const Vector< T > &v) const`
- `const Vector operator * (const T &a) const`
- `label size () const`
- `void SET_size (const label newSize)`
- `T * values () const`
- `T Sum () const`
- `T SumMag () const`
- `T SumSqr () const`
- `scalar SumSqrt () const`
- `T * begin () const`
- `T * end () const`
- `void SET_zero ()`

Private Attributes

- `label length_`
- `T * values_`

6.25.1 Constructor & Destructor Documentation

6.25.1.1 `Vector()` [1/6]

```
template<typename T >
SMS::Vector< T >::Vector ( )
```

6.25.1.2 Vector() [2/6]

```
template<typename T>
SMS::Vector< T >::Vector (
    const label length )
```

6.25.1.3 Vector() [3/6]

```
template<typename T>
SMS::Vector< T >::Vector (
    const Vector< T > & v )
```

6.25.1.4 Vector() [4/6]

```
template<typename T>
SMS::Vector< T >::Vector (
    const T * val,
    const label & length )
```

6.25.1.5 Vector() [5/6]

```
template<typename T>
SMS::Vector< T >::Vector (
    const T * val,
    const label & length,
    const bool reUse )
```

6.25.1.6 Vector() [6/6]

```
template<typename T>
SMS::Vector< T >::Vector (
    const label length,
    const T value )
```

6.25.1.7 ~Vector()

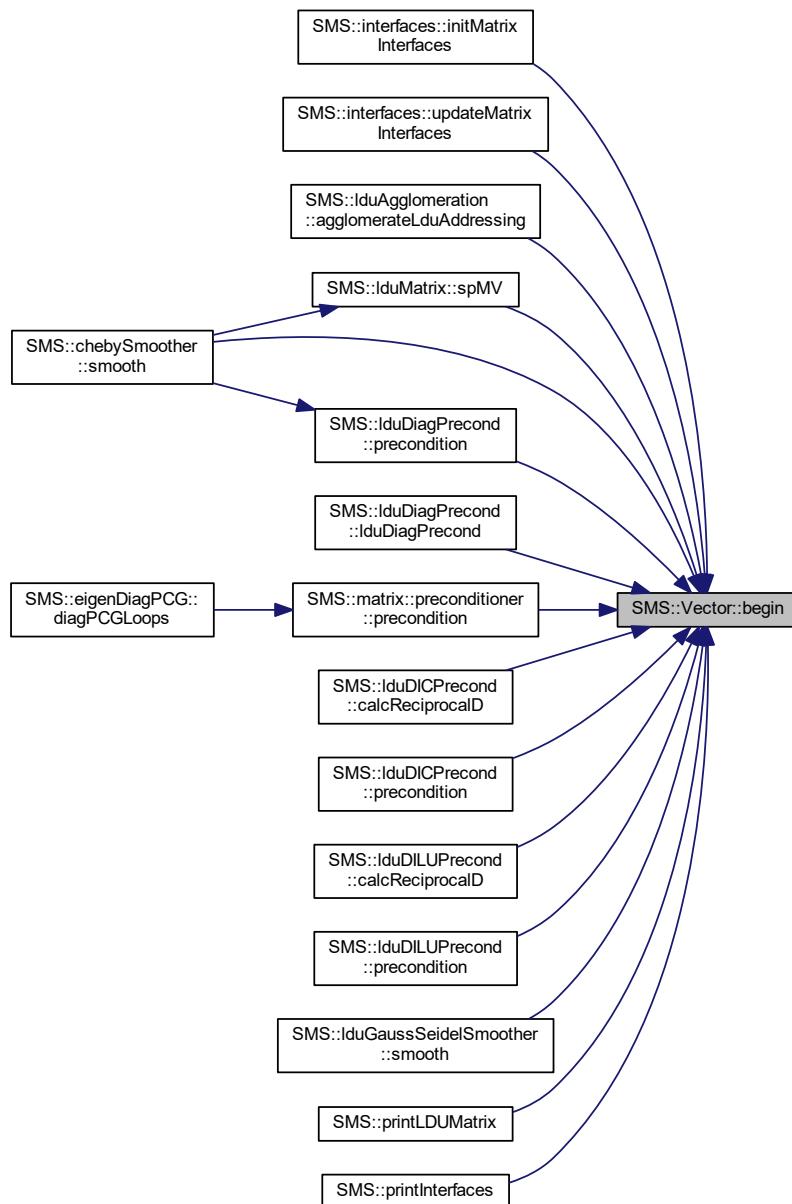
```
template<typename T>
SMS::Vector< T >::~Vector () [virtual]
```

6.25.2 Member Function Documentation

6.25.2.1 begin()

```
template<typename T>
T* SMS::Vector< T >::begin ( ) const [inline]
```

Here is the caller graph for this function:



6.25.2.2 end()

```
template<typename T>
T* SMS::Vector< T >::end ( ) const [inline]
```

6.25.2.3 operator *()

```
template<typename T>
const Vector< T > SMS::Vector< T >::operator * (
    const T & a ) const
```

6.25.2.4 operator *=()

```
template<typename T>
Vector< T > & SMS::Vector< T >::operator *= (
    const T & a )
```

6.25.2.5 operator+()

```
template<typename T>
const Vector< T > SMS::Vector< T >::operator+ (
    const Vector< T > & v ) const
```

6.25.2.6 operator+=() [1/2]

```
template<typename T>
Vector< T > & SMS::Vector< T >::operator+= (
    const Vector< T > & v )
```

6.25.2.7 operator+=() [2/2]

```
template<typename T>
Vector< T > & SMS::Vector< T >::operator+= (
    const T & a )
```

6.25.2.8 operator-()

```
template<typename T>
const Vector< T > & SMS::Vector< T >::operator- (
    const Vector< T > & v ) const
```

6.25.2.9 operator-() [1/2]

```
template<typename T>
Vector< T > & SMS::Vector< T >::operator- = (
    const Vector< T > & v )
```

6.25.2.10 operator-() [2/2]

```
template<typename T>
Vector< T > & SMS::Vector< T >::operator- = (
    const T & a )
```

6.25.2.11 operator=() [1/2]

```
template<typename T>
Vector< T > & SMS::Vector< T >::operator= (
    const Vector< T > & v )
```

6.25.2.12 operator=() [2/2]

```
template<typename T>
Vector< T > & SMS::Vector< T >::operator= (
    const T & a )
```

6.25.2.13 operator[]() [1/2]

```
template<typename T>
T& SMS::Vector< T >::operator[] (
    const label i ) [inline]
```

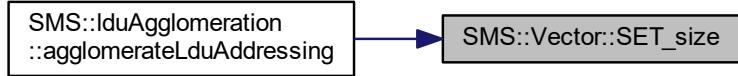
6.25.2.14 operator[]() [2/2]

```
template<typename T>
const T SMS::Vector< T >::operator[ ] (
    const label i ) const [inline]
```

6.25.2.15 SET_size()

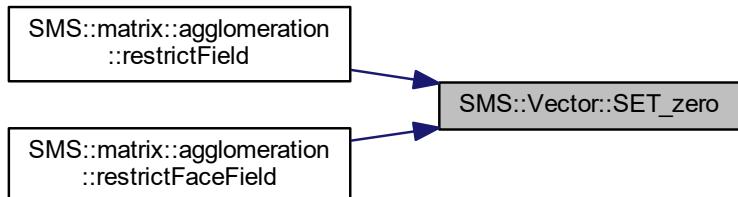
```
template<typename T >
void SMS::Vector< T >::SET_size (
    const label newSize )
```

Here is the caller graph for this function:

**6.25.2.16 SET_zero()**

```
template<typename T >
void SMS::Vector< T >::SET_zero ( )
```

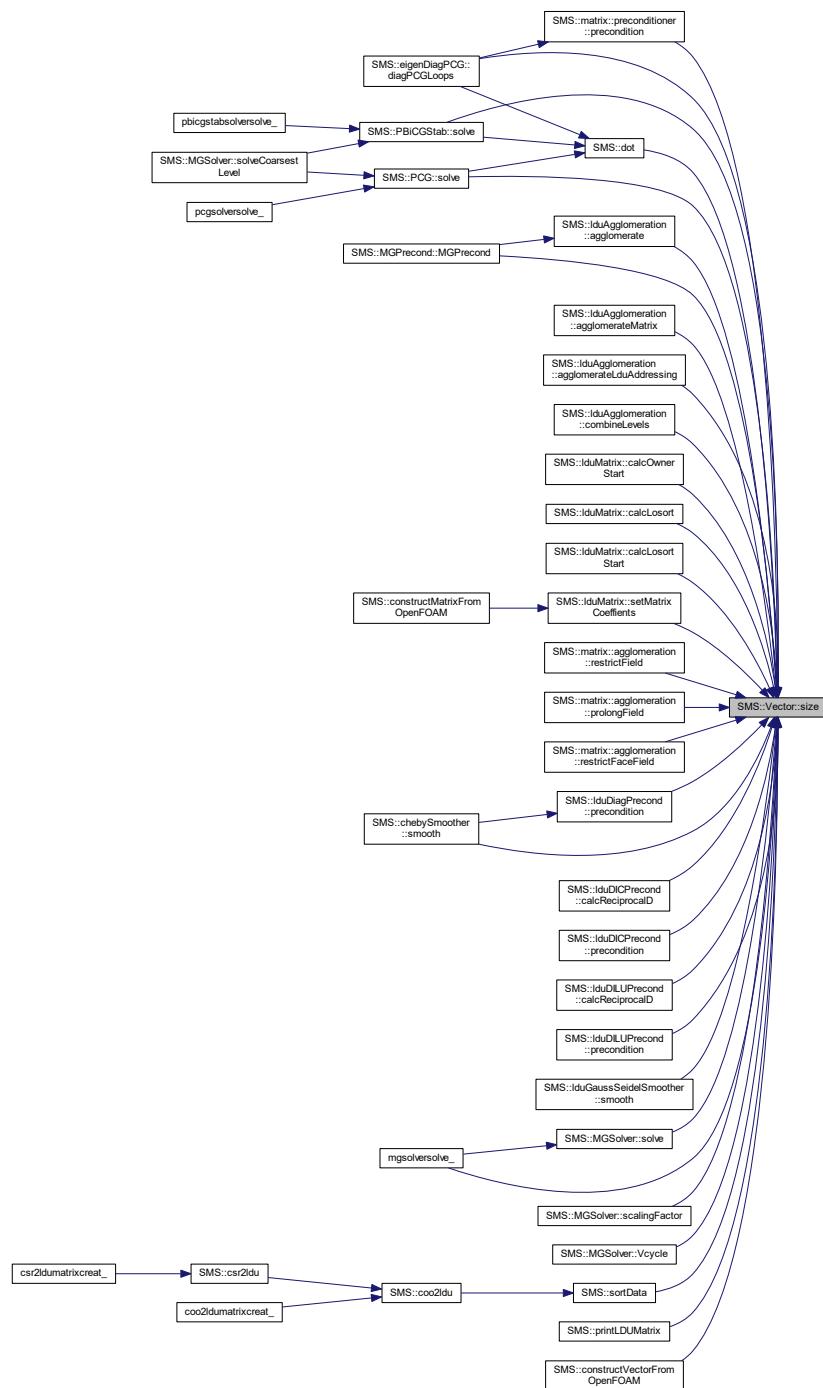
Here is the caller graph for this function:



6.25.2.17 size()

```
template<typename T>
label SMS::Vector< T >::size ( ) const [inline]
```

Here is the caller graph for this function:



6.25.2.18 Sum()

```
template<typename T >
T SMS::Vector< T >::Sum( ) const
```

6.25.2.19 SumMag()

```
template<typename T >
T SMS::Vector< T >::SumMag( ) const
```

6.25.2.20 SumSqr()

```
template<typename T >
T SMS::Vector< T >::SumSqr( ) const
```

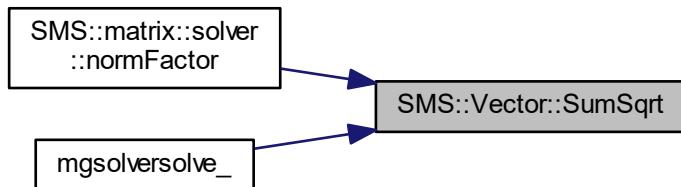
Here is the caller graph for this function:



6.25.2.21 SumSqrt()

```
template<typename T >
scalar SMS::Vector< T >::SumSqrt( ) const
```

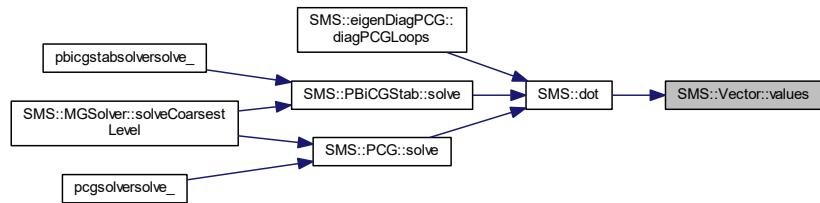
Here is the caller graph for this function:



6.25.2.22 values()

```
template<typename T>
T* SMS::Vector< T >::values ( ) const [inline]
```

Here is the caller graph for this function:



6.25.3 Member Data Documentation

6.25.3.1 length_

```
template<typename T>
label SMS::Vector< T >::length_ [private]
```

6.25.3.2 values_

```
template<typename T>
T* SMS::Vector< T >::values_ [private]
```

The documentation for this class was generated from the following file:

- SunwayMatrixSolver/src/base/vectors/[smsVector.hpp](#)

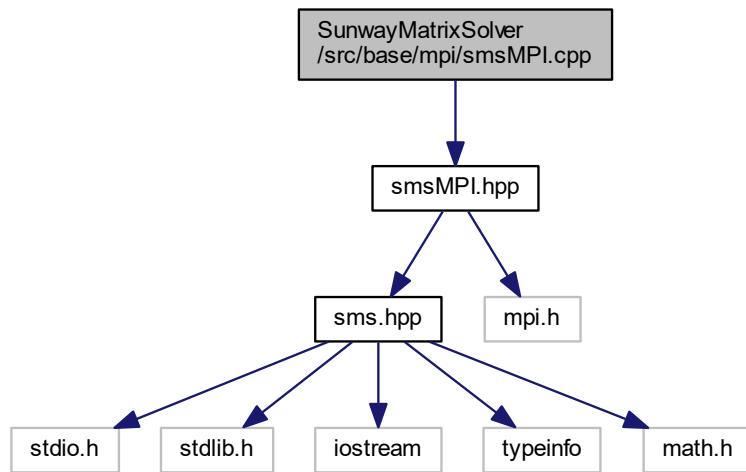
Chapter 7

File Documentation

7.1 SunwayMatrixSolver/src/base/mpi/smsMPI.cpp File Reference

```
#include "smsMPI.hpp"
```

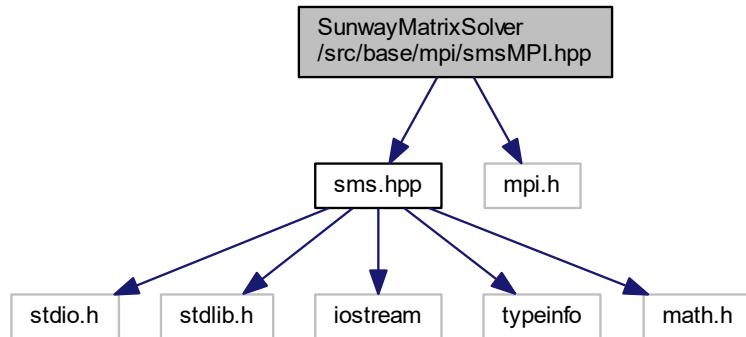
Include dependency graph for smsMPI.cpp:



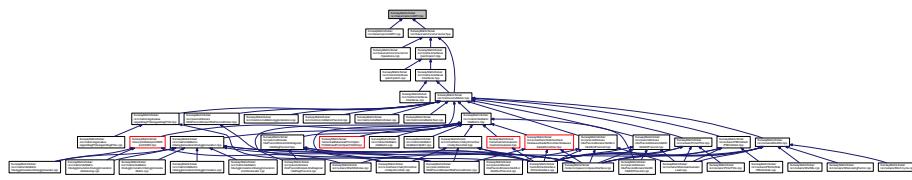
7.2 SunwayMatrixSolver/src/base/mpi/smsMPI.hpp File Reference

brief information to be added

```
#include "sms.hpp"
#include <mpi.h>
Include dependency graph for smsMPI.hpp:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::smsMPI](#)

Namespaces

- [SMS](#)

Macros

- `#define MYID smsMPI::myProcNo()`
- `#define PARRUN smsMPI::parRun()`
- `#define MPI_LABEL smsMPI::smsLabel()`
- `#define MPI_SCALAR smsMPI::smsScalar()`
- `#define NPROCS smsMPI::nProcs()`

Functions

- template<typename T >
`void SMS::reduce (T &v)`

7.2.1 Detailed Description

brief information to be added

Detailed information to be added

Author

Hanfeng GU

Version

1.0

Date

2019-01-30

7.2.2 Macro Definition Documentation

7.2.2.1 MPI_LABEL

```
#define MPI_LABEL smsMPI::smsLabel()
```

7.2.2.2 MPI_SCALAR

```
#define MPI_SCALAR smsMPI::smsScalar()
```

7.2.2.3 MYID

```
#define MYID smsMPI::myProcNo()
```

7.2.2.4 NPROCS

```
#define NPROCS smsMPI::nProcs()
```

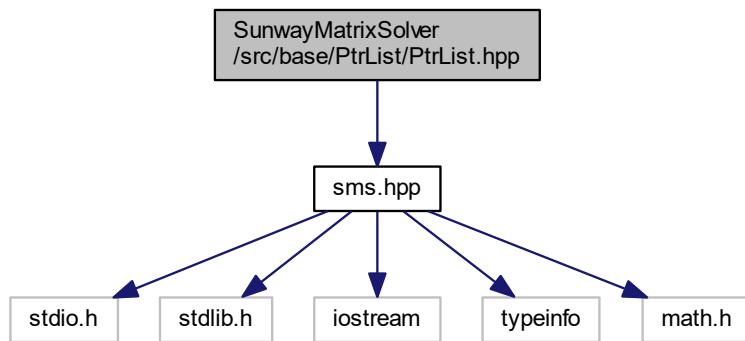
7.2.2.5 PARRUN

```
#define PARRUN smsMPI::parRun()
```

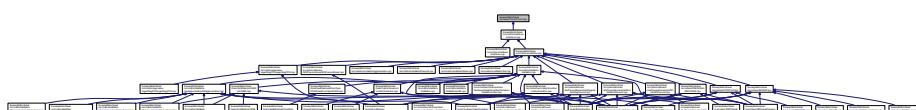
7.3 SunwayMatrixSolver/src/base/PtrList/PtrList.hpp File Reference

brief information to be added

```
#include "sms.hpp"
Include dependency graph for PtrList.hpp:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::PtrList< T >](#)

Namespaces

- [SMS](#)

7.3.1 Detailed Description

brief information to be added

Detailed information to be added

Author

Hanfeng GU

Version

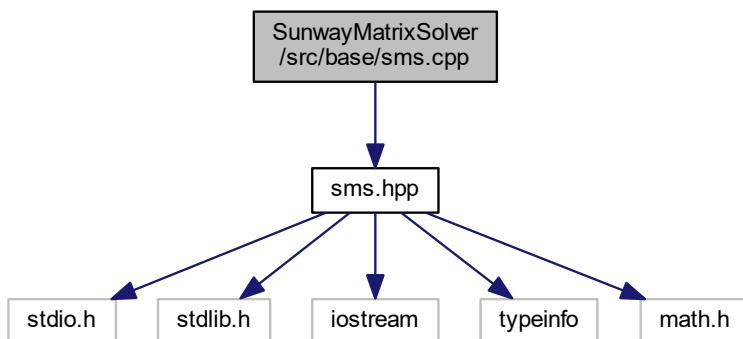
1.0

Date

2019-01-30

7.4 SunwayMatrixSolver/src/base/sms.cpp File Reference

```
#include "sms.hpp"
Include dependency graph for sms.cpp:
```



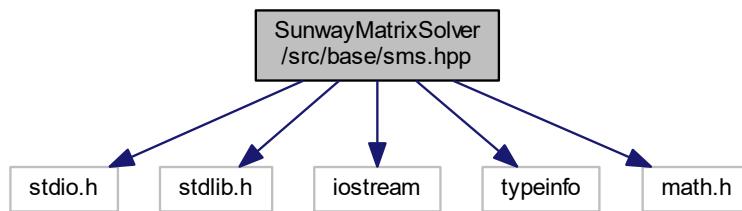
Namespaces

- [SMS](#)

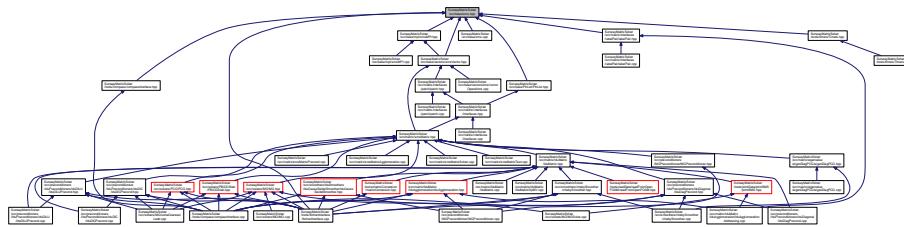
7.5 SunwayMatrixSolver/src/base/sms.hpp File Reference

brief information to be added

```
#include <stdio.h>
#include <stdlib.h>
#include <iostream>
#include <typeinfo>
#include <math.h>
Include dependency graph for sms.hpp:
```



This graph shows which files directly or indirectly include this file:



Namespaces

- [SMS](#)

Macros

- `#define COUT std::cout`
- `#define ENDL std::endl`
- `#define EXIT exit(0)`
- `#define ERROR_EXIT exit(1)`
- `#define forAll(i, length) for(int i=0; i<length; i++)`
- `#define SMALL (1.0e-37)`
- `#define VSMALL (1.0e-37)`
- `#define GREAT (1.0e+37)`
- `#define VGREAT (1.0e+37)`
- `#define CHECK_POINTER(ptr)`
- `#define DELETE_POINTER(ptr)`
- `#define DELETE_OBJECT_POINTER(ptr)`
- `#define ALLOCATE_POINTER(ptr, oldObj, T)`

Functions

- `unsigned int SMS::sleep (const unsigned int)`
- `template<typename T >`
`T SMS::max (T a, T b)`
- `template<typename T >`
`T SMS::min (T a, T b)`
- `template<typename T >`
`T SMS::mag (T v)`

Variables

- `ERROR_EXIT`

7.5.1 Detailed Description

brief information to be added

Detailed information to be added

Author

Hanfeng GU

Version

1.0

Date

2019-01-30

7.5.2 Macro Definition Documentation

7.5.2.1 ALLOCATE_POINTER

```
#define ALLOCATE_POINTER(  
    ptr,  
    oldObj,  
    T )  
  
Value:  
{ \  
    DELETE_OBJECT_POINTER(ptr) \  
    ptr = new T(oldObj.size()); \  
    T& newObj = *ptr; \  
    newObj = oldObj; \  
}
```

7.5.2.2 CHECK_POINTER

```
#define CHECK_POINTER(
    ptr )
```

Value:

```
if(!ptr) \
{ \
    COUT << "ERROR in " << __FILE__ << " " << __LINE__ \
    << ":" << #ptr << " is NULL!" << endl; \
    ERROR_EXIT; \
}
```

7.5.2.3 COUT

```
#define COUT std::cout
```

7.5.2.4 DELETE_OBJECT_POINTER

```
#define DELETE_OBJECT_POINTER(
    ptr )
```

Value:

```
if(ptr) \
{ \
    delete ptr; \
    ptr = NULL; \
}
```

7.5.2.5 DELETE_POINTER

```
#define DELETE_POINTER(
    ptr )
```

Value:

```
if(ptr) \
{ \
    delete [] ptr; \
    ptr = NULL; \
}
```

7.5.2.6 ENDL

```
#define ENDL std::endl
```

7.5.2.7 ERROR_EXIT

```
#define ERROR_EXIT exit(1)
```

7.5.2.8 EXIT

```
#define EXIT exit(0)
```

7.5.2.9 forAll

```
#define forAll(  
    i,  
    length ) for(int i=0; i<length; i++)
```

7.5.2.10 GREAT

```
#define GREAT (1.0e+37)
```

7.5.2.11 SMALL

```
#define SMALL (1.0e-37)
```

7.5.2.12 VGREAT

```
#define VGREAT (1.0e+37)
```

7.5.2.13 VSMALL

```
#define VSMALL (1.0e-37)
```

7.5.3 Variable Documentation

7.5.3.1 ERROR_EXIT

ERROR_EXIT

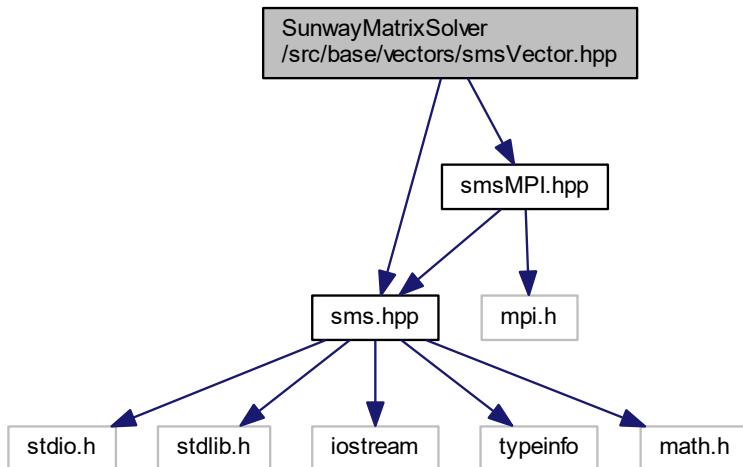
Initial value:

```
{
    COUT << "Error: label size is not defined." << endl
```

7.6 SunwayMatrixSolver/src/base/vectors/smsVector.hpp File Reference

brief information to be added

```
#include "sms.hpp"
#include "smsMPI.hpp"
Include dependency graph for smsVector.hpp:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::Vector< T >](#)

Namespaces

- [SMS](#)

Typedefs

- `typedef Vector< label > SMS::labelField`
- `typedef Vector< scalar > SMS::scalarField`

Functions

- `scalar SMS::dot (const scalarField &v1, const scalarField &v2)`

7.6.1 Detailed Description

brief information to be added

Detailed information to be added

Author

Hanfeng GU

Version

1.0

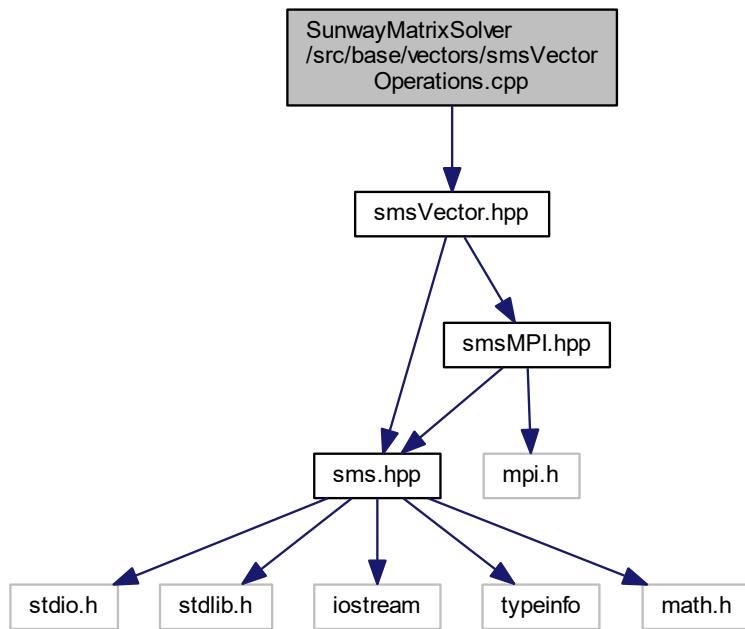
Date

2019-01-30

7.7 SunwayMatrixSolver/src/base/vectors/smsVectorOperations.cpp File Reference

```
#include "smsVector.hpp"
```

Include dependency graph for smsVectorOperations.cpp:



Namespaces

- [SMS](#)

Functions

- scalar [SMS::dot](#) (const scalarField &v1, const scalarField &v2)

7.8 SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/csrAgglomeration.cpp File Reference

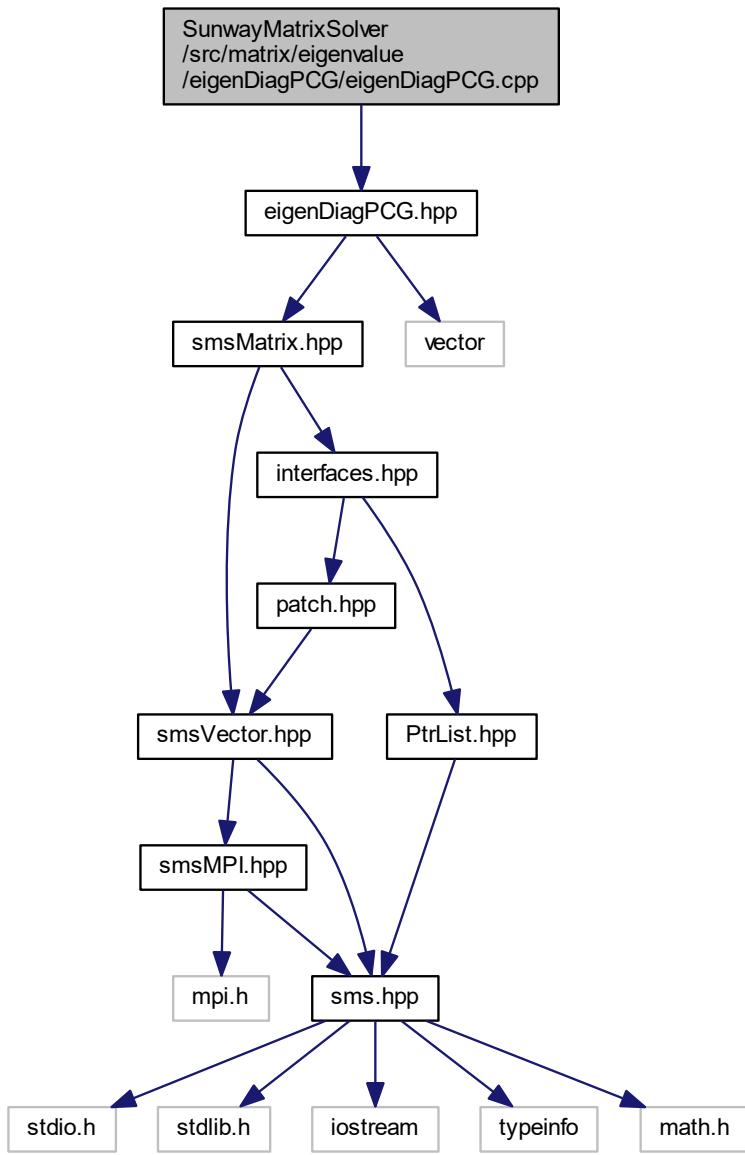
7.9 SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/csrAgglomeration.hpp File Reference

7.10 SunwayMatrixSolver/src/matrix/csrMatrix/csrMatrix.cpp File Reference

7.11 SunwayMatrixSolver/src/matrix/csrMatrix/csrMatrix.hpp File Reference

7.12 SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/eigenDiagPCG.cpp File Reference

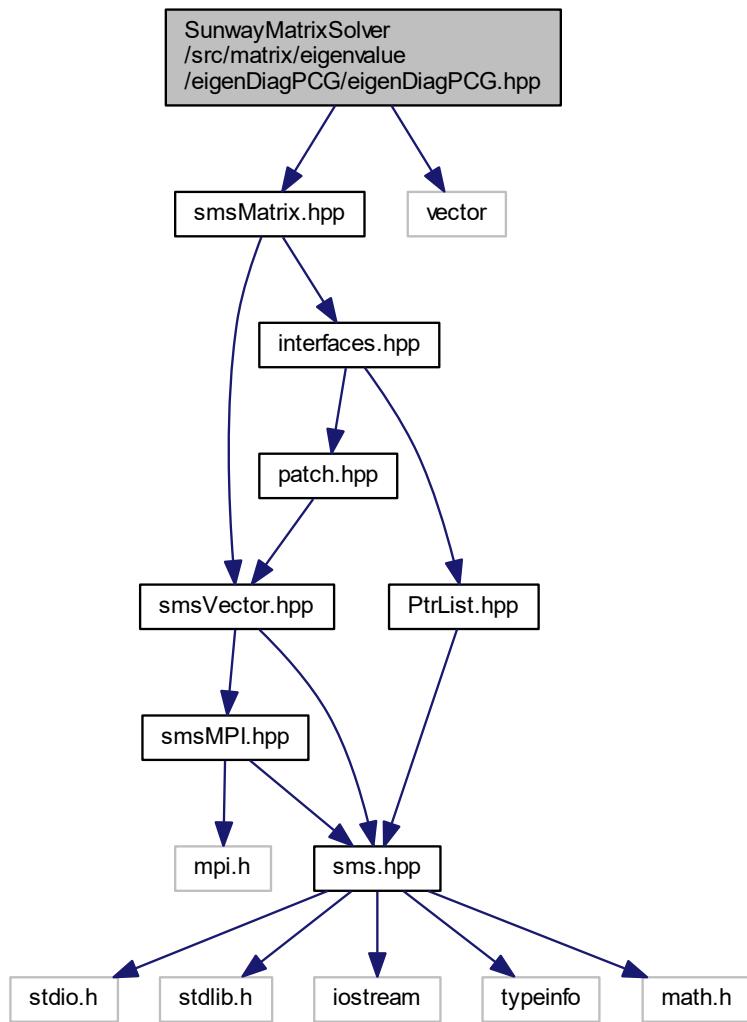
```
#include "eigenDiagPCG.hpp"
Include dependency graph for eigenDiagPCG.cpp:
```



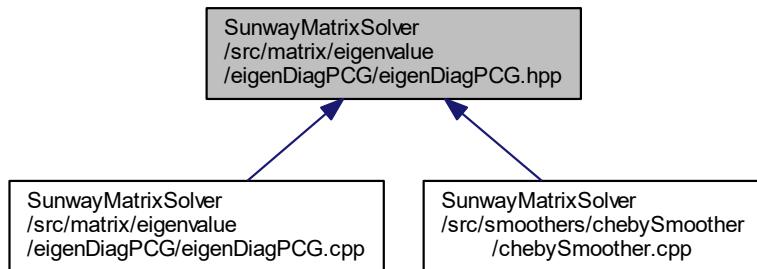
7.13 SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/eigenDiagPCG.hpp File Reference

```
#include "smsMatrix.hpp"
```

```
#include <vector>
Include dependency graph for eigenDiagPCG.hpp:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::eigenDiagPCG](#)

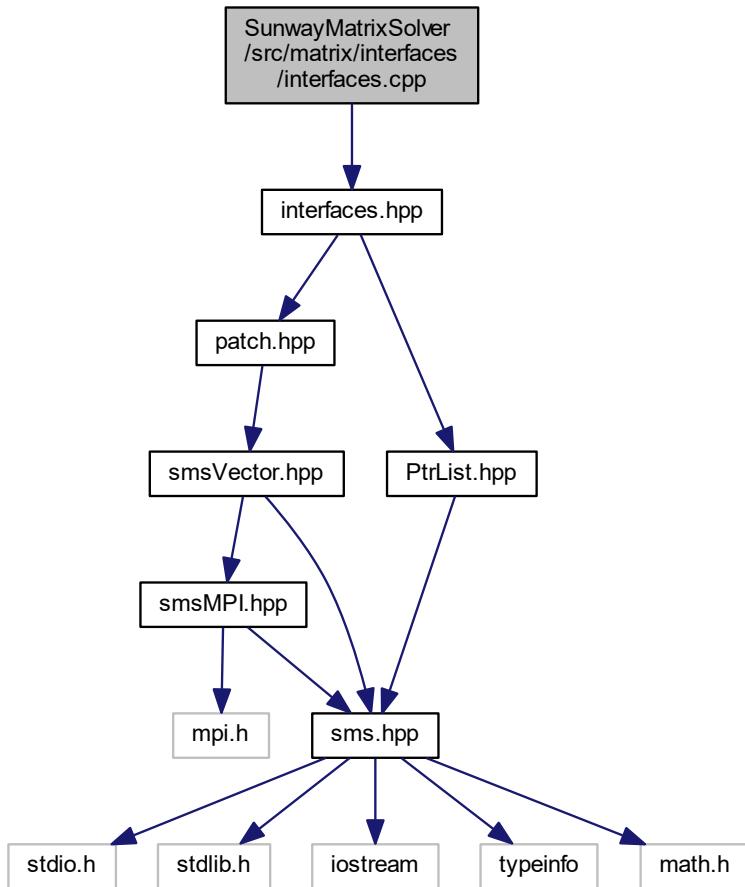
Namespaces

- [SMS](#)

7.14 SunwayMatrixSolver/src/matrix/interfaces/interfaces.cpp File Reference

```
#include "interfaces.hpp"
```

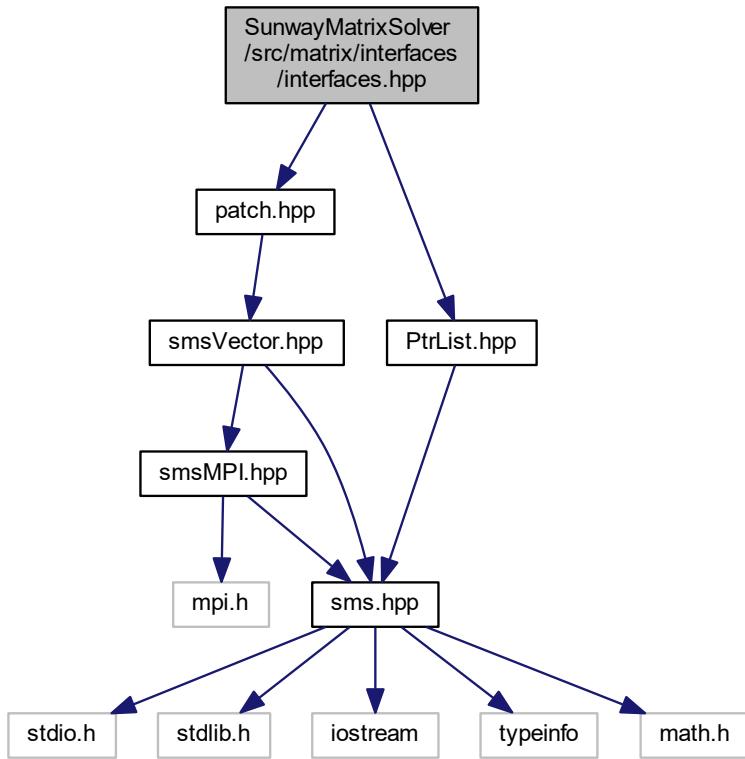
Include dependency graph for interfaces.cpp:



7.15 SunwayMatrixSolver/src/matrix/interfaces/interfaces.hpp File Reference

```
#include "patch.hpp"
#include "PtrList.hpp"
```

Include dependency graph for interfaces.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::interfaces](#)

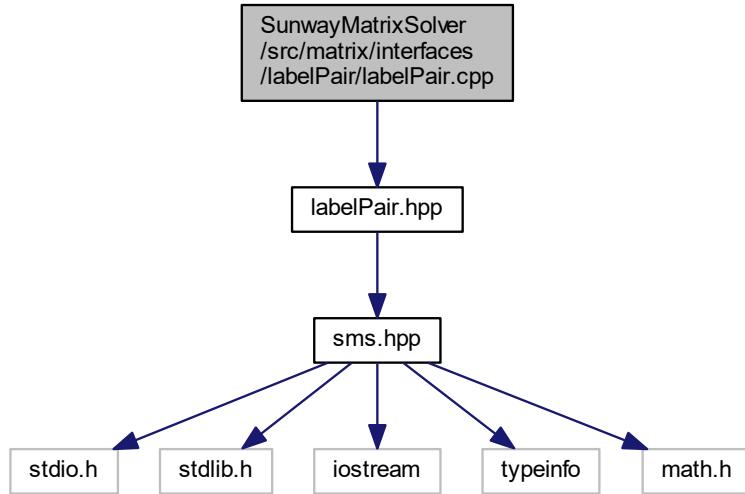
Namespaces

- [SMS](#)

7.16 SunwayMatrixSolver/src/matrix/interfaces/labelPair/labelPair.cpp File Reference

```
#include "labelPair.hpp"
```

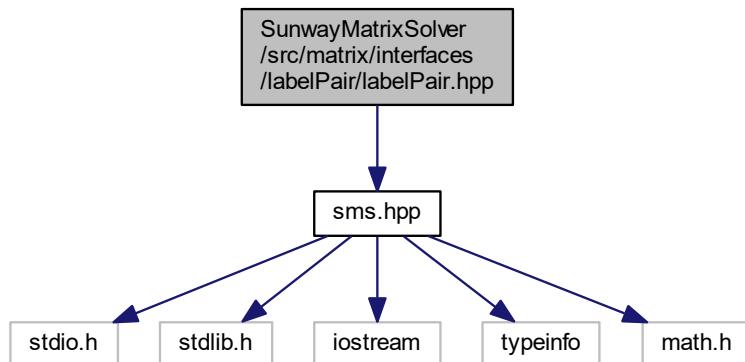
Include dependency graph for labelPair.cpp:



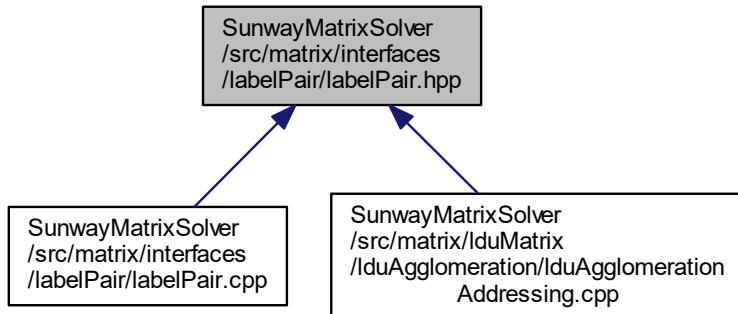
7.17 SunwayMatrixSolver/src/matrix/interfaces/labelPair/labelPair.hpp File Reference

```
#include "sms.hpp"
```

Include dependency graph for labelPair.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::labelPair](#)

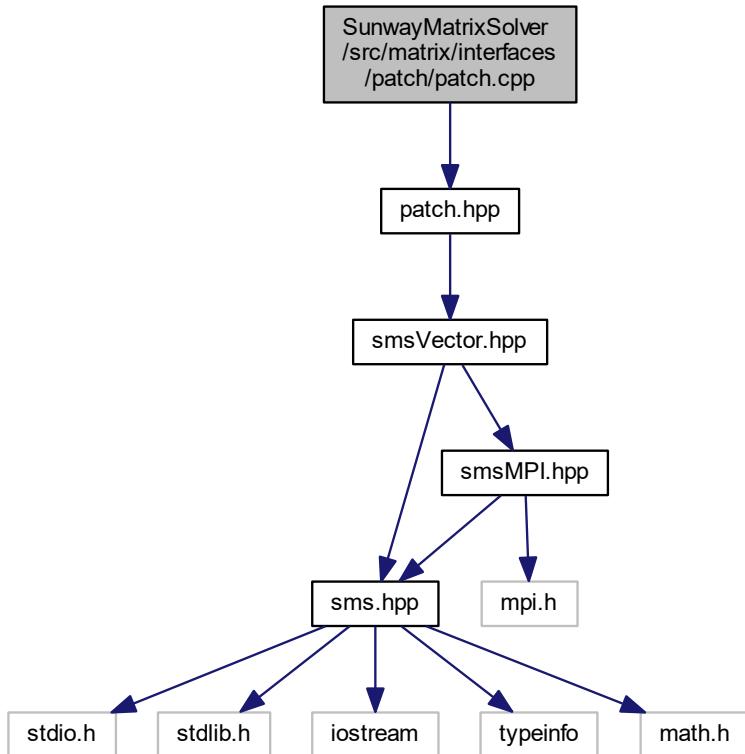
Namespaces

- [SMS](#)

7.18 SunwayMatrixSolver/src/matrix/interfaces/patch/patch.cpp File Reference

```
#include "patch.hpp"
```

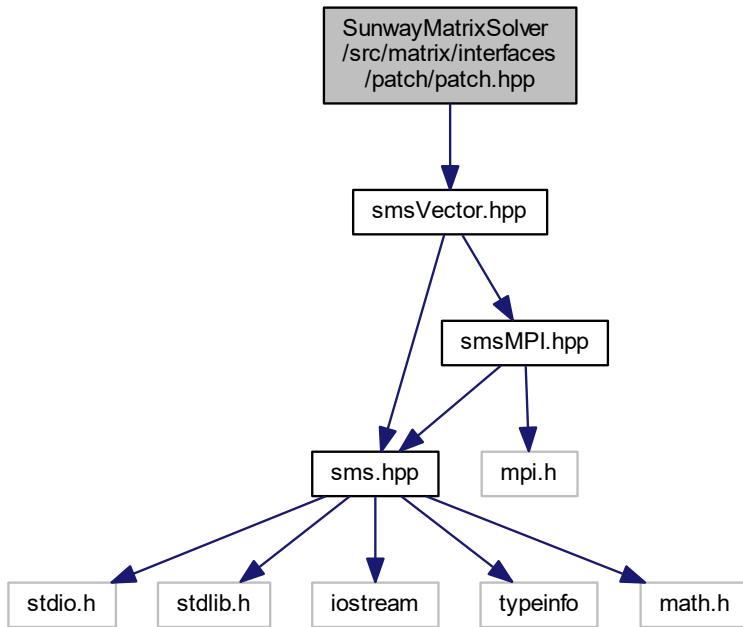
Include dependency graph for patch.cpp:



7.19 SunwayMatrixSolver/src/matrix/interfaces/patch/patch.hpp File Reference

```
#include "smsVector.hpp"
```

Include dependency graph for patch.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::patch](#)

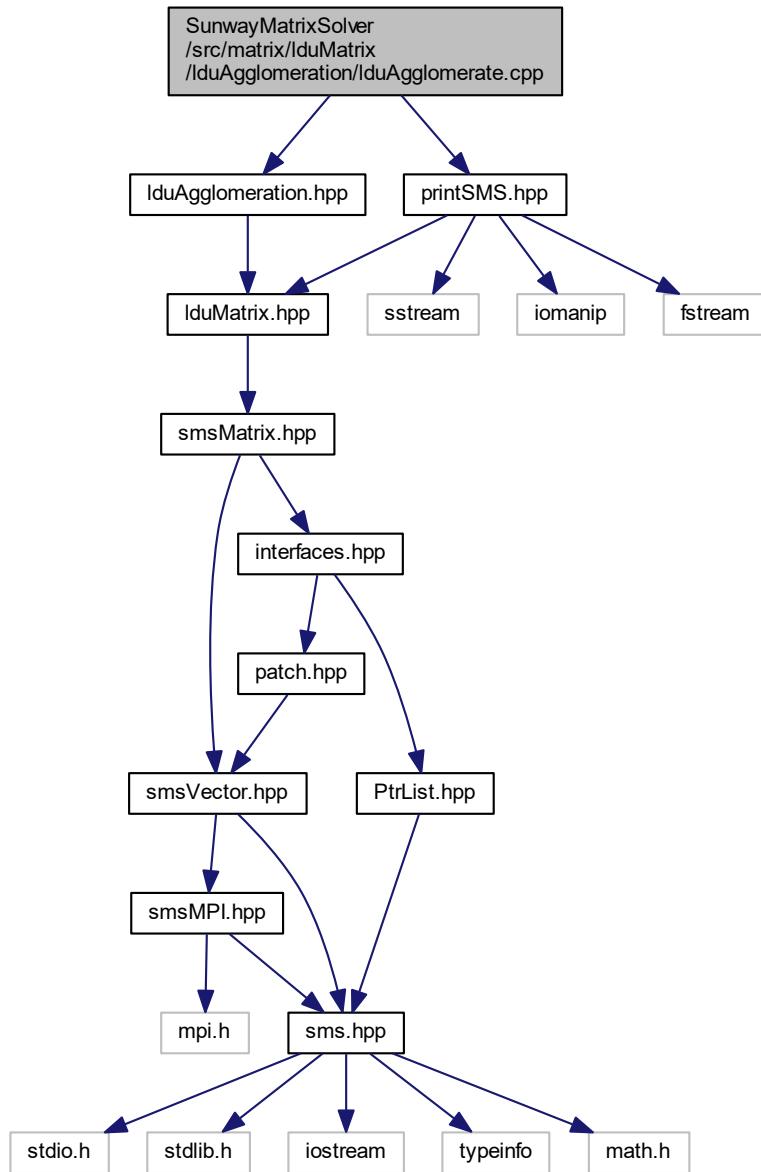
Namespaces

- [SMS](#)

7.20 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerate.cpp File Reference

```
#include "lduAgglomeration.hpp"
#include "printSMS.hpp"
```

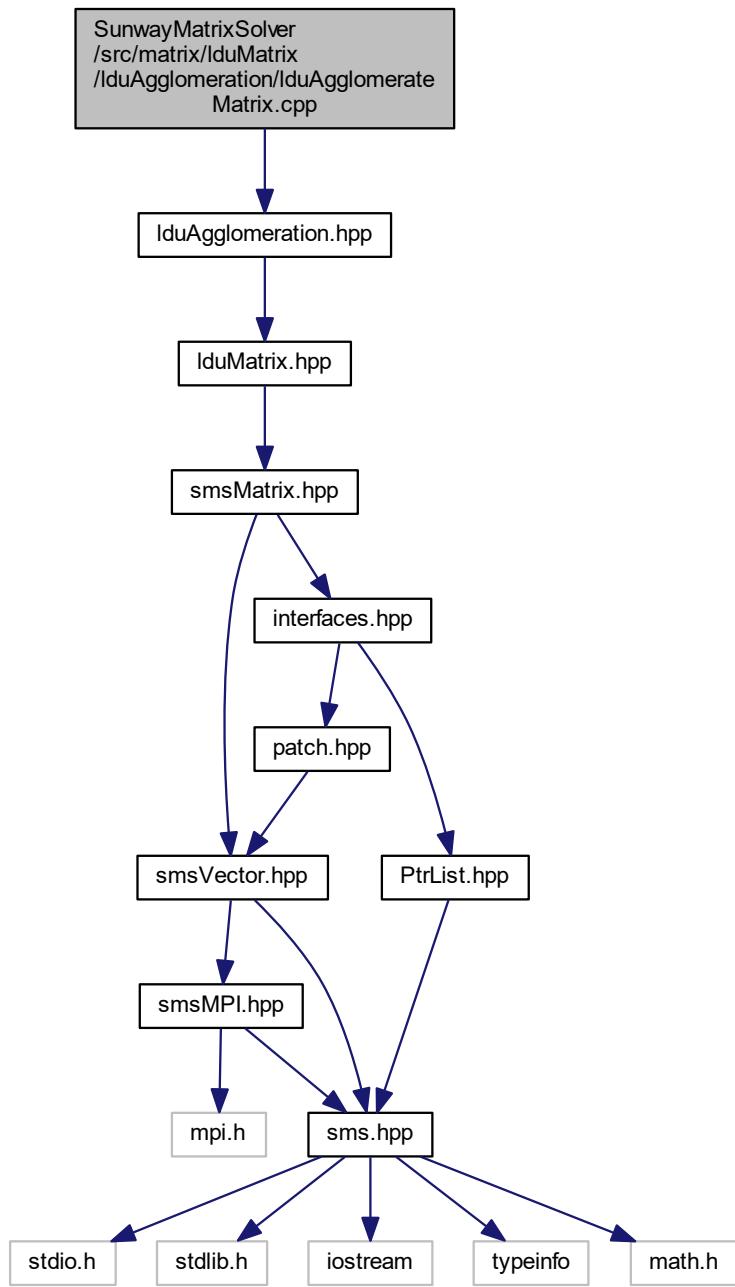
Include dependency graph for lduAgglomerate.cpp:



7.21 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerate.cpp File Reference

```
#include "lduAgglomeration.hpp"
```

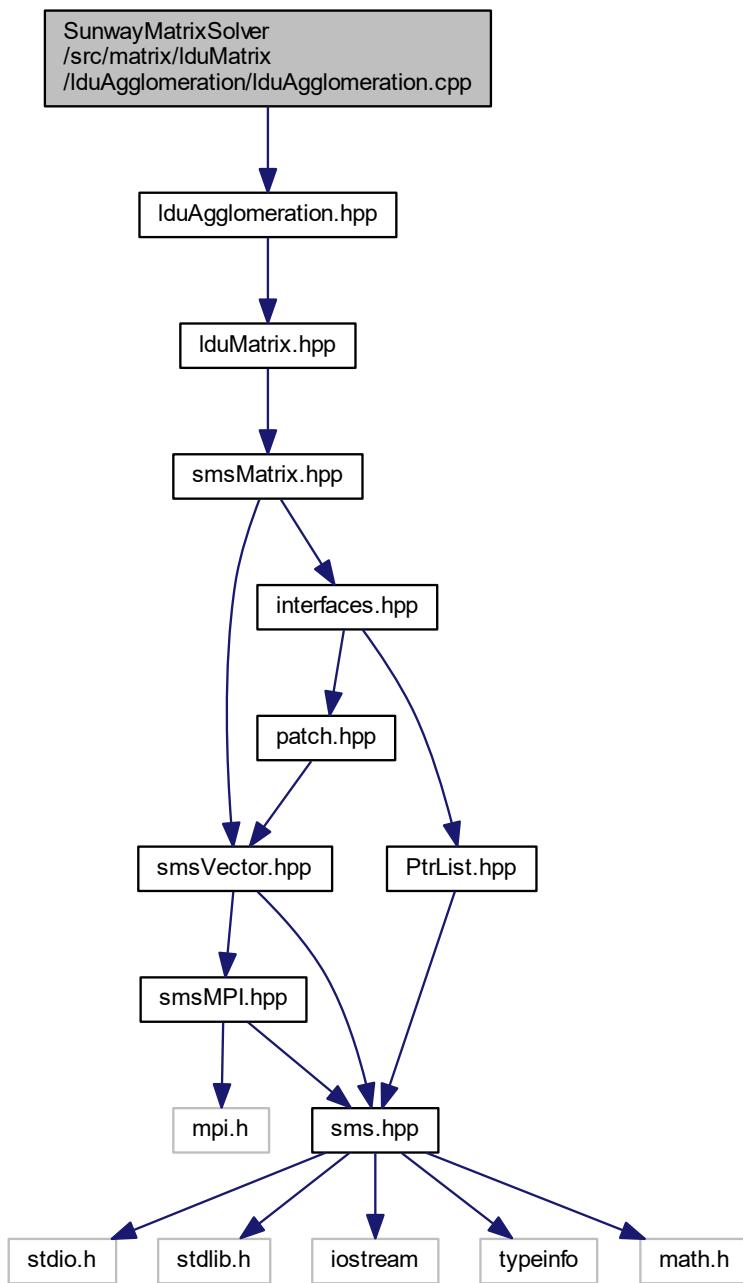
Include dependency graph for lduAgglomerateMatrix.cpp:



7.22 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomeration.cpp File Reference

```
#include "lduAgglomeration.hpp"
```

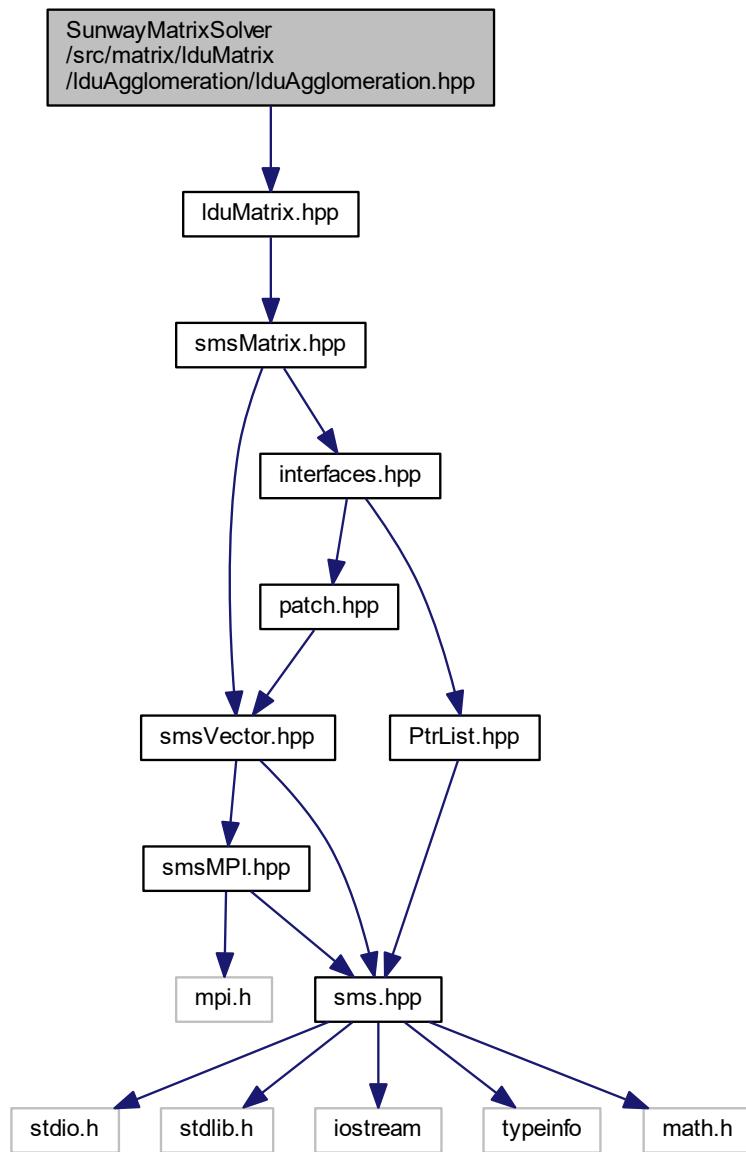
Include dependency graph for lduAgglomeration.cpp:



7.23 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomeration.hpp File Reference

```
#include "lduMatrix.hpp"
```

Include dependency graph for IduAgglomeration.hpp:



This graph shows which files directly or indirectly include this file:



Classes

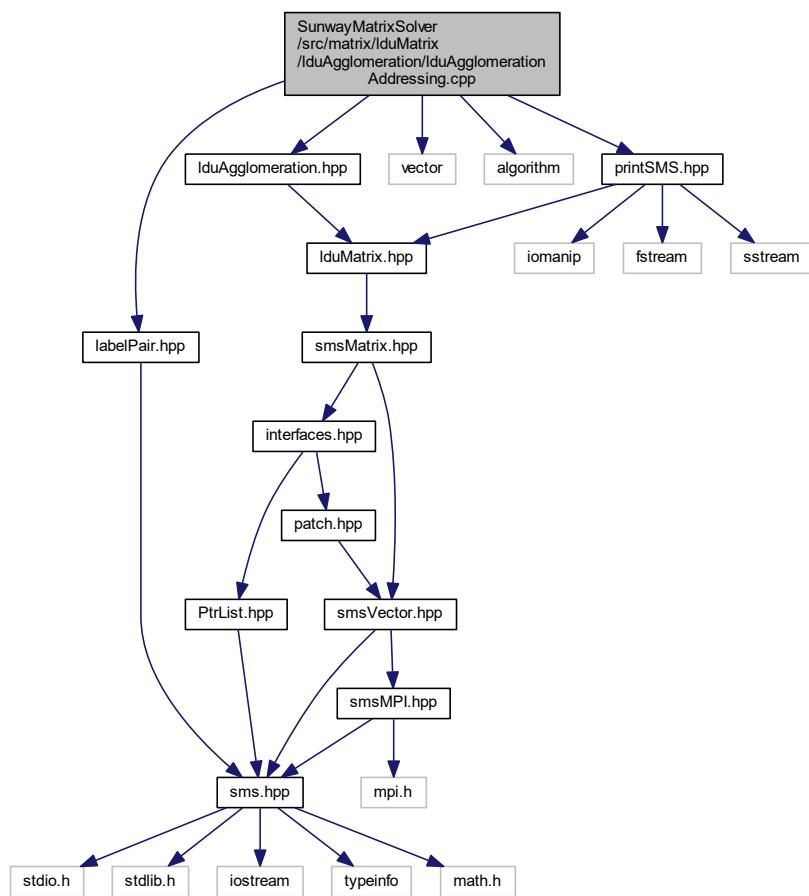
- class [SMS::IduAgglomeration](#)

Namespaces

- SMS

7.24 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerationAddressing.cpp File Reference

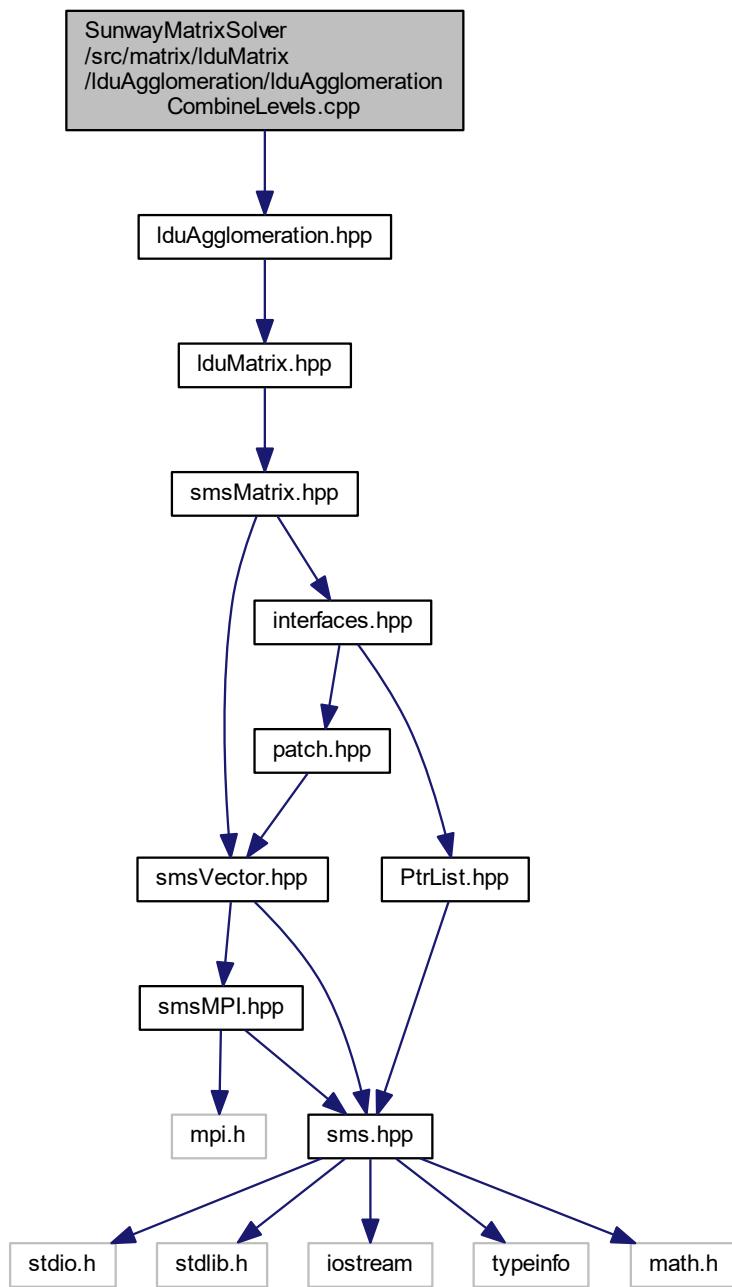
```
#include "lduAgglomeration.hpp"
#include "labelPair.hpp"
#include <vector>
#include <algorithm>
#include "printSMS.hpp"
Include dependency graph for lduAgglomerationAddressing.cpp:
```



7.25 SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration/lduAgglomerationCombineLevels.cpp File Reference

```
#include "lduAgglomeration.hpp"
```

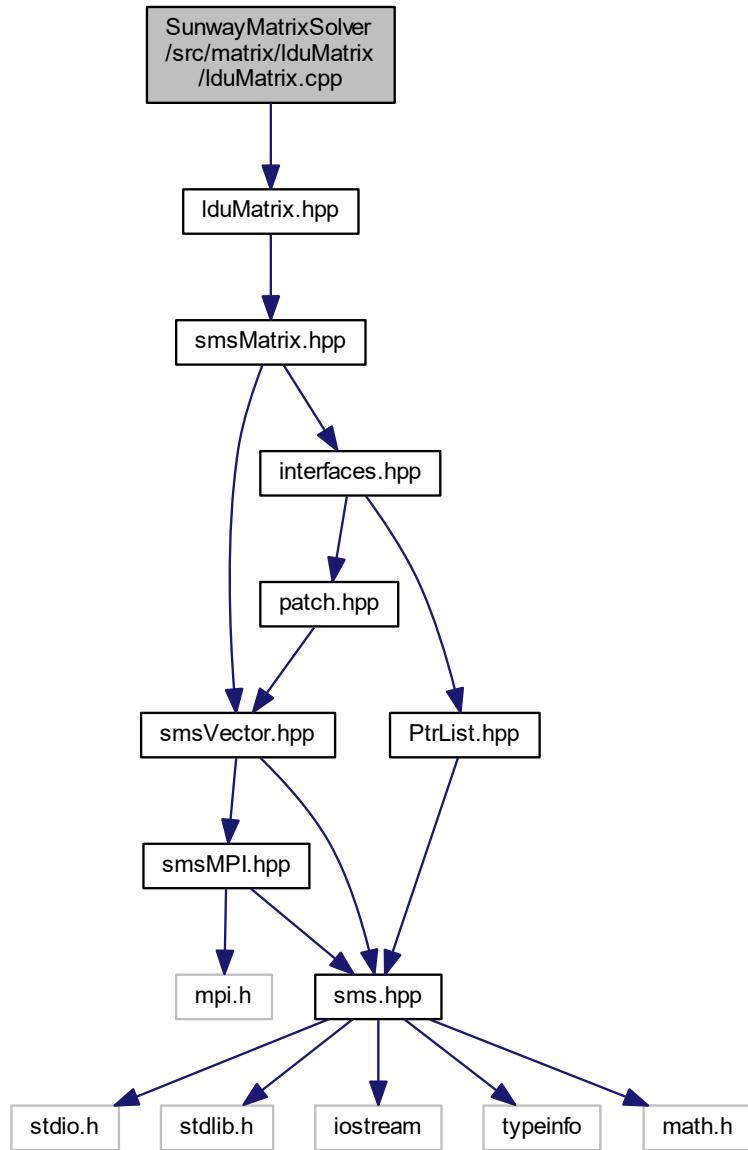
Include dependency graph for lDuAgglomerationCombineLevels.cpp:



7.26 SunwayMatrixSolver/src/matrix/lDuMatrix/lDuMatrix.cpp File Reference

```
#include "lDuMatrix.hpp"
```

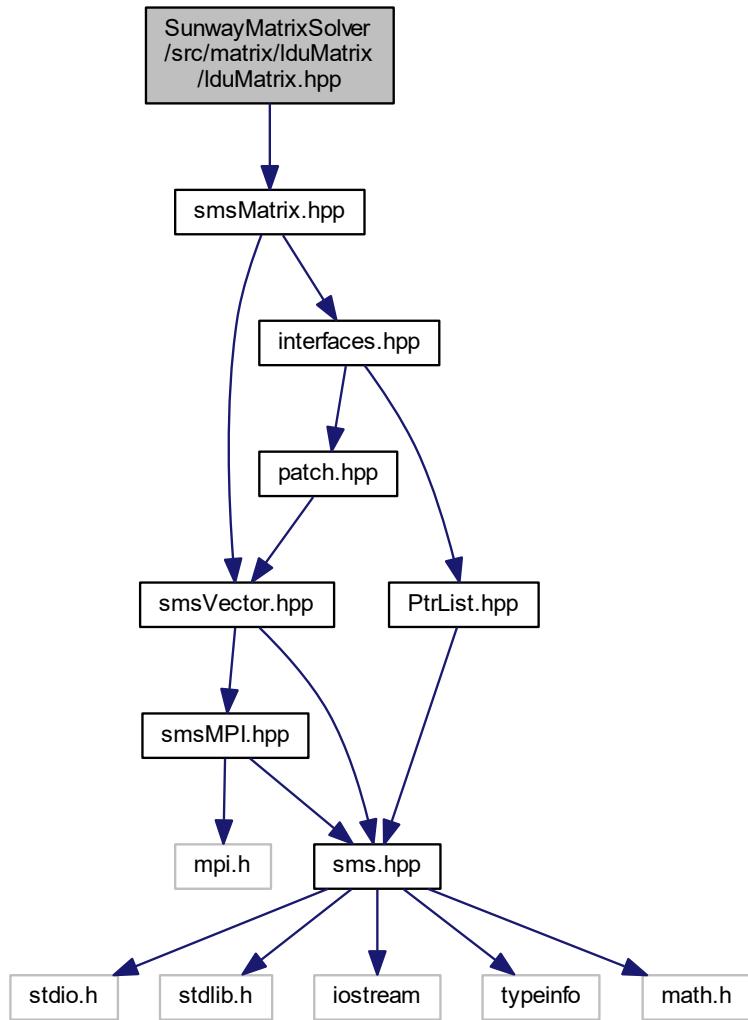
Include dependency graph for `lDUMatrix.cpp`:



7.27 SunwayMatrixSolver/src/matrix/lDUMatrix/lDUMatrix.hpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for IduMatrix.hpp:



This graph shows which files directly or indirectly include this file:



Classes

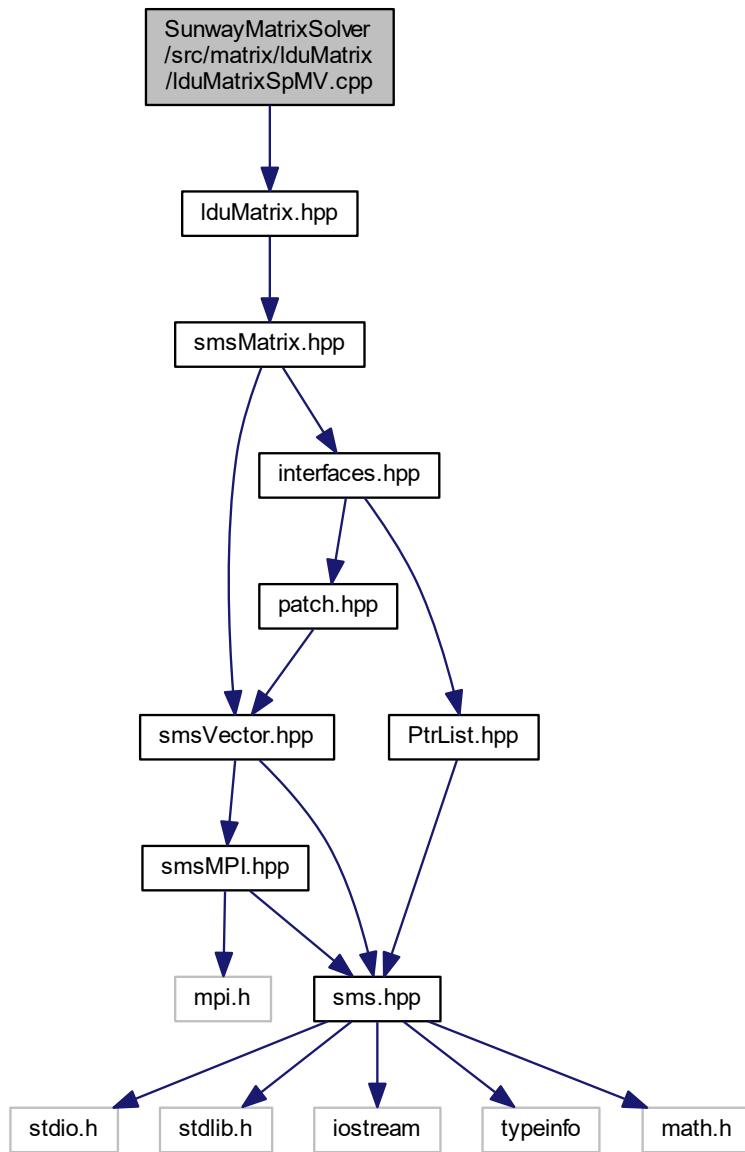
- class [SMS::IduMatrix](#)

Namespaces

- [SMS](#)

7.28 SunwayMatrixSolver/src/matrix/lduMatrix/lduMatrixSpMV.cpp File Reference

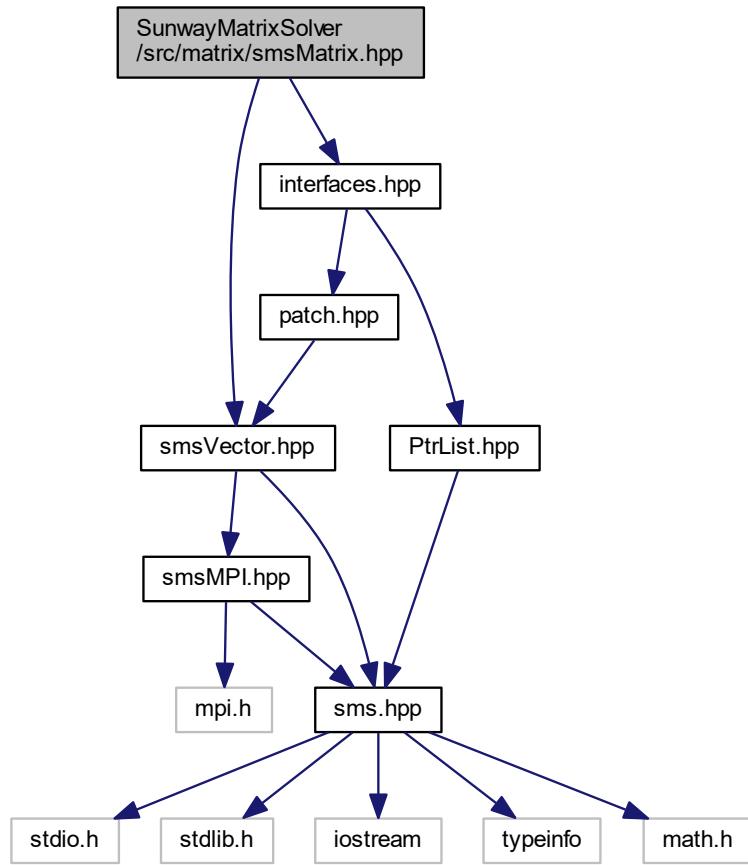
```
#include "lduMatrix.hpp"
Include dependency graph for lduMatrixSpMV.cpp:
```



7.29 SunwayMatrixSolver/src/matrix/smsMatrix.hpp File Reference

```
#include "smsVector.hpp"
#include "interfaces.hpp"
```

Include dependency graph for smsMatrix.hpp:



This graph shows which files directly or indirectly include this file:



Classes

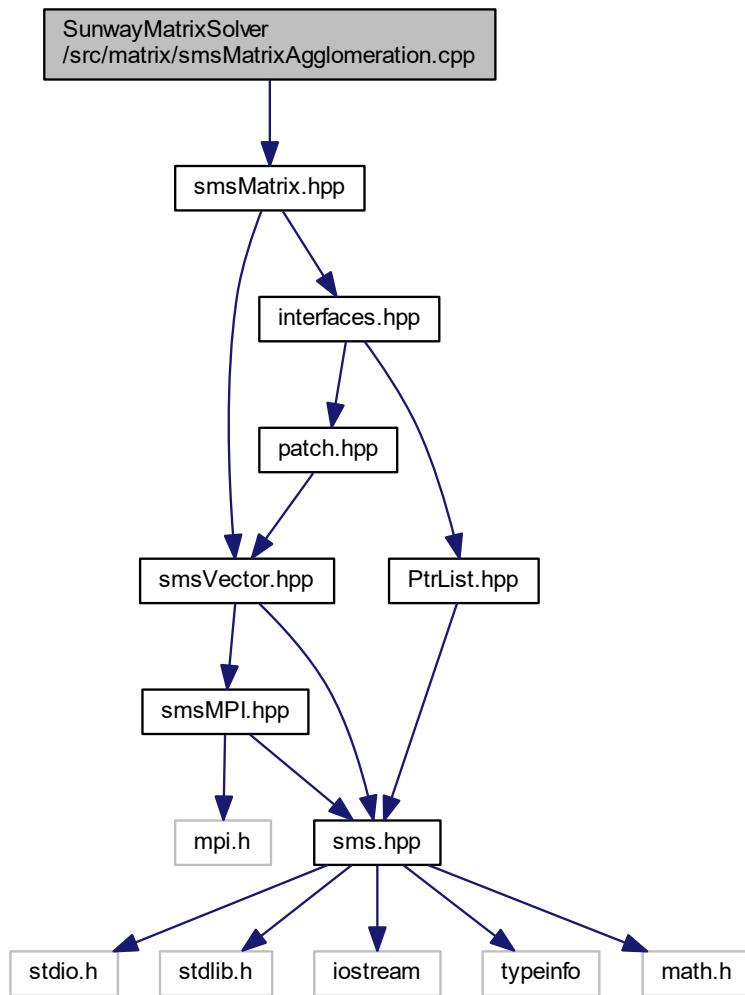
- class [SMS::matrix](#)
- class [SMS::matrix::solverPerformance](#)
- class [SMS::matrix::agglomeration](#)
- class [SMS::matrix::solver](#)
- class [SMS::matrix::preconditioner](#)
- class [SMS::matrix::smoother](#)

Namespaces

- [SMS](#)

7.30 SunwayMatrixSolver/src/matrix/smsMatrixAgglomeration.cpp File Reference

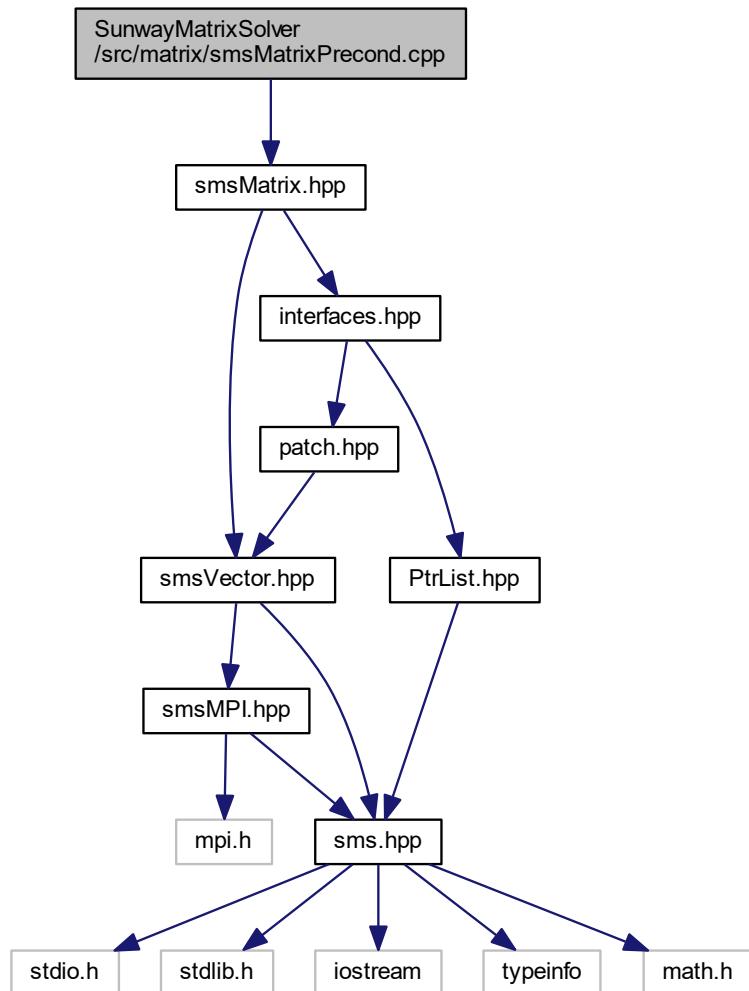
```
#include "smsMatrix.hpp"
Include dependency graph for smsMatrixAgglomeration.cpp:
```



7.31 SunwayMatrixSolver/src/matrix/smsMatrixPrecond.cpp File Reference

```
#include "smsMatrix.hpp"
```

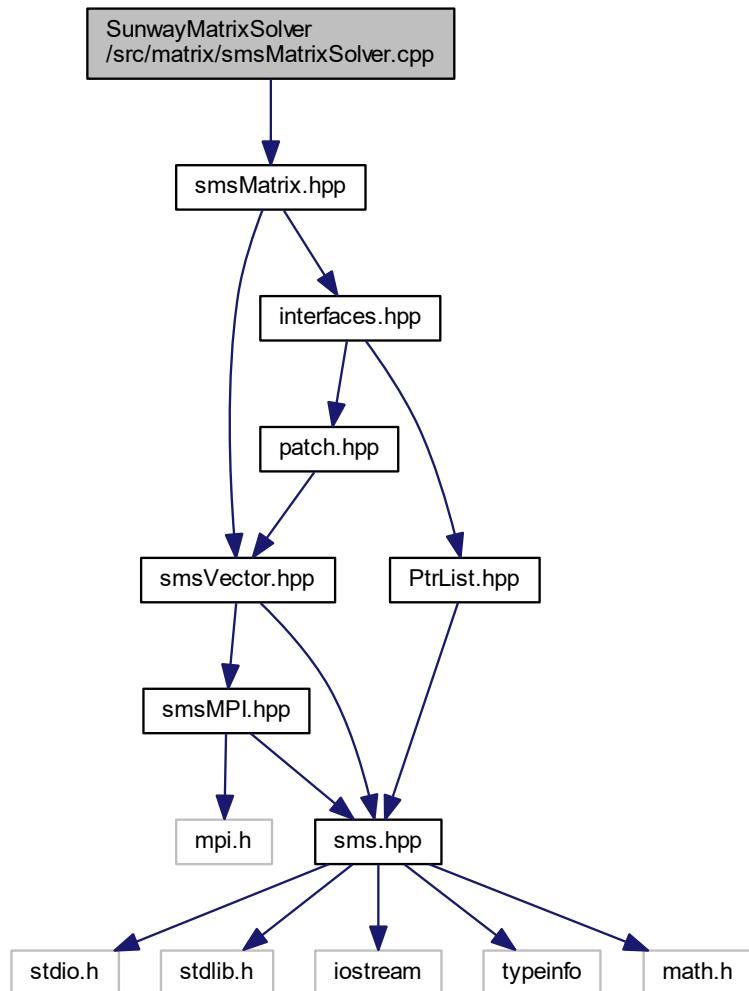
Include dependency graph for smsMatrixPrecond.cpp:



7.32 SunwayMatrixSolver/src/matrix/smsMatrixSolver.cpp File Reference

```
#include "smsMatrix.hpp"
```

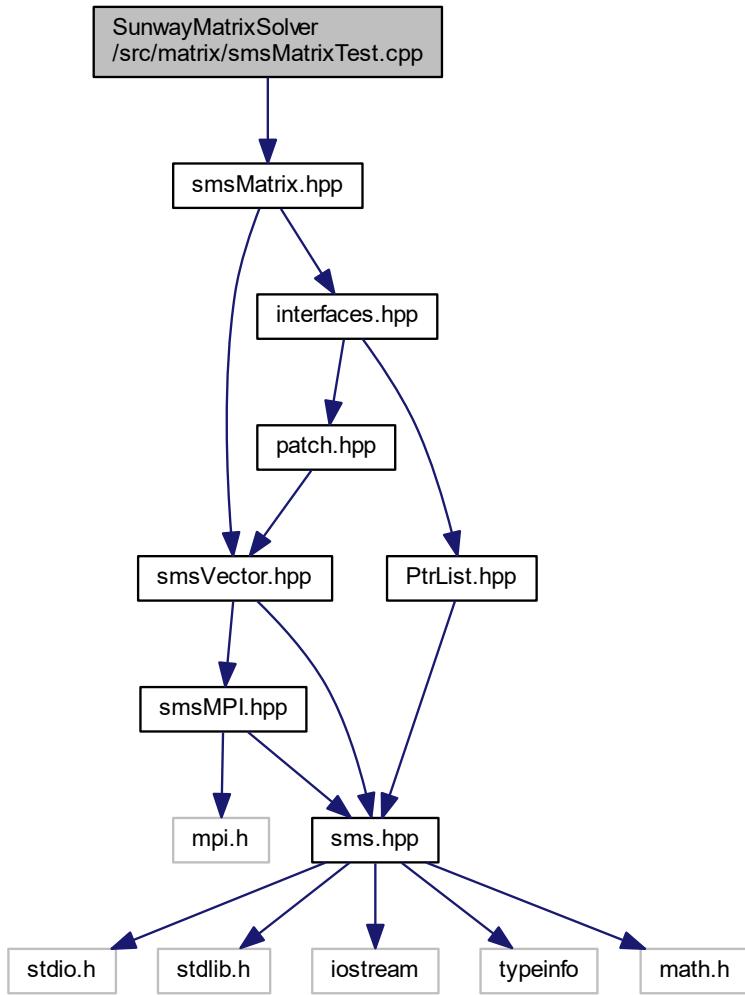
Include dependency graph for smsMatrixSolver.cpp:



7.33 SunwayMatrixSolver/src/matrix/smsMatrixTest.cpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for smsMatrixTest.cpp:



7.34 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDiagonal/csrDiagPrecond.cpp File Reference ←

7.35 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDiagonal/csrDiagPrecond.hpp File Reference ←

7.36 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDIC/csrDICPrecond.cpp File Reference

7.37 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDIC/csrDICPrecond.hpp File Reference

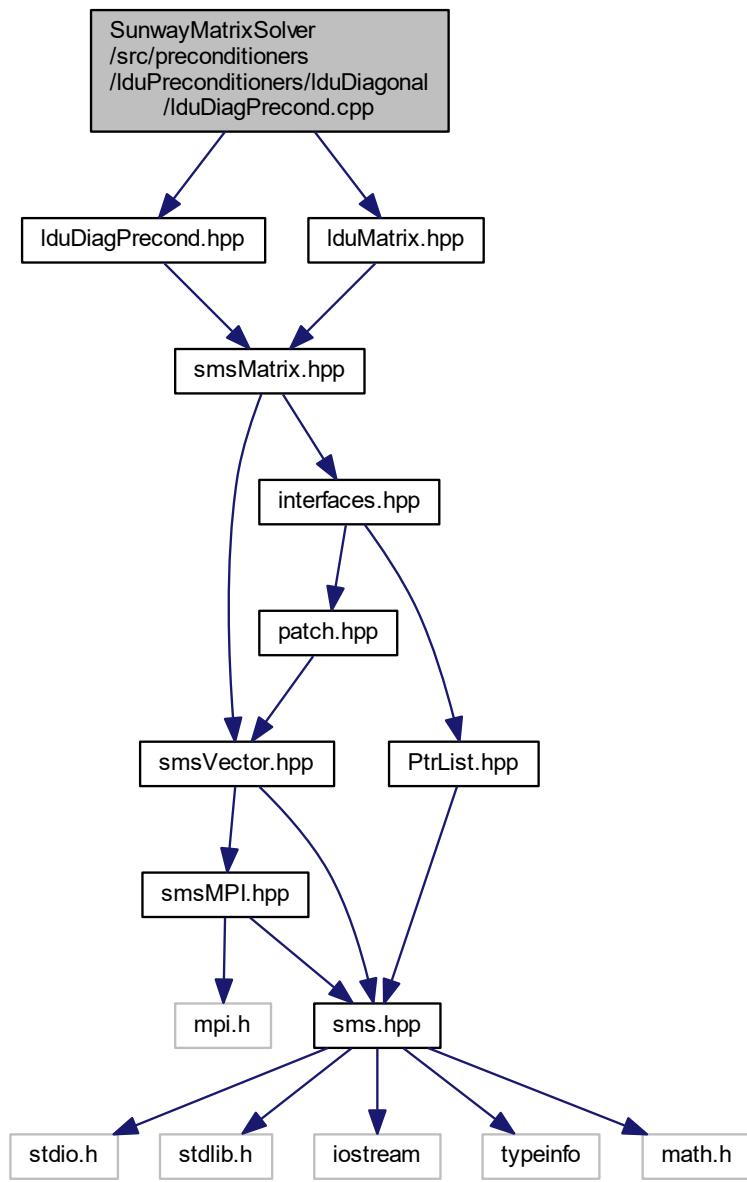
7.38 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDILU/csrDILU
Precond.cpp File Reference

7.39 SunwayMatrixSolver/src/preconditioners/csrPreconditioners/csrDILU/csrDILU
Precond.hpp File Reference

7.40 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDiagonal/lduDiag
Precond.cpp File Reference

```
#include "lduDiagPrecond.hpp"
#include "lduMatrix.hpp"
```

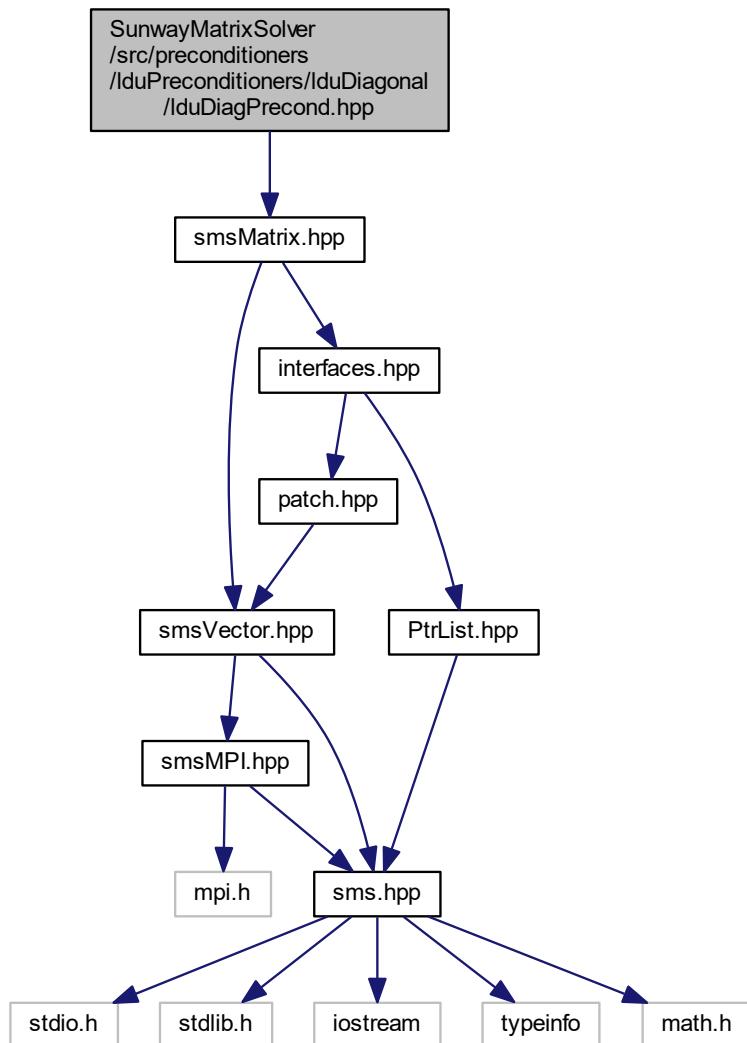
Include dependency graph for lduDiagPrecond.cpp:



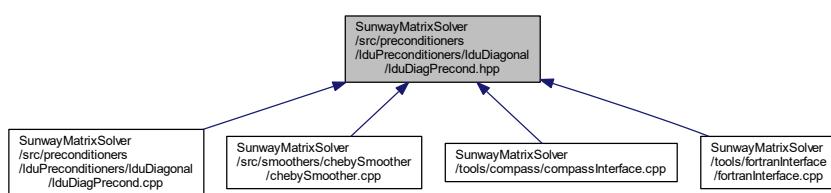
7.41 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDiagonal/lduDiagPrecond.hpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for `lDUDiagPrecond.hpp`:



This graph shows which files directly or indirectly include this file:



Classes

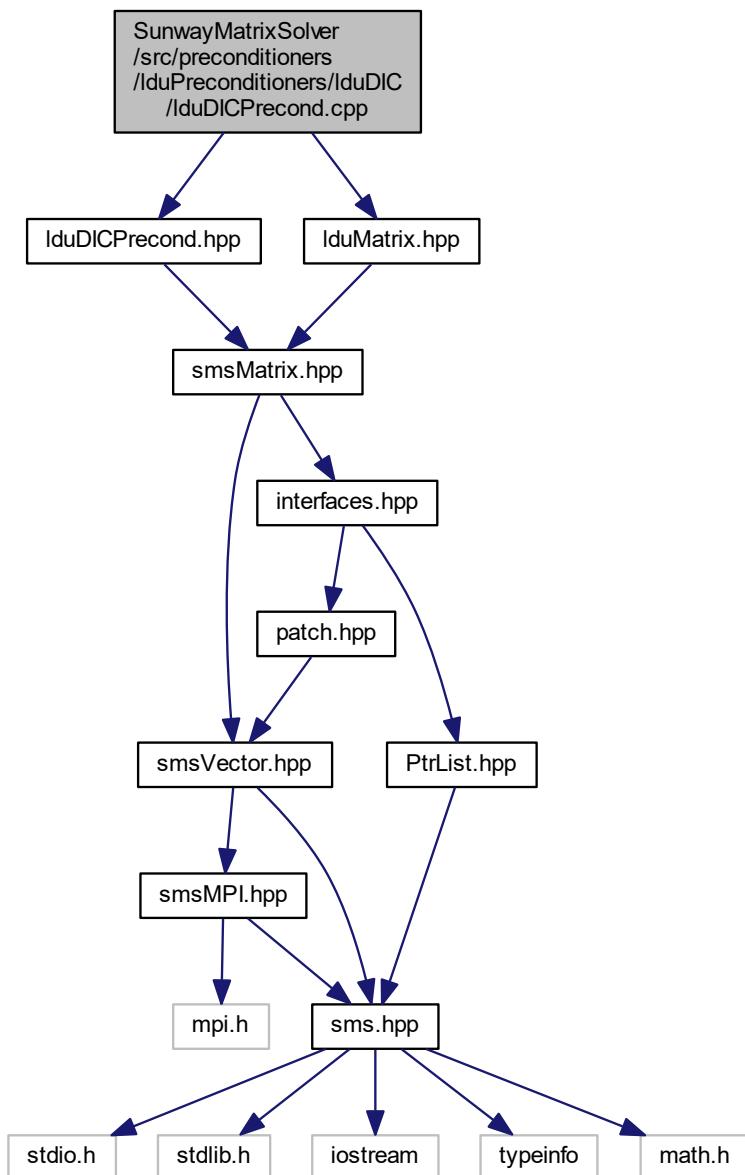
- class [SMS::lDUDiagPrecond](#)

Namespaces

- SMS

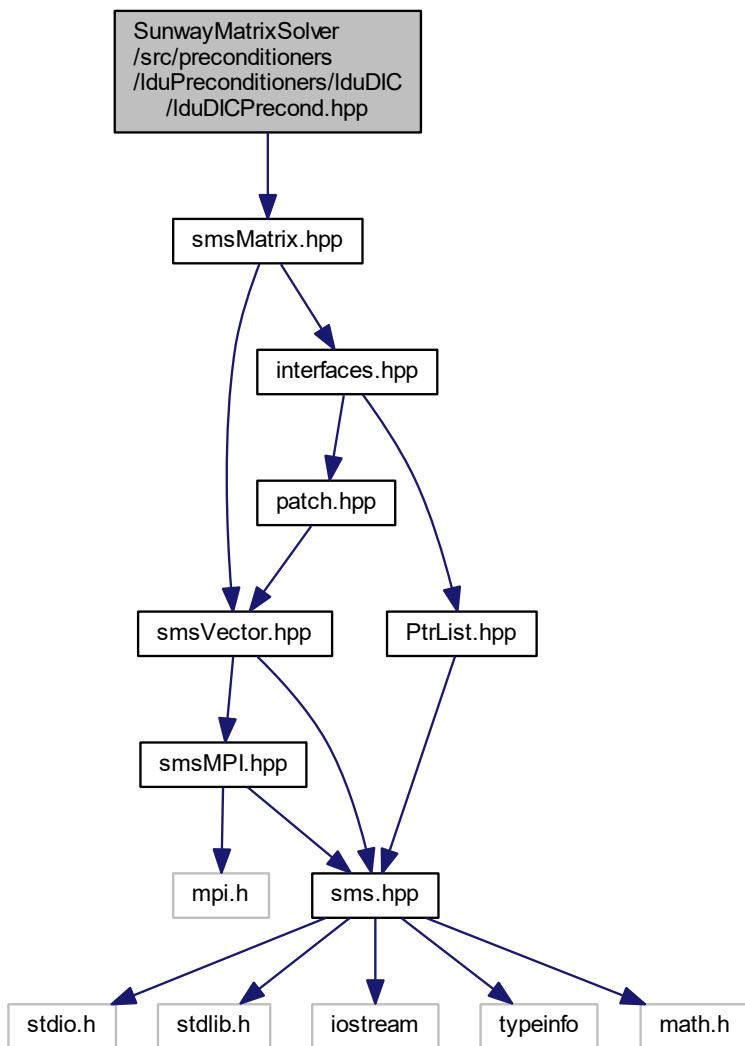
7.42 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDIC/lduDICPrecond.cpp File Reference

```
#include "lduDICPrecond.hpp"
#include "lduMatrix.hpp"
Include dependency graph for lduDICPrecond.cpp:
```

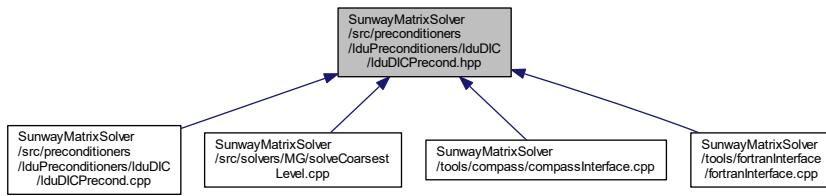


7.43 SunwayMatrixSolver/src/preconditioners/lDuPreconditioners/lDuDIC/lDuDICPrecond.hpp File Reference

```
#include "smsMatrix.hpp"
Include dependency graph for lDuDICPrecond.hpp:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::lduDICPrecond](#)

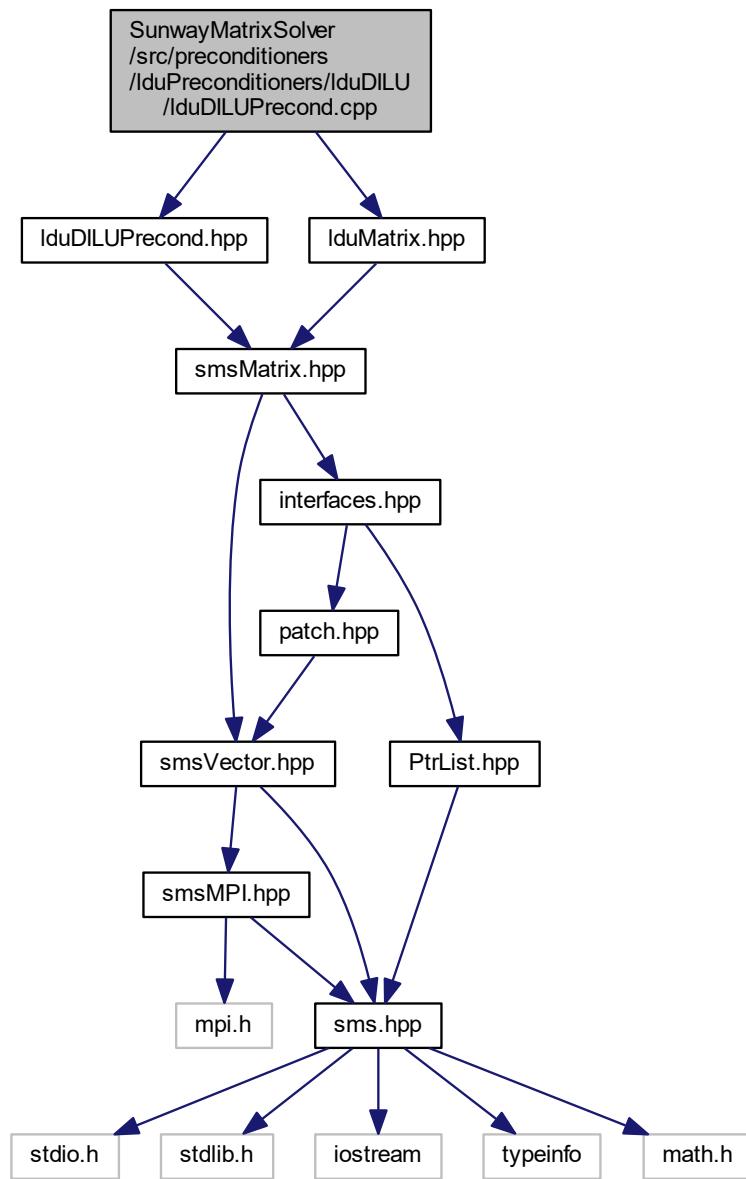
Namespaces

- [SMS](#)

7.44 SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDILU/lduDILUPrecond.cpp File Reference

```
#include "lduDILUPrecond.hpp"
#include "lduMatrix.hpp"
```

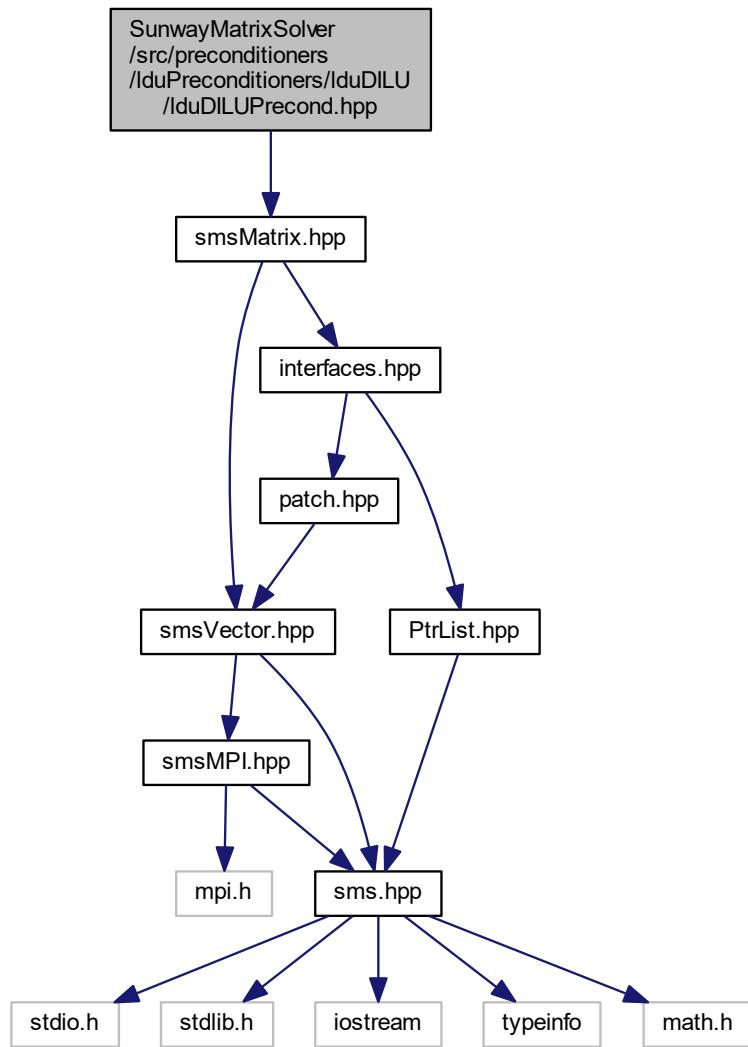
Include dependency graph for `IduDILUPrecond.cpp`:



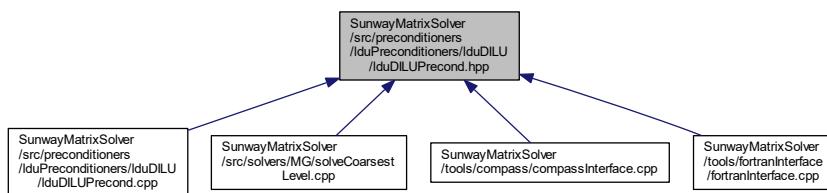
7.45 SunwayMatrixSolver/src/preconditioners/IduPreconditioners/IduDILU/IduDILUPrecond.hpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for lduDILUPrecond.hpp:



This graph shows which files directly or indirectly include this file:



Classes

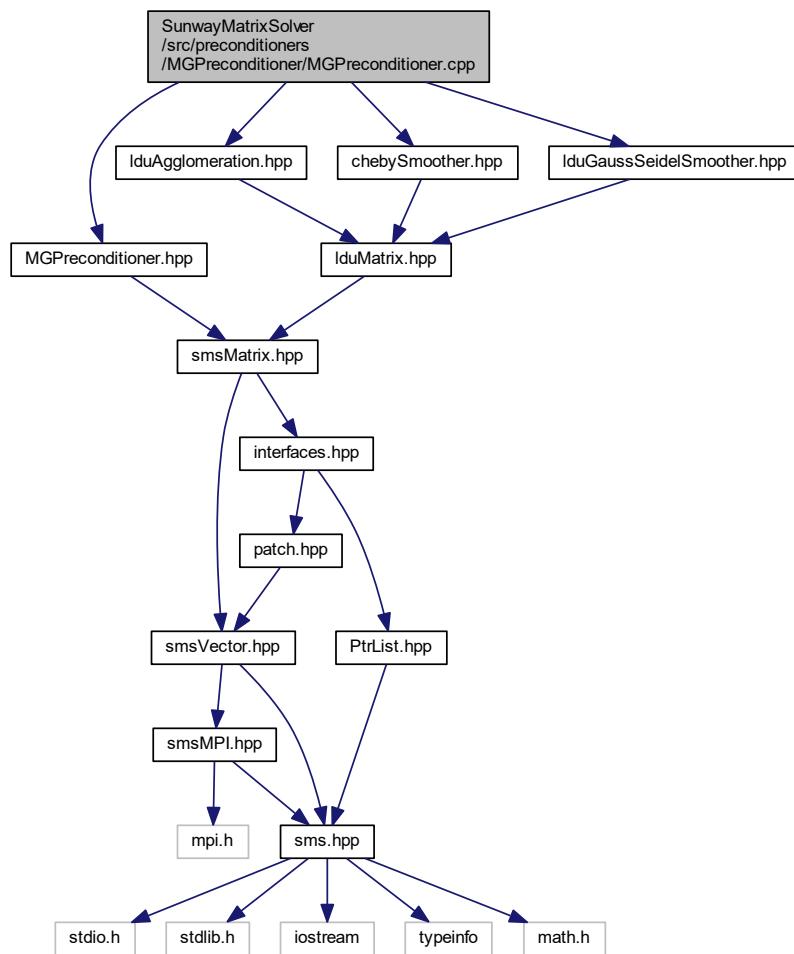
- class [SMS::lduDILUPrecond](#)

Namespaces

- SMS

7.46 SunwayMatrixSolver/src/preconditioners/MGPreconditioner/MGPreconditioner.cpp File Reference

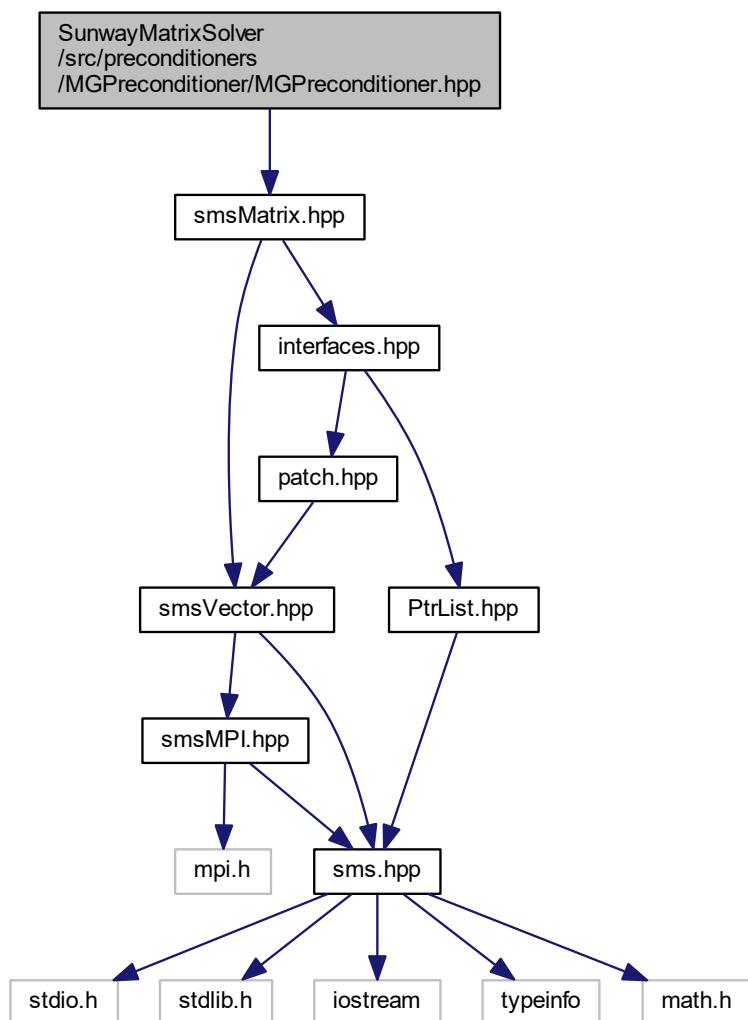
```
#include "MGPreconditioner.hpp"
#include "lduAgglomeration.hpp"
#include "chebySmoother.hpp"
#include "lduGaussSeidelSmoother.hpp"
Include dependency graph for MGPreconditioner.cpp:
```



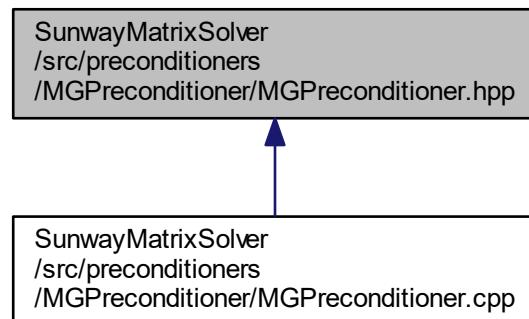
7.47 SunwayMatrixSolver/src/preconditioners/MGPreconditioner/MGPreconditioner.hpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for MGPreconditioner.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::MGPrecond](#)

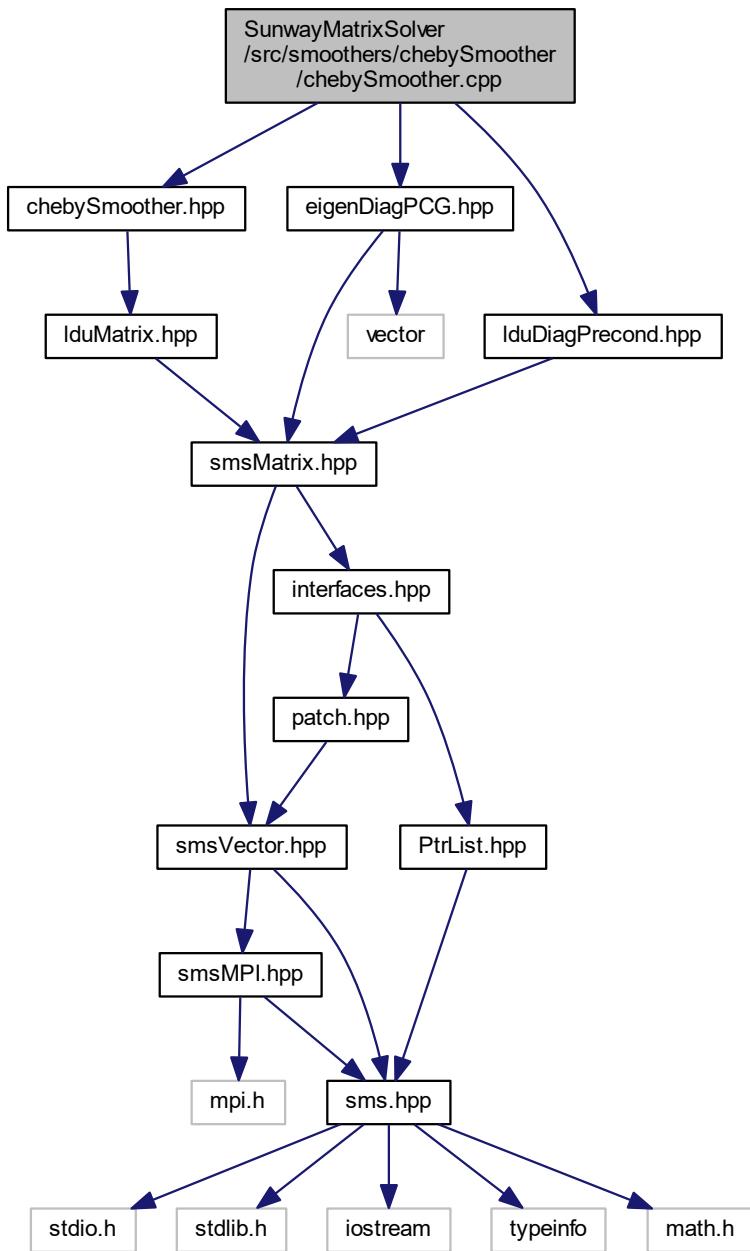
Namespaces

- [SMS](#)

7.48 SunwayMatrixSolver/src/smoothers/chebySmoothen/chebySmoothen.cpp File Reference

```
#include "chebySmoothen.hpp"
#include "eigenDiagPCG.hpp"
#include "lduDiagPrecond.hpp"
```

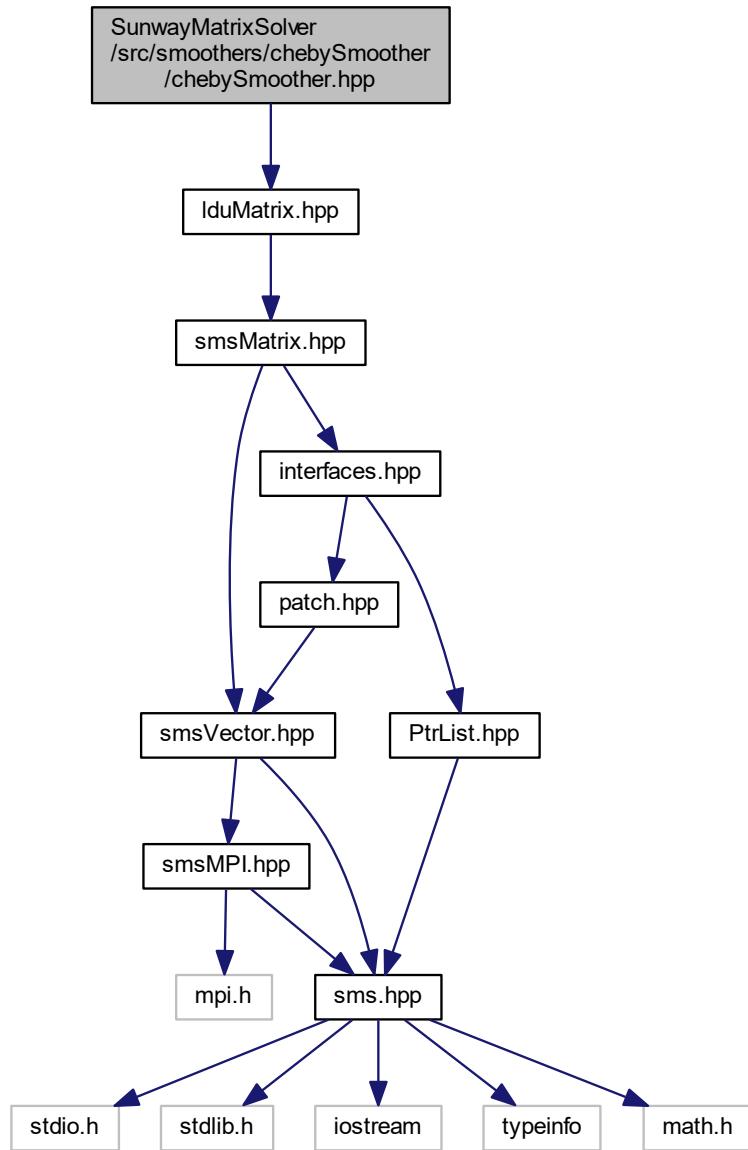
Include dependency graph for chebySmoother.cpp:



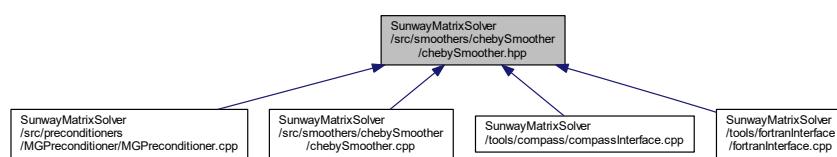
7.49 SunwayMatrixSolver/src/smoothers/chebySmoother/chebySmoother.hpp File Reference

```
#include "lduMatrix.hpp"
```

Include dependency graph for chebySmoother.hpp:



This graph shows which files directly or indirectly include this file:



- class [SMS::chebySmoother](#)

Namespaces

- [SMS](#)

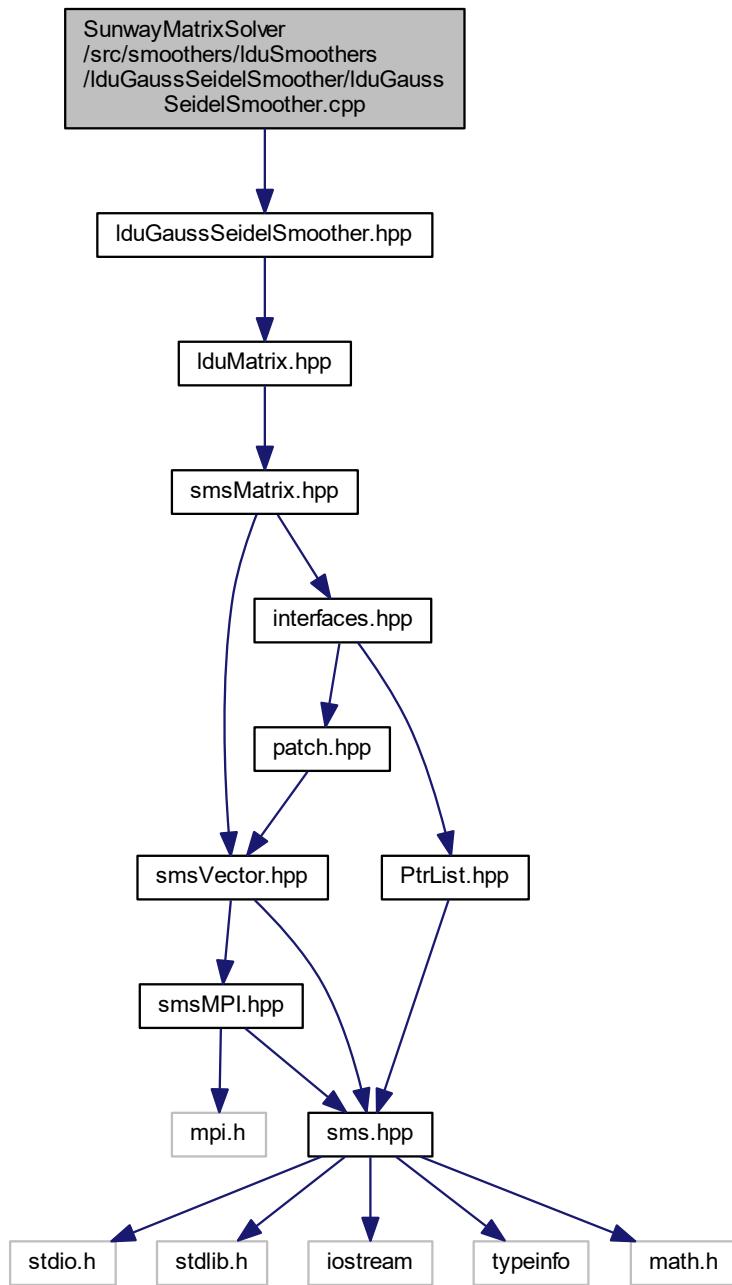
7.50 SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmoother/csrGaussSeidelSmoother.cpp File Reference

7.51 SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmoother/csrGaussSeidelSmoother.hpp File Reference

7.52 SunwayMatrixSolver/src/smoothers/lduSmoothers/lдуGaussSeidelSmoother/lduGaussSeidelSmoother.cpp File Reference

```
#include "lduGaussSeidelSmoother.hpp"
```

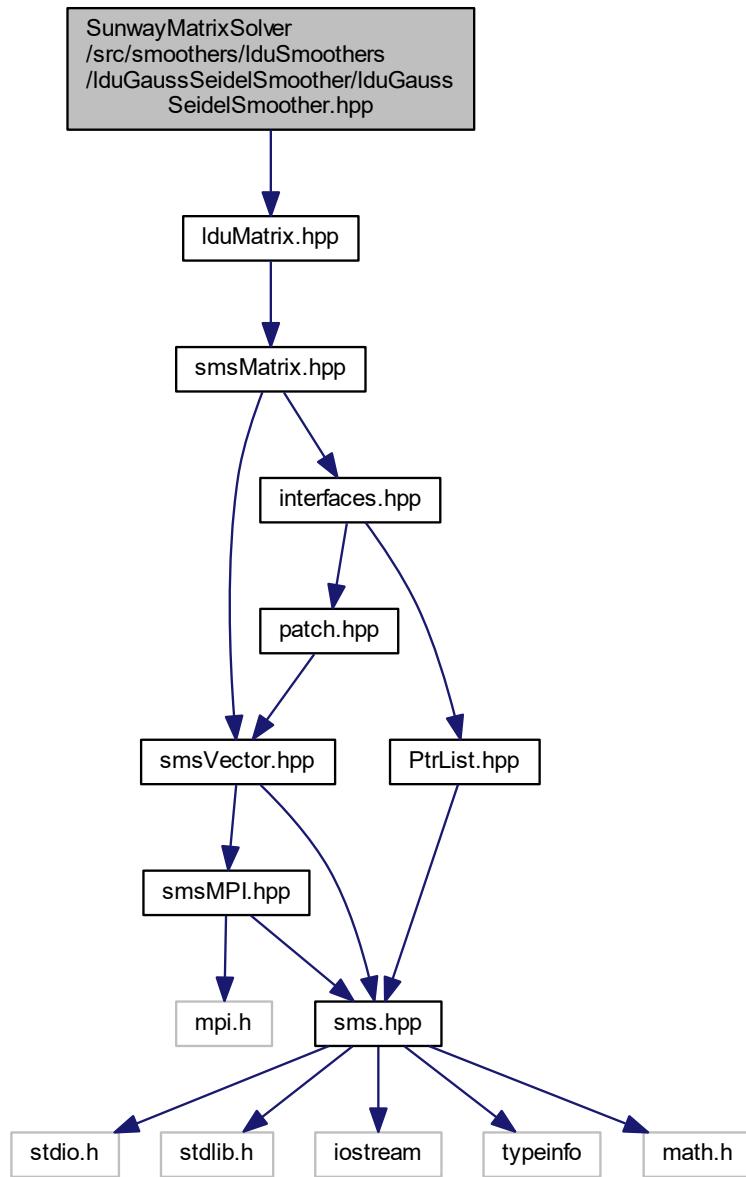
Include dependency graph for `lDUGaussSeidelSmoothes.cpp`:



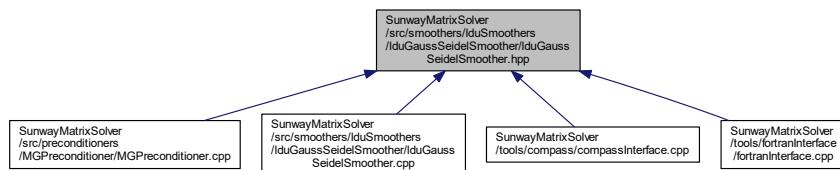
7.53 SunwayMatrixSolver/src/smoothers/lDU Smoothers/lDUGaussSeidelSmoothes/lDUGaussSeidelSmoothes.hpp File Reference

```
#include "lDUMatrix.hpp"
```

Include dependency graph for IduGaussSeidelSmoothe...er.hpp:



This graph shows which files directly or indirectly include this file:



Classes

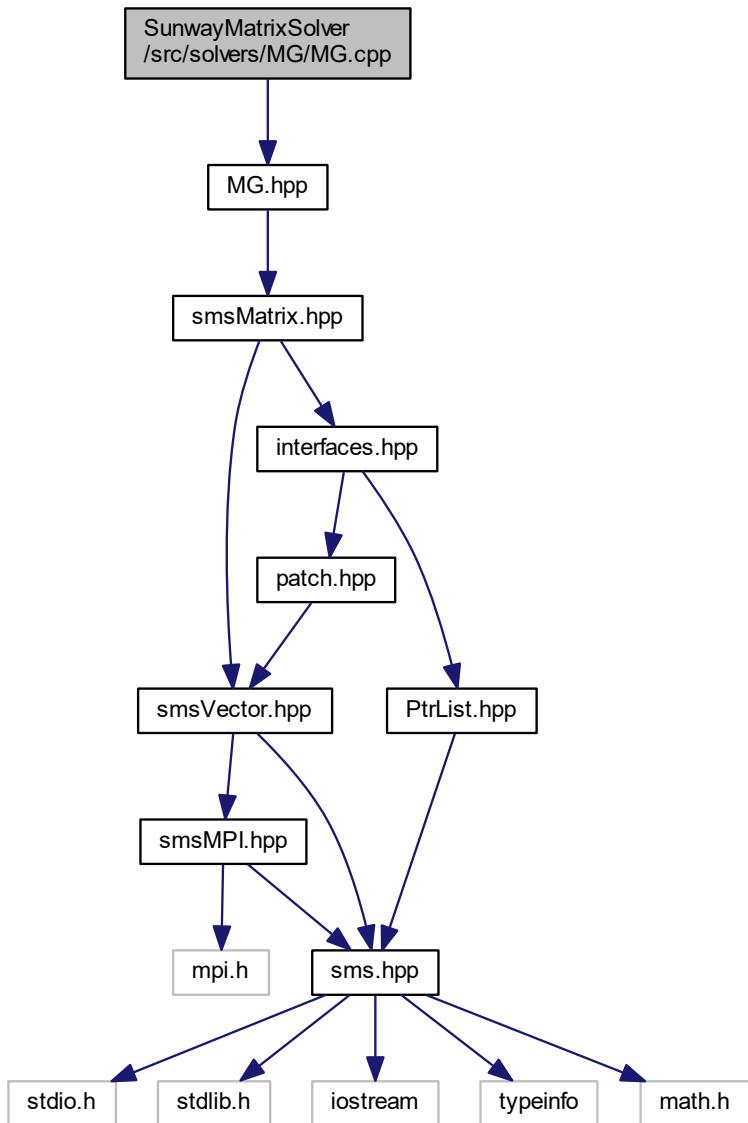
- class `SMS::IduGaussSeidelSmoothes`

Namespaces

- SMS

7.54 SunwayMatrixSolver/src/solvers/MG/MG.cpp File Reference

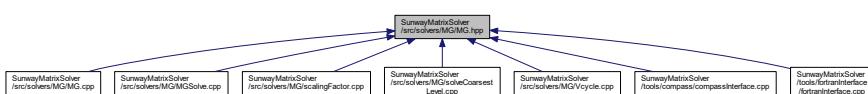
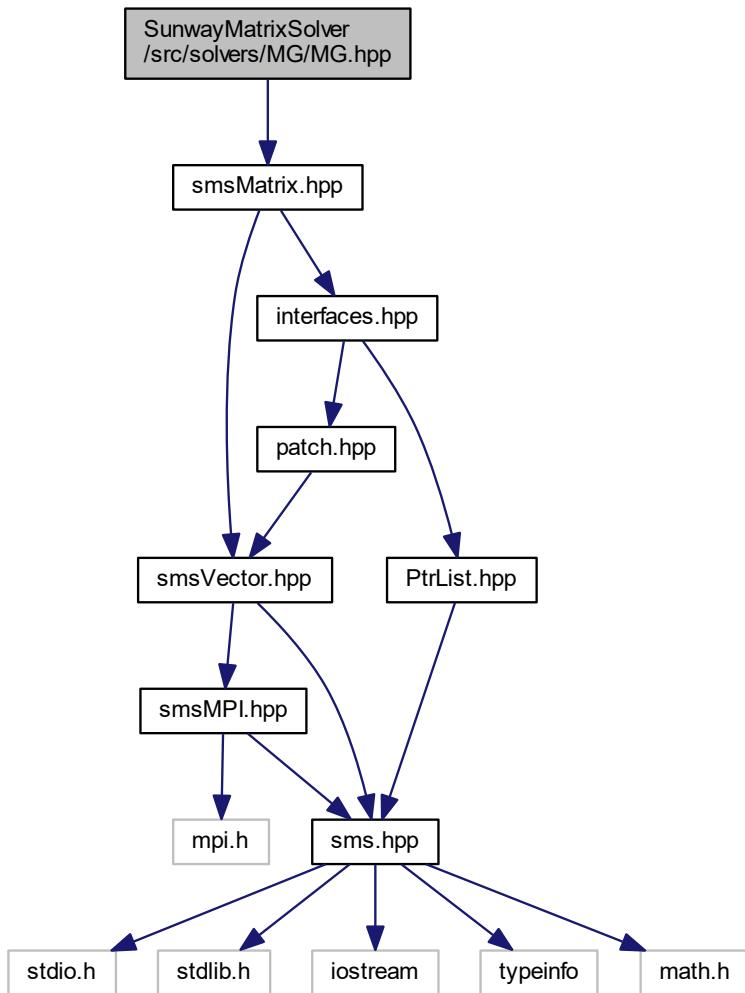
```
#include "MG.hpp"
Include dependency graph for MG.cpp:
```



7.55 SunwayMatrixSolver/src/solvers/MG/MG.hpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for MG.hpp:



Classes

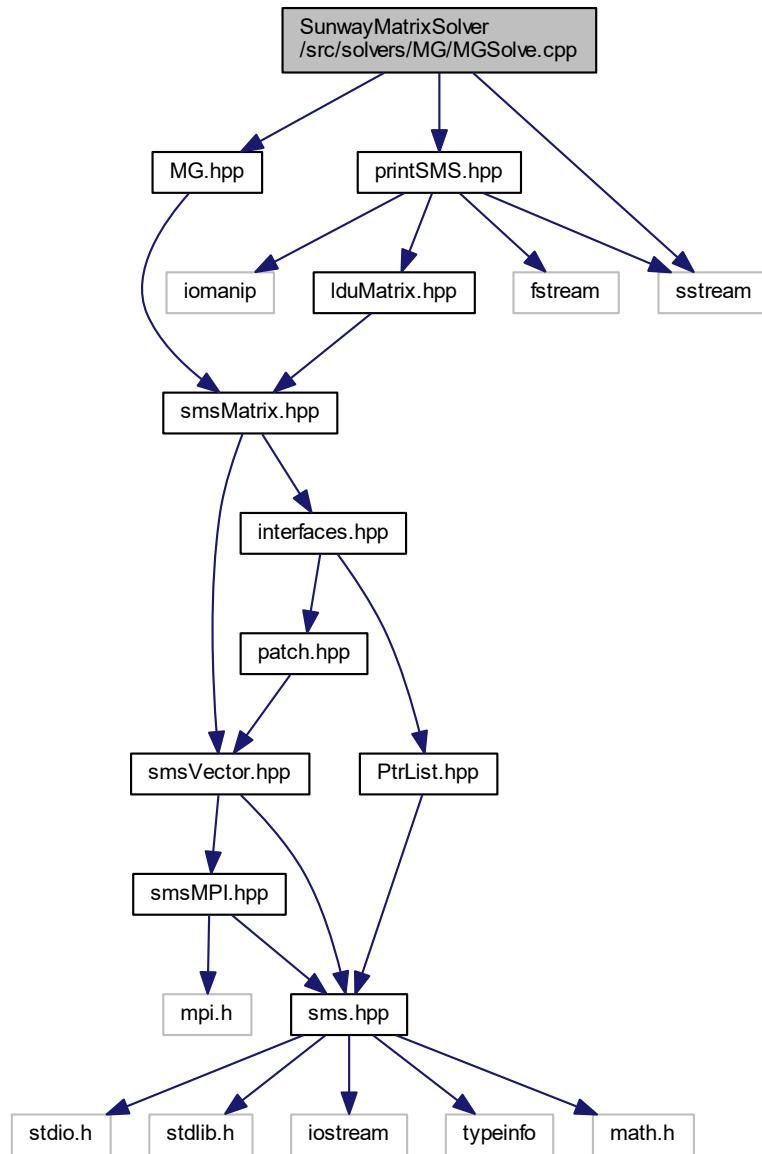
- class [SMS::MGSolver](#)

Namespaces

- SMS

7.56 SunwayMatrixSolver/src/solvers/MG/MGSolve.cpp File Reference

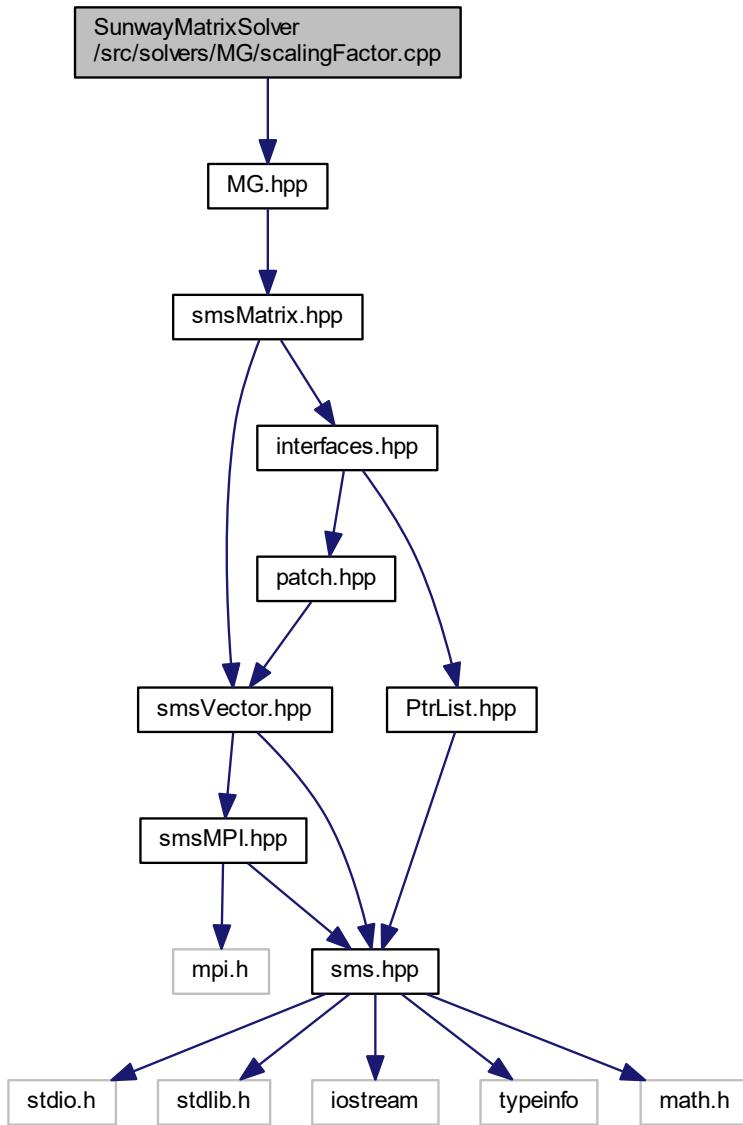
```
#include "MG.hpp"
#include "printSMS.hpp"
#include <iostream>
Include dependency graph for MGSolve.cpp:
```



7.57 SunwayMatrixSolver/src/solvers/MG/scalingFactor.cpp File Reference

```
#include "MG.hpp"
```

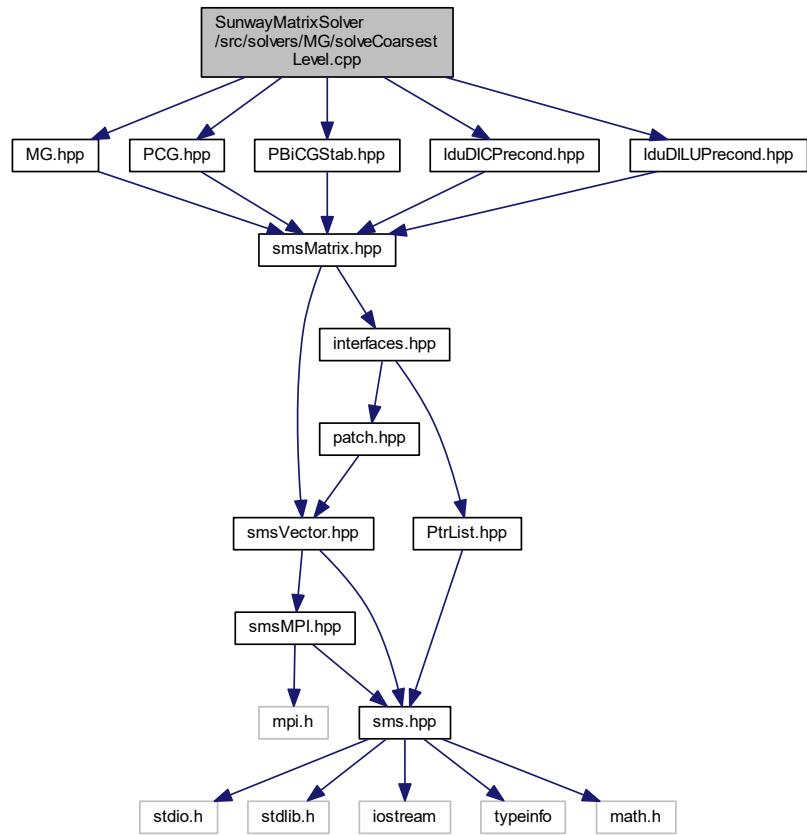
Include dependency graph for scalingFactor.cpp:



7.58 SunwayMatrixSolver/src/solvers/MG/solveCoarsestLevel.cpp File Reference

```
#include "MG.hpp"
#include "PCG.hpp"
#include "PBiCGStab.hpp"
#include "lduDICPrecond.hpp"
```

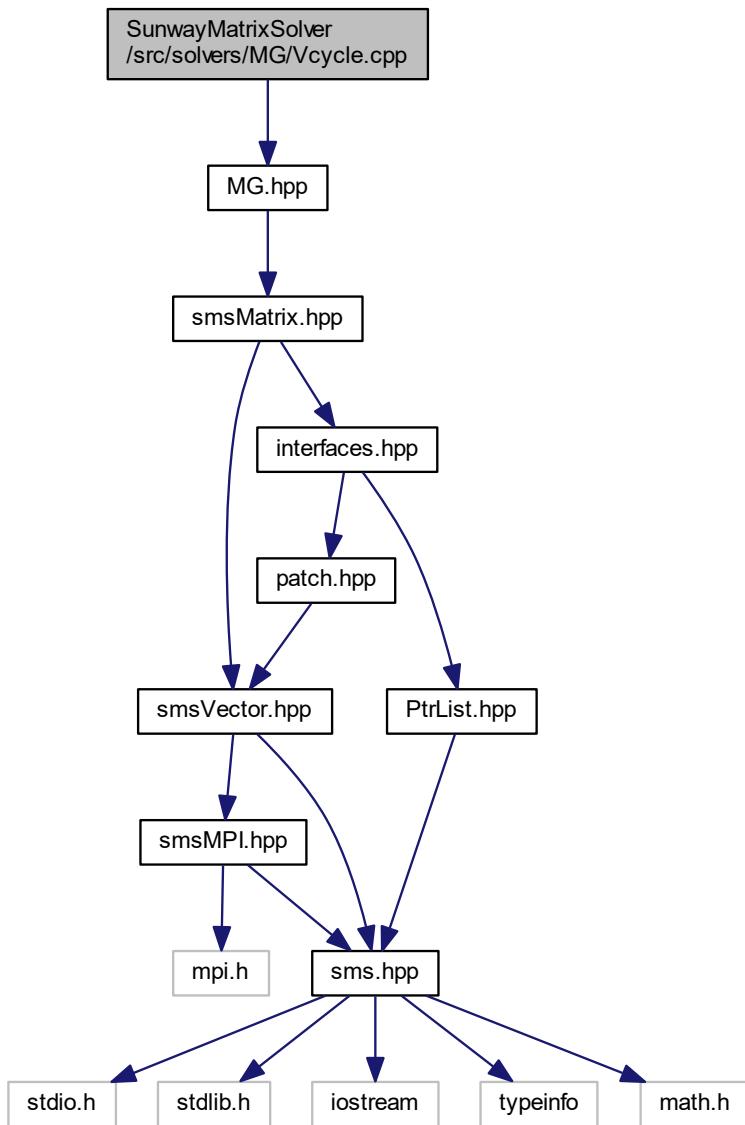
```
#include "lduDILUPrecond.hpp"
Include dependency graph for solveCoarsestLevel.cpp:
```



7.59 SunwayMatrixSolver/src/solvers/MG/Vcycle.cpp File Reference

```
#include "MG.hpp"
```

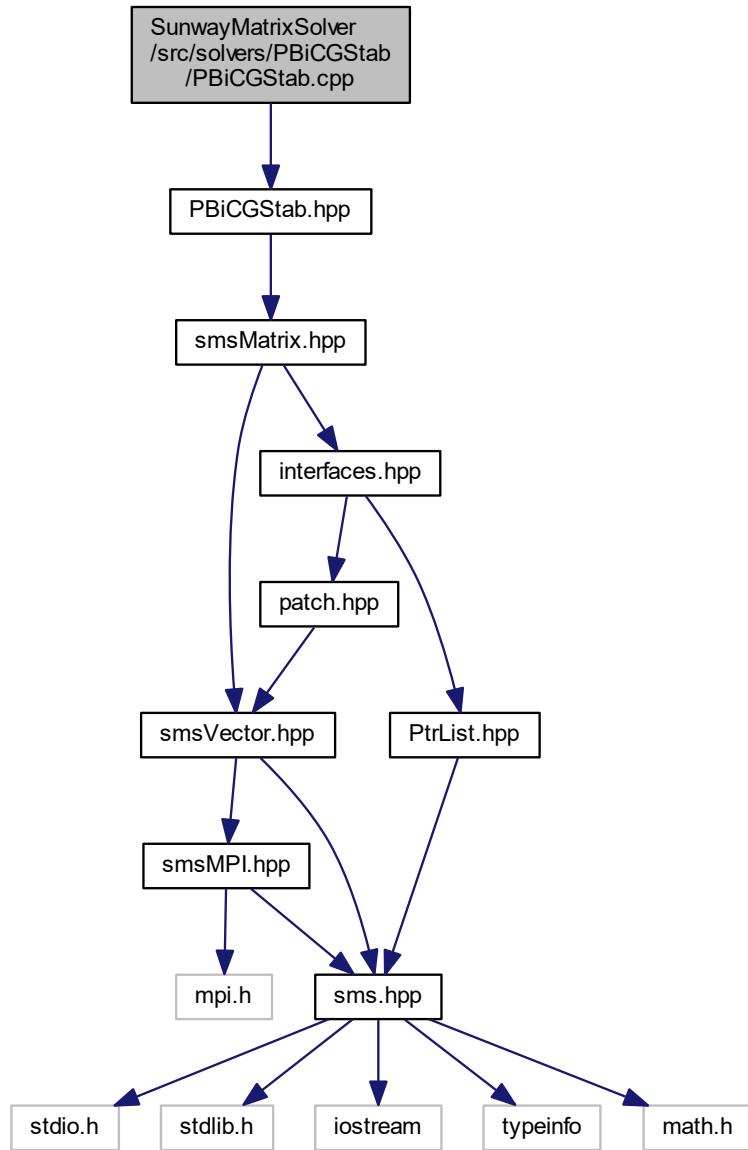
Include dependency graph for Vcycle.cpp:



7.60 SunwayMatrixSolver/src/solvers/PBiCGStab/PBiCGStab.cpp File Reference

```
#include "PBiCGStab.hpp"
```

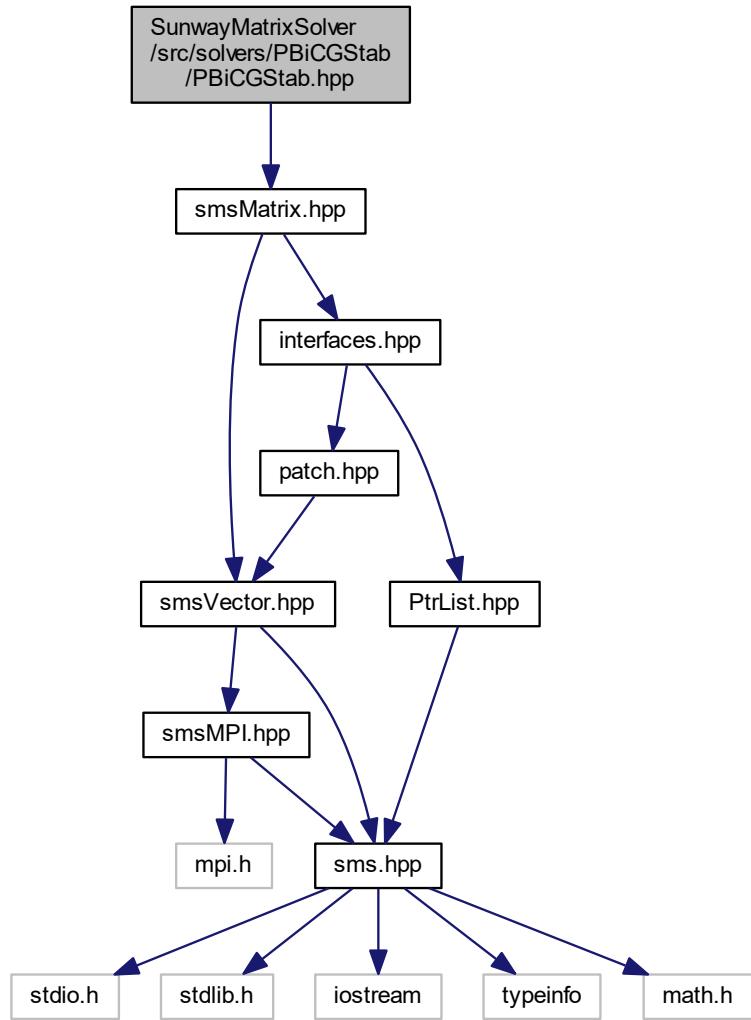
Include dependency graph for PBiCGStab.cpp:



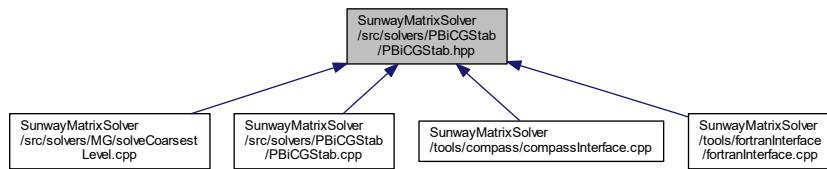
7.61 SunwayMatrixSolver/src/solvers/PBiCGStab/PBiCGStab.hpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for PBiCGStab.hpp:



This graph shows which files directly or indirectly include this file:



Classes

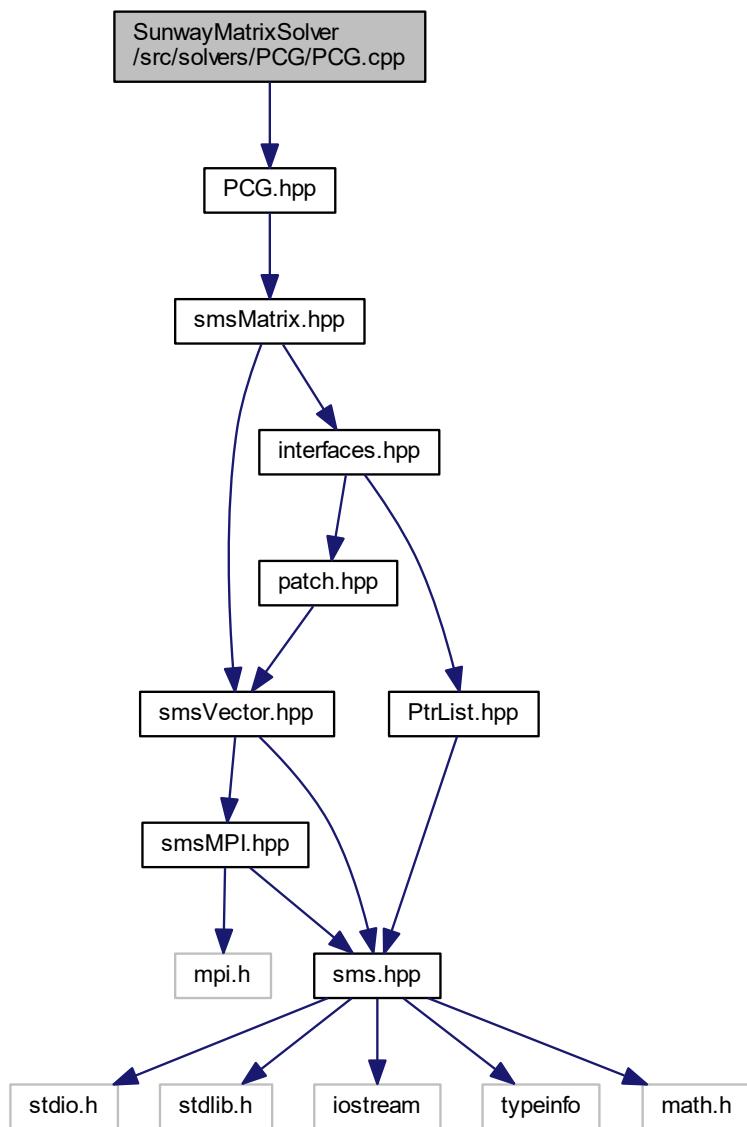
- class [SMS::PBiCGStab](#)

Namespaces

- SMS

7.62 SunwayMatrixSolver/src/solvers/PCG/PCG.cpp File Reference

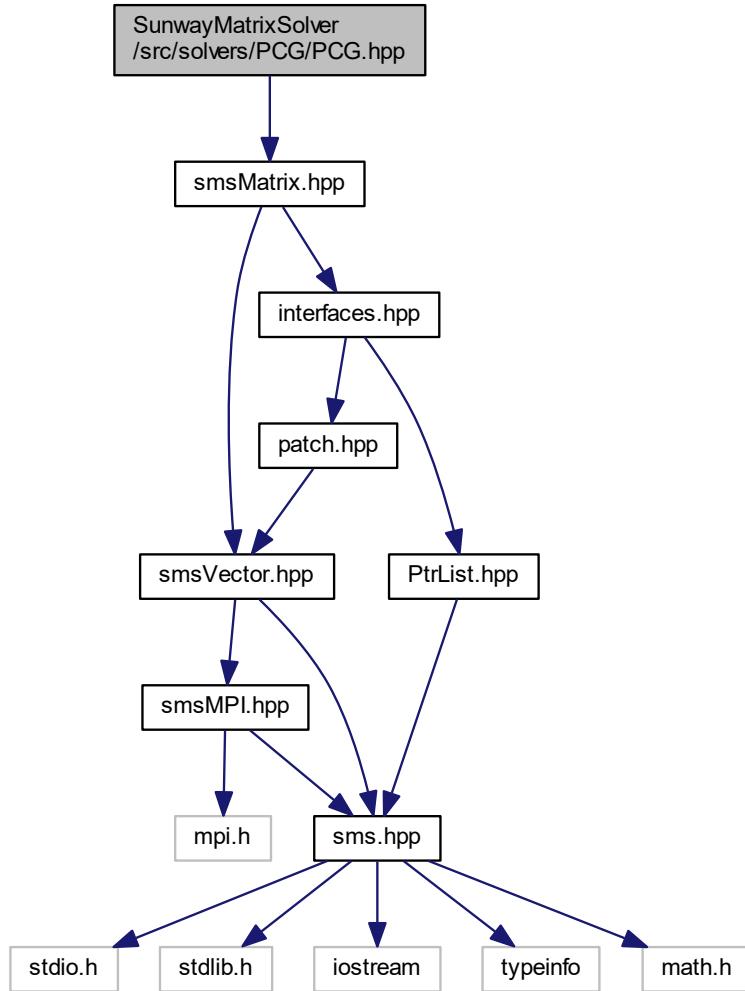
```
#include "PCG.hpp"  
Include dependency graph for PCG.cpp:
```



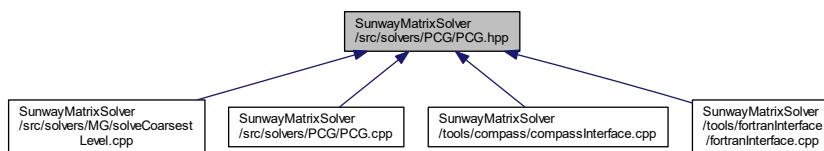
7.63 SunwayMatrixSolver/src/solvers/PCG/PCG.hpp File Reference

```
#include "smsMatrix.hpp"
```

Include dependency graph for PCG.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [SMS::PCG](#)

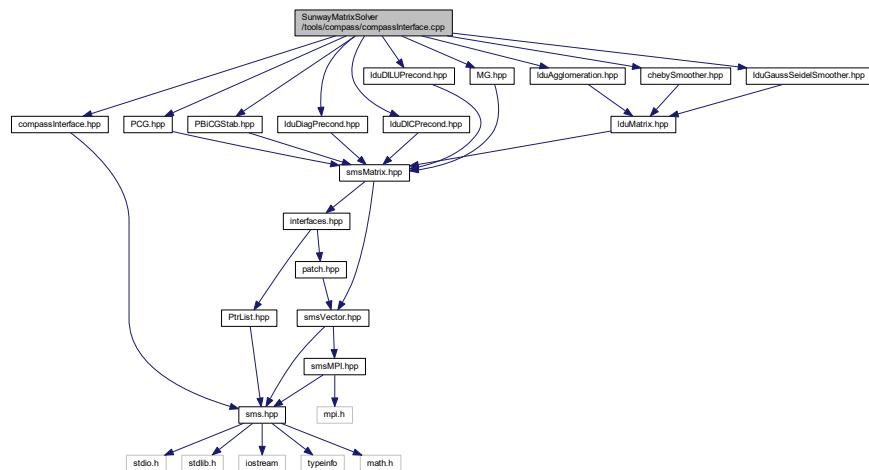
Namespaces

- [SMS](#)

7.64 SunwayMatrixSolver/tools/compass/compassInterface.cpp File Reference

```
#include "compassInterface.hpp"
#include "PCG.hpp"
#include "PBiCGStab.hpp"
#include "lduDiagPrecond.hpp"
#include "lduDICPrecond.hpp"
#include "lduDILUPrecond.hpp"
#include "MG.hpp"
#include "lduAgglomeration.hpp"
#include "chebySmoothen.hpp"
#include "lduGaussSeidelSmoothen.hpp"
```

Include dependency graph for compassInterface.cpp:



Macros

- #define [PTR2OBJ\(ptr, cls, obj\)](#)

7.64.1 Macro Definition Documentation

7.64.1.1 PTR2OBJ

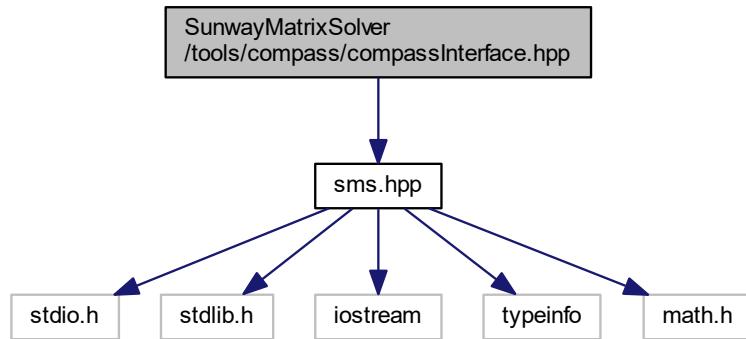
```
#define PTR2OBJ(
    ptr,
    cls,
    obj )
```

Value:

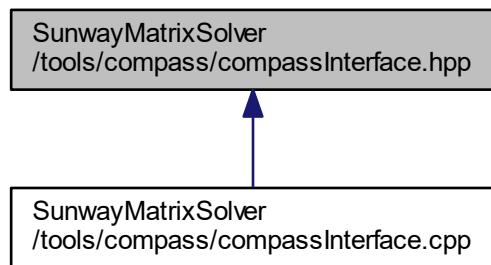
```
cls* ptr##_tmp = (cls*)*ptr; \
CHECK_POINTER(ptr##_tmp) \
cls& obj = *ptr##_tmp;
```

7.65 SunwayMatrixSolver/tools/compass/compassInterface.hpp File Reference

#include "sms.hpp"
Include dependency graph for compassInterface.hpp:



This graph shows which files directly or indirectly include this file:



Namespaces

- [SMS](#)

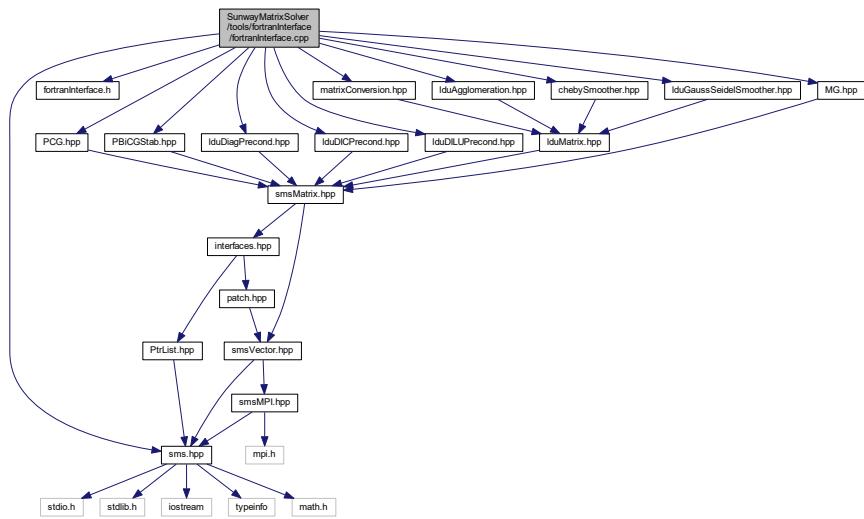
Functions

- void [SMS::construct_sw_matrix_](#) (long int *APtrPtr, const label *nCellsPtr, const label *rowsPtr, const label *colsPtr, const label *sizePtr)
- void [SMS::construct_sw_matrix_interfaces_](#) (long int *APtrPtr, const label *nNeiProcsPtr, const label *destRankPtr, const label *offDiagRowsPtr, const label *offDiagStartsPtr)
- void [SMS::fill_sw_matrix_coefficients_](#) (long int *APtrPtr, const scalar *diagPtr, const scalar *upperPtr, const scalar *lowerPtr)
- void [SMS::fill_sw_matrix_interfaces_coefficients_](#) (long int *APtrPtr, const label *offDiagStartsPtr, const scalar *offDiagCoeffs)
- void [SMS::construct_solver_mg_](#) (long int *mgPtrPtr, long int *APtrPtr, const scalar *faceAreaPtr, const label *smTypePtr, const label *maxLevelsPtr, const label *nCellsCoarsestPtr)
- void [SMS::sw_solver_mg_set_maxiter_](#) (long int *solverPtrPtr, const label *maxIterPtr)
- void [SMS::sw_solver_mg_set_miniter_](#) (long int *solverPtrPtr, const label *minIterPtr)
- void [SMS::sw_solver_mg_set_tol_](#) (long int *solverPtrPtr, const scalar *tolPtr)
- void [SMS::sw_solver_mg_set_reltol_](#) (long int *solverPtrPtr, const scalar *reltolPtr)
- void [SMS::sw_solver_mg_set_npresweeps_](#) (long int *solverPtrPtr, const label *numPtr)
- void [SMS::sw_solver_mg_set_npostsweeps_](#) (long int *solverPtrPtr, const label *numPtr)
- void [SMS::sw_solver_mg_set_nfinesstsweeps_](#) (long int *solverPtrPtr, const label *numPtr)
- void [SMS::sw_solve_mg_](#) (long int *mgPtrPtr, long int *APtrPtr, scalar *xPtr, scalar *bPtr)
- void [SMS::construct_solver_pbicgstab_](#) (long int *solverPtrPtr)
- void [SMS::sw_solver_pbicgstab_set_maxiter_](#) (long int *solverPtrPtr, const label *maxIterPtr)
- void [SMS::sw_solver_pbicgstab_set_miniter_](#) (long int *solverPtrPtr, const label *minIterPtr)
- void [SMS::sw_solver_pbicgstab_set_tol_](#) (long int *solverPtrPtr, const scalar *tolPtr)
- void [SMS::sw_solver_pbicgstab_set_reltol_](#) (long int *solverPtrPtr, const scalar *reltolPtr)
- void [SMS::sw_solver_pbicgstab_set_precond_](#) (long int *solverPtrPtr, long int *APtrPtr, const label *precondTypePtr)
- void [SMS::sw_solve_pbicgstab_](#) (long int *solverPtrPtr, long int *APtrPtr, scalar *xPtr, scalar *bPtr)
- void [SMS::sw_matrix_destroy_](#) (long int *APtrPtr)
- void [SMS::sw_solver_destroy_mg_](#) (long int *solverPtrPtr)
- void [SMS::sw_solver_destroy_pbicgstab_](#) (long int *solverPtrPtr)

7.66 SunwayMatrixSolver/tools/fortranInterface/fortranInterface.cpp File Reference

```
#include "sms.hpp"
#include "fortranInterface.h"
#include "matrixConversion.hpp"
#include "PCG.hpp"
#include "PBiCGStab.hpp"
#include "lduDiagPrecond.hpp"
#include "lduDICPrecond.hpp"
#include "lduDILUPrecond.hpp"
#include "MG.hpp"
#include "lduAgglomeration.hpp"
#include "chebySmoothen.hpp"
```

```
#include "lduGaussSeidelSmoothen.hpp"
Include dependency graph for fortranInterface.cpp:
```



Macros

- `#define RETURN_VALUE(x, objectX, nCells)`

Functions

- `void ldumatrixcreat_ (long int *APtrPtr, label *nCellsPtr, label *upperSizePtr, label *lowerAddrValue, label *upperAddrValue, scalar *lowerValue, scalar *diagValue, scalar *upperValue)`
- `void coo2ldumatrixcreat_ (long int *APtrPtr, const scalar *dataPtr, const label *fRowsPtr, const label *fColsPtr, const label *nCellsPtr, const label *sizePtr, const label *symmPtr)`
- `void csr2ldumatrixcreat_ (long int *APtrPtr, const scalar *dataPtr, const label *fRowsPtr, const label *fColsPtr, const label *nCellsPtr, const label *sizePtr, const label *symmPtr)`
- `void matrixinterfacescreat_ (long int *APtrPtr, const label *nNeiProcsPtr, const label *destRankPtr, const label *locPositionPtr, const label *faceCellsPtr, const scalar *dataPtr)`
- `void pcgsolversolve_ (scalar *xValue, long int *APtrPtr, scalar *bValue, label *nCellsPtr, label *precondTypePtr, scalar *tolPtr, scalar *relTolPtr, label *maxIterPtr, label *minIterPtr, label *num_iterationsPtr, scalar *res_normPtr)`
- `void pbicgstabsversolve_ (scalar *xValue, long int *APtrPtr, scalar *bValue, label *nCellsPtr, label *precondTypePtr, scalar *tolPtr, scalar *relTolPtr, label *maxIterPtr, label *minIterPtr, label *num_iterationsPtr, scalar *res_normPtr)`
- `void mgsolversolve_ (scalar *xValue, long int *APtrPtr, scalar *bValue, label *nCellsPtr, label *aggTypePtr, label *smootherTypePtr, scalar *tolPtr, scalar *relTolPtr, label *maxIterPtr, label *minIterPtr, label *num_iterationsPtr, scalar *res_normPtr, scalar *faceAreaPtr)`

7.66.1 Macro Definition Documentation

7.66.1.1 RETURN_VALUE

```
#define RETURN_VALUE (
    x,
    objectX,
    nCells )
```

Value:

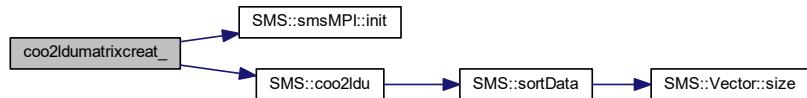
```
forAll(i, nCells) \
{ \
    x[i] = objectX[i]; \
}
```

7.66.2 Function Documentation

7.66.2.1 coo2ldumatrixcreat_()

```
void coo2ldumatrixcreat_ (
    long int * APtrPtr,
    const scalar * dataPtr,
    const label * fRowsPtr,
    const label * fColsPtr,
    const label * nCellsPtr,
    const label * sizePtr,
    const label * symmPtr )
```

Here is the call graph for this function:



7.66.2.2 csr2ldumatrixcreat_()

```
void csr2ldumatrixcreat_ (
    long int * APtrPtr,
    const scalar * dataPtr,
    const label * fRowsPtr,
    const label * fColsPtr,
    const label * nCellsPtr,
    const label * sizePtr,
    const label * symmPtr )
```

Here is the call graph for this function:



7.66.2.3 ldumatrixcreat_()

```

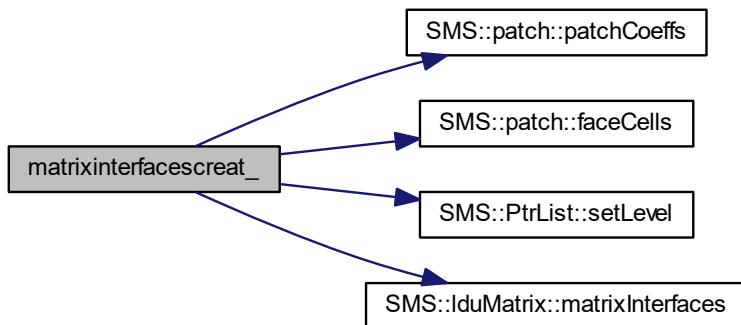
void ldumatrixcreat_ (
    long int * APtrPtr,
    label * nCellsPtr,
    label * upperSizePtr,
    label * lowerAddrValue,
    label * upperAddrValue,
    scalar * lowerValue,
    scalar * diagValue,
    scalar * upperValue )
  
```

7.66.2.4 matrixinterfacescreat_()

```

void matrixinterfacescreat_ (
    long int * APtrPtr,
    const label * nNeiProcsPtr,
    const label * destRankPtr,
    const label * locPositionPtr,
    const label * faceCellsPtr,
    const scalar * dataPtr )
  
```

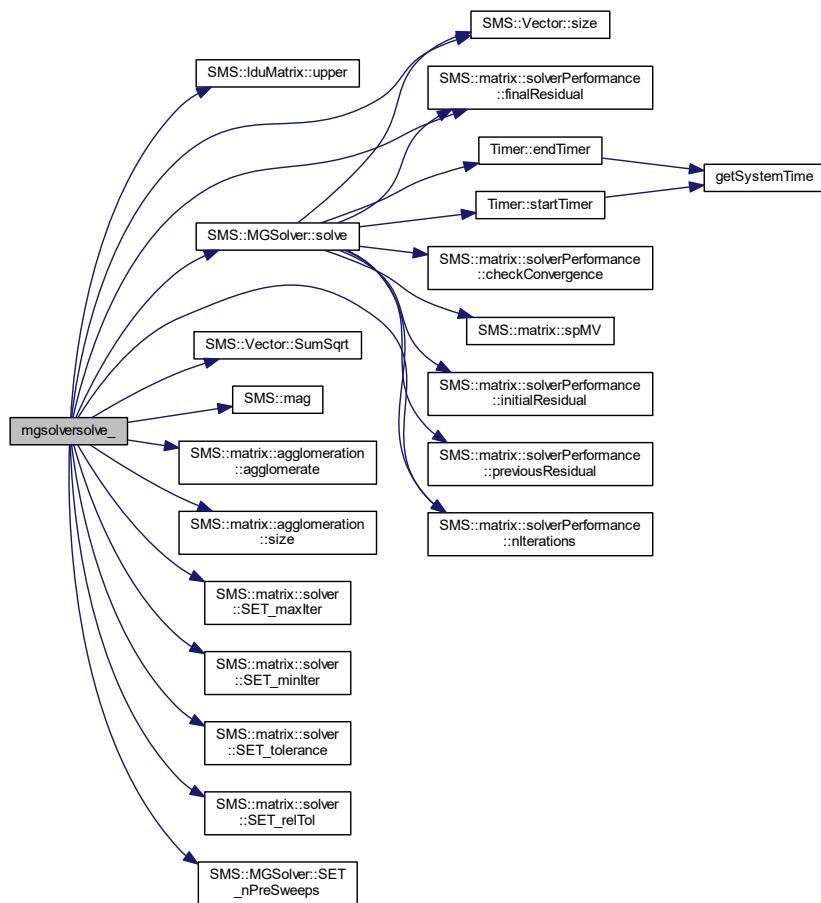
Here is the call graph for this function:



7.66.2.5 mgsolversolve_()

```
void mgsolversolve_ (
    scalar * xValue,
    long int * APtrPtr,
    scalar * bValue,
    label * nCellsPtr,
    label * agglTypePtr,
    label * smootherTypePtr,
    scalar * tolPtr,
    scalar * relTolPtr,
    label * maxIterPtr,
    label * minIterPtr,
    label * num_iterationsPtr,
    scalar * res_normPtr,
    scalar * faceAreaPtr )
```

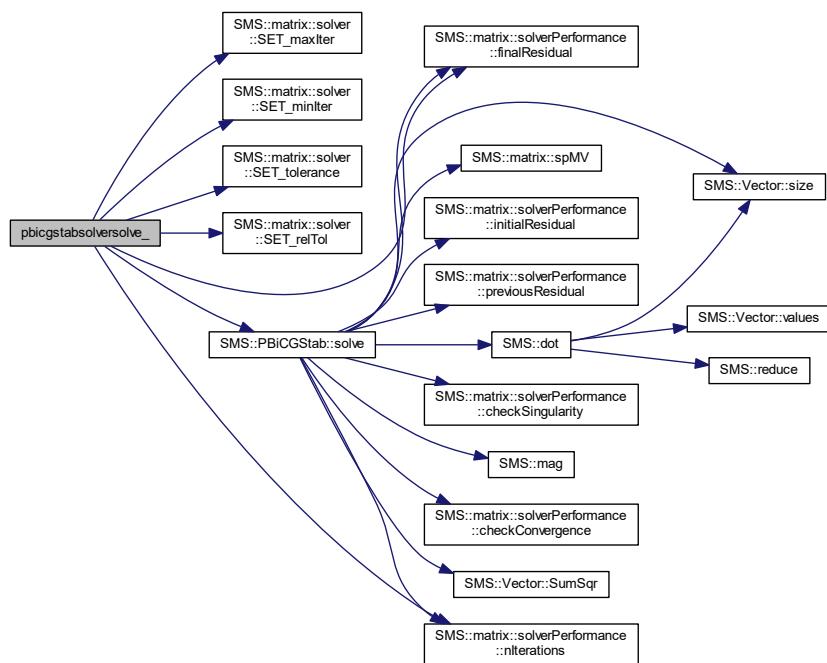
Here is the call graph for this function:



7.66.2.6 pbicgstabsolversolve_()

```
void pbicgstabsolversolve_ (
    scalar * xValue,
    long int * APtrPtr,
    scalar * bValue,
    label * nCellsPtr,
    label * precondTypePtr,
    scalar * tolPtr,
    scalar * relTolPtr,
    label * maxIterPtr,
    label * minIterPtr,
    label * num_iterationsPtr,
    scalar * res_normPtr )
```

Here is the call graph for this function:



7.66.2.7 pcgsolversolve_()

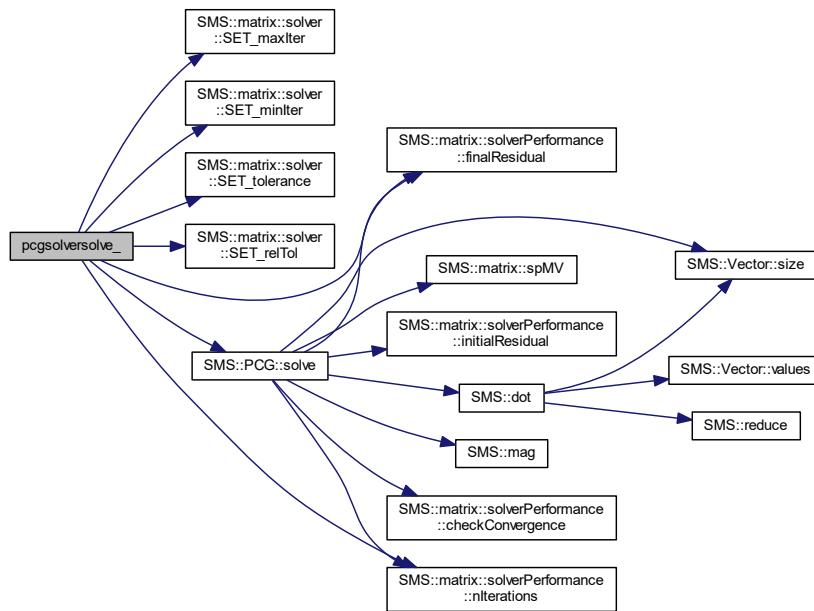
```
void pcgsolversolve_ (
    scalar * xValue,
    long int * APtrPtr,
    scalar * bValue,
    label * nCellsPtr,
    label * precondTypePtr,
    scalar * tolPtr,
    scalar * relTolPtr,
```

```

label * maxIterPtr,
label * minIterPtr,
label * num_iterationsPtr,
scalar * res_normPtr )

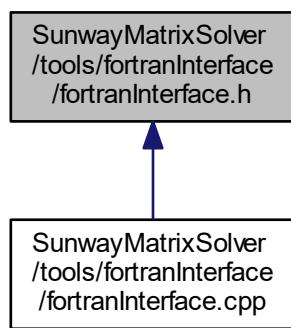
```

Here is the call graph for this function:



7.67 SunwayMatrixSolver/tools/fortranInterface/fortranInterface.h File Reference

This graph shows which files directly or indirectly include this file:



Functions

- void `ldumatrixcreat_` (long int *APtrPtr, int *nCells, int *upperSize, int *lowerAddr, int *upperAddr, double *lower, double *diag, double *upper)
- void `coo2ldumatrixcreat_` (long int *APtrPtr, const double *dataPtr, const int *rowsPtr, const int *columnPtr, const int *nCells, const int *size, const int *symm)
- void `csr2ldumatrixcreat_` (long int *APtrPtr, const double *dataPtr, const int *compRowsPtr, const int *columnPtr, const int *nCells, const int *size, const int *symm)
- void `matrixinterfacescreat_` (long int *APtrPtr, const int *nNeiProcs, const int *destRank, const int *locPosition, const int *faceCells, const double *data)
- void `pcgsolversolve_` (double *x, long int *APtrPtr, double *b, int *nCells, int *precond, double *tol, double *relTol, int *maxIter, int *minIter, int *num_iterations, double *final_res_norm)
- void `pbicgstabsversolve_` (double *x, long int *APtrPtr, double *b, int *nCells, int *precond, double *tol, double *relTol, int *maxIter, int *minIter, int *num_iterations, double *final_res_norm)
- void `mgsolversolve_` (double *x, long int *APtrPtr, double *b, int *nCells, int *aggl, int *smoother, double *tol, double *relTol, int *maxIter, int *minIter, int *num_iterations, double *final_res_norm, double *faceAreaPtr)

7.67.1 Function Documentation

7.67.1.1 `coo2ldumatrixcreat_()`

```
void coo2ldumatrixcreat_ (
    long int * APtrPtr,
    const double * dataPtr,
    const int * rowsPtr,
    const int * columnPtr,
    const int * nCells,
    const int * size,
    const int * symm )
```

7.67.1.2 `csr2ldumatrixcreat_()`

```
void csr2ldumatrixcreat_ (
    long int * APtrPtr,
    const double * dataPtr,
    const int * compRowsPtr,
    const int * columnPtr,
    const int * nCells,
    const int * size,
    const int * symm )
```

7.67.1.3 ldumatrixcreat_()

```
void ldumatrixcreat_ (
    long int * APtrPtr,
    int * nCells,
    int * upperSize,
    int * lowerAddr,
    int * upperAddr,
    double * lower,
    double * diag,
    double * upper )
```

7.67.1.4 matrixinterfacescreat_()

```
void matrixinterfacescreat_ (
    long int * APtrPtr,
    const int * nNeiProcs,
    const int * destRank,
    const int * locPosition,
    const int * faceCells,
    const double * data )
```

7.67.1.5 mgsolversolve_()

```
void mgsolversolve_ (
    double * x,
    long int * APtrPtr,
    double * b,
    int * nCells,
    int * aggl,
    int * smoother,
    double * tol,
    double * relTol,
    int * maxIter,
    int * minIter,
    int * num_iterations,
    double * final_res_norm,
    double * faceAreaPtr )
```

7.67.1.6 pbicgstabsolversolve_()

```
void pbicgstabsolversolve_ (
    double * x,
    long int * APtrPtr,
    double * b,
    int * nCells,
    int * precond,
```

```
    double * tol,
    double * relTol,
    int * maxIter,
    int * minIter,
    int * num_iterations,
    double * final_res_norm )
```

7.67.1.7 pcgsolversolve_()

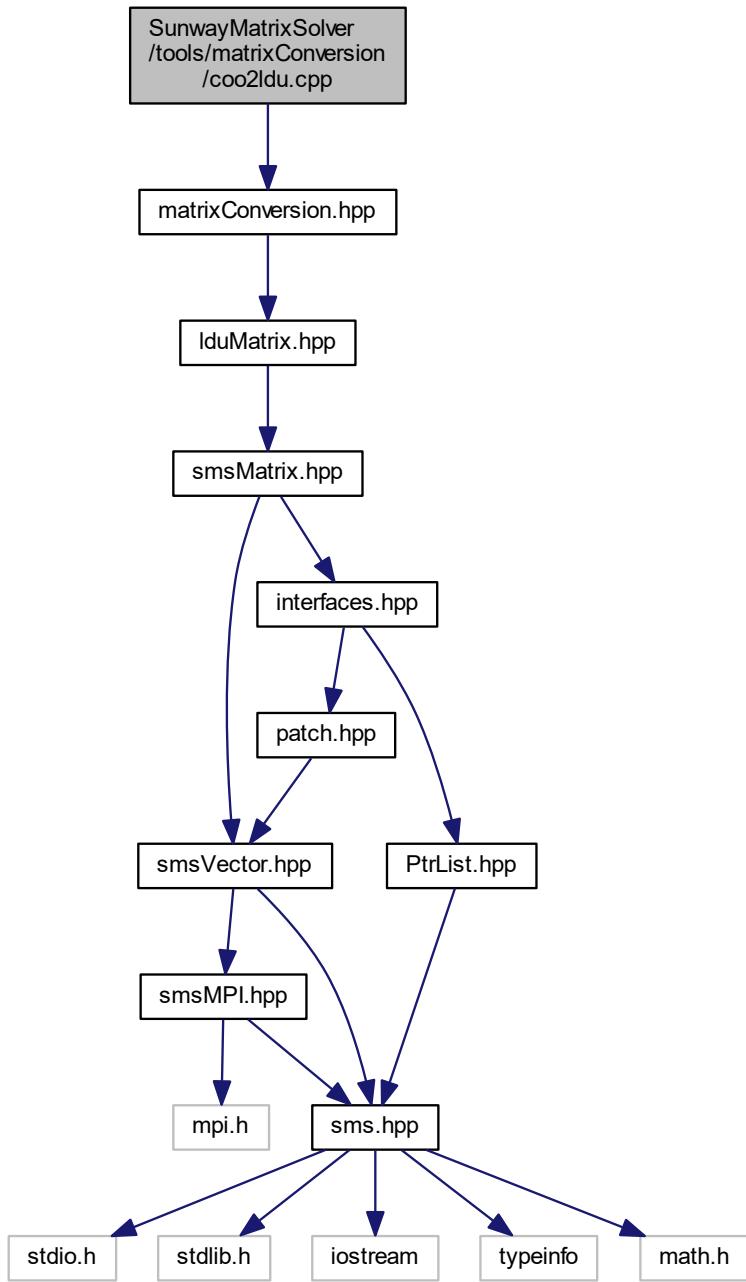
```
void pcgsolversolve_ (
    double * x,
    long int * APtrPtr,
    double * b,
    int * nCells,
    int * precond,
    double * tol,
    double * relTol,
    int * maxIter,
    int * minIter,
    int * num_iterations,
    double * final_res_norm )
```

7.68 SunwayMatrixSolver/tools/matrixConversion/coo2csr.cpp File Reference

7.69 SunwayMatrixSolver/tools/matrixConversion/coo2ldu.cpp File Reference

```
#include "matrixConversion.hpp"
```

Include dependency graph for coo2ldu.cpp:

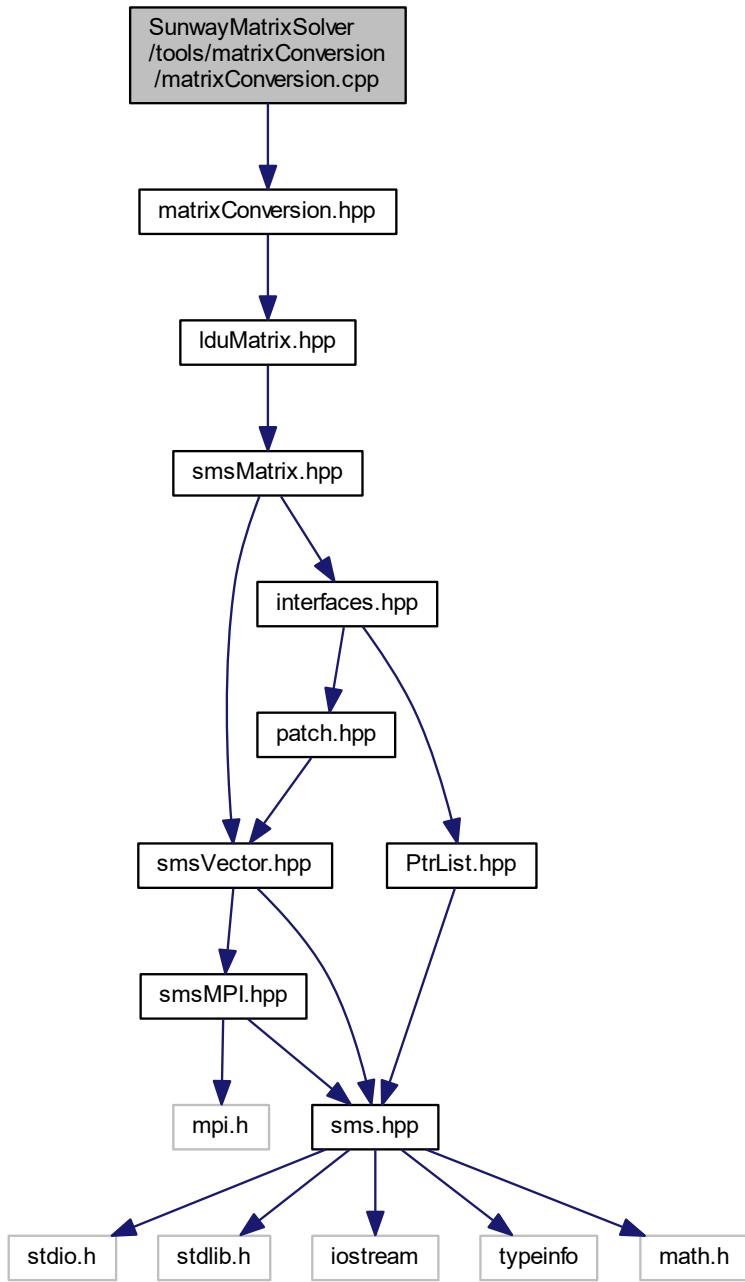


7.70 SunwayMatrixSolver/tools/matrixConversion/csr2coo.cpp File Reference

7.71 SunwayMatrixSolver/tools/matrixConversion/matrixConversion.cpp File Reference

```
#include "matrixConversion.hpp"
```

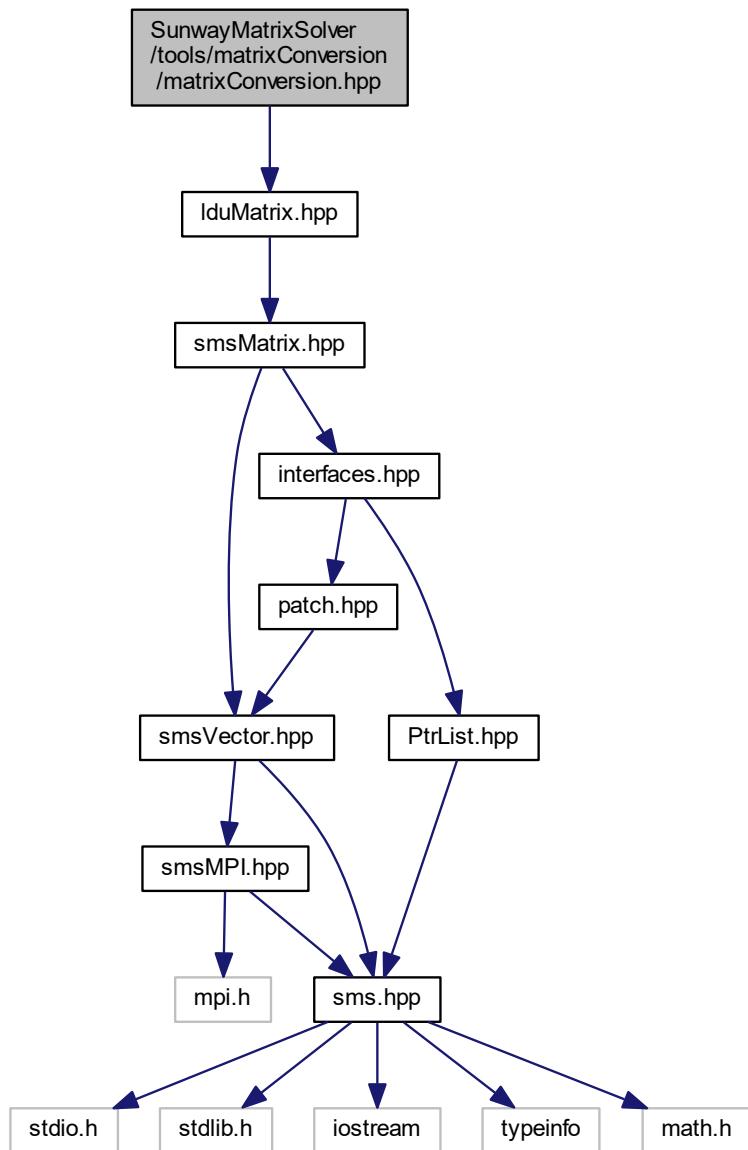
Include dependency graph for matrixConversion.cpp:



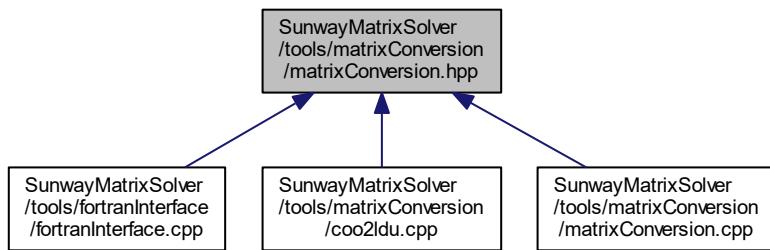
7.72 SunwayMatrixSolver/tools/matrixConversion/matrixConversion.hpp File Reference

```
#include "lduMatrix.hpp"
```

Include dependency graph for matrixConversion.hpp:



This graph shows which files directly or indirectly include this file:



Namespaces

- [SMS](#)

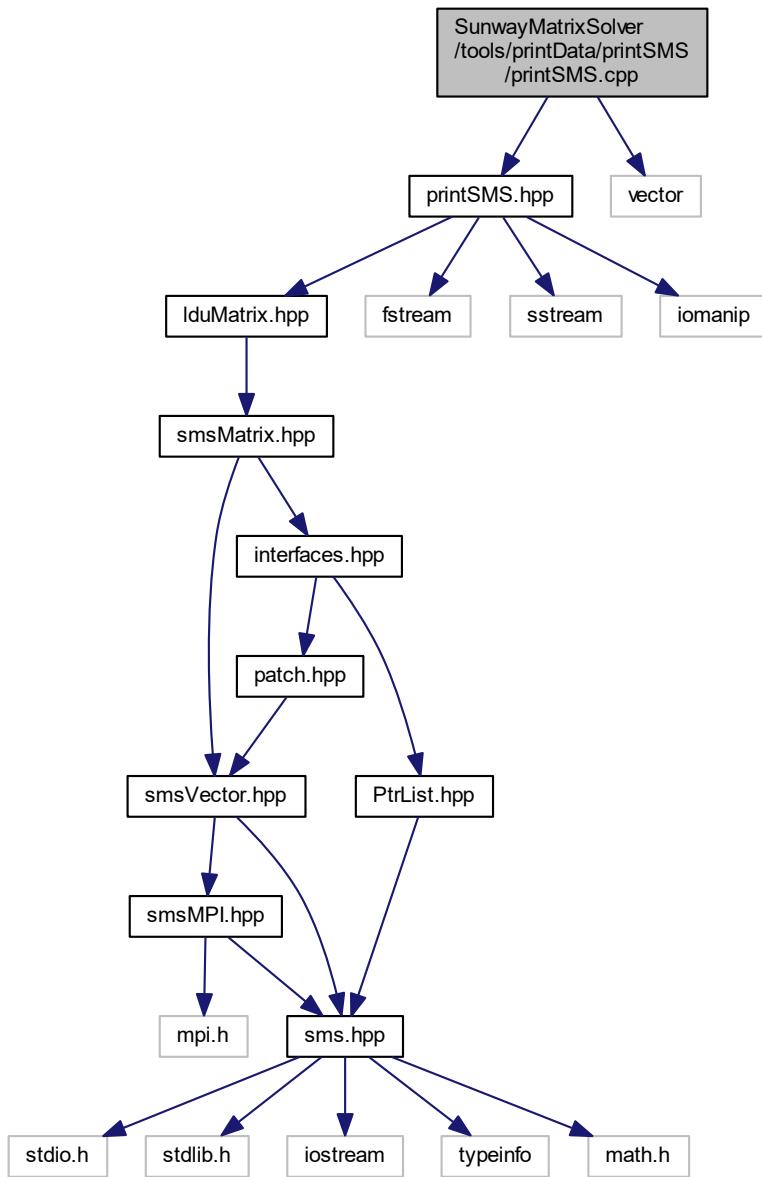
Functions

- void [`SMS::sortData`](#) (`scalarField &data, const labelField &order, const labelField &cellFaces`)
- `lduMatrix & SMS::coo2ldu` (`const scalar *dataPtr, const label *rowsPtr, const label *columnPtr, const label nCells, const label size, const bool symm`)
- `lduMatrix & SMS::csr2ldu` (`const scalar *dataPtr, const label *compRowsPtr, const label *columnPtr, const label nCells, const label size, const bool symm`)

7.73 SunwayMatrixSolver/tools/printData/printSMS/printSMS.cpp File Reference

```
#include "printSMS.hpp"
#include <vector>
```

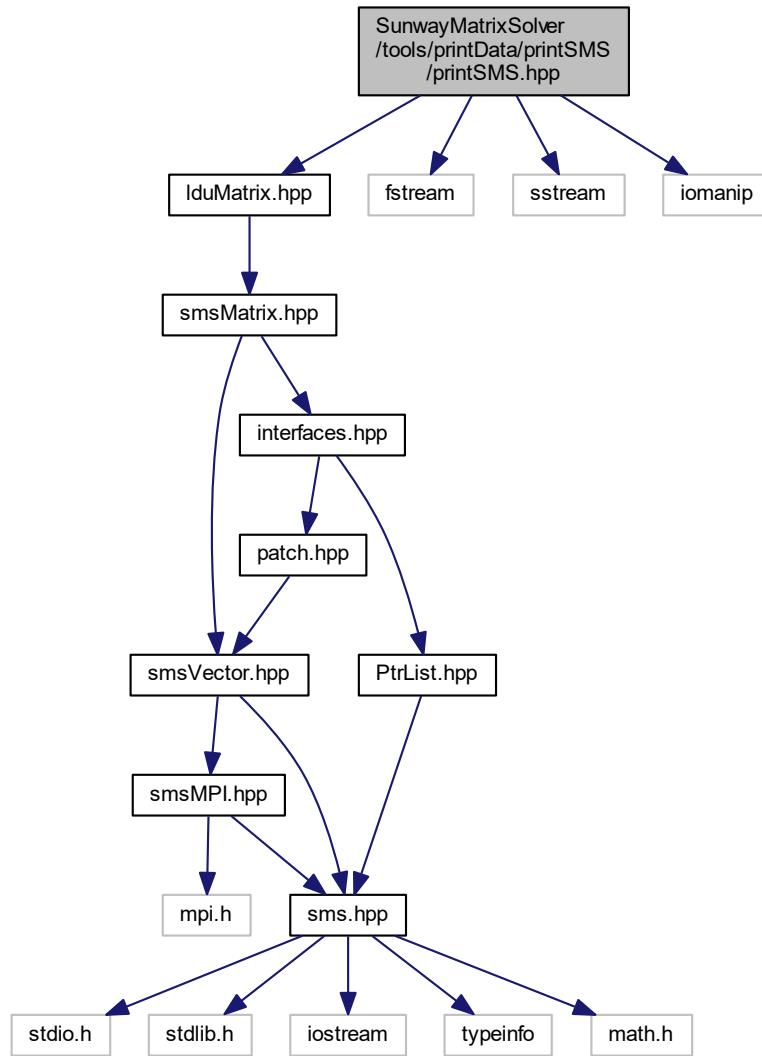
Include dependency graph for printSMS.cpp:



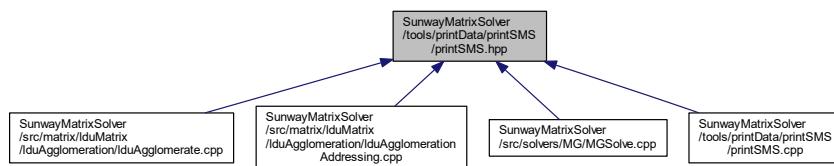
7.74 SunwayMatrixSolver/tools/printData/printSMS/printSMS.hpp File Reference

```
#include "lduMatrix.hpp"
#include <fstream>
#include <sstream>
#include <iomanip>
```

Include dependency graph for printSMS.hpp:



This graph shows which files directly or indirectly include this file:



Namespaces

- SMS

Macros

- #define FILEOPEN(fout, fileName)
- #define FILECLOSE(fout, fileName) fout.close()

Functions

- void SMS::printLDUMatrix (const lduMatrix &A, const char *name)
- template<typename T >
void SMS::printVector (const T &b, const char *name)
- void SMS::printInterfaces (const lduMatrix &A, const char *name)

7.74.1 Macro Definition Documentation

7.74.1.1 FILECLOSE

```
#define FILECLOSE(
    fout,
    fileName ) fout.close()
```

7.74.1.2 FILEOPEN

```
#define FILEOPEN(
    fout,
    fileName )
```

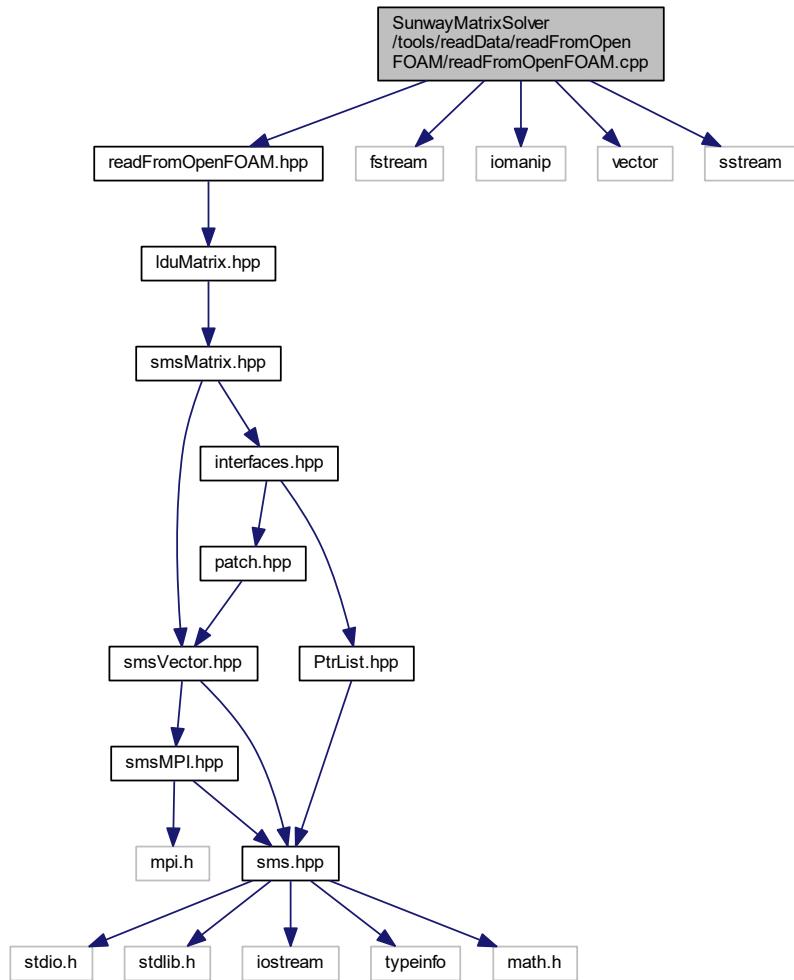
Value:

```
std::ostringstream os; \
os << fileName << "_" << SMS::MYID << ".txt"; \
fout.open(os.str().c_str())
```

7.75 SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.cpp File Reference

```
#include "readFromOpenFOAM.hpp"
#include <fstream>
#include <iomanip>
#include <vector>
```

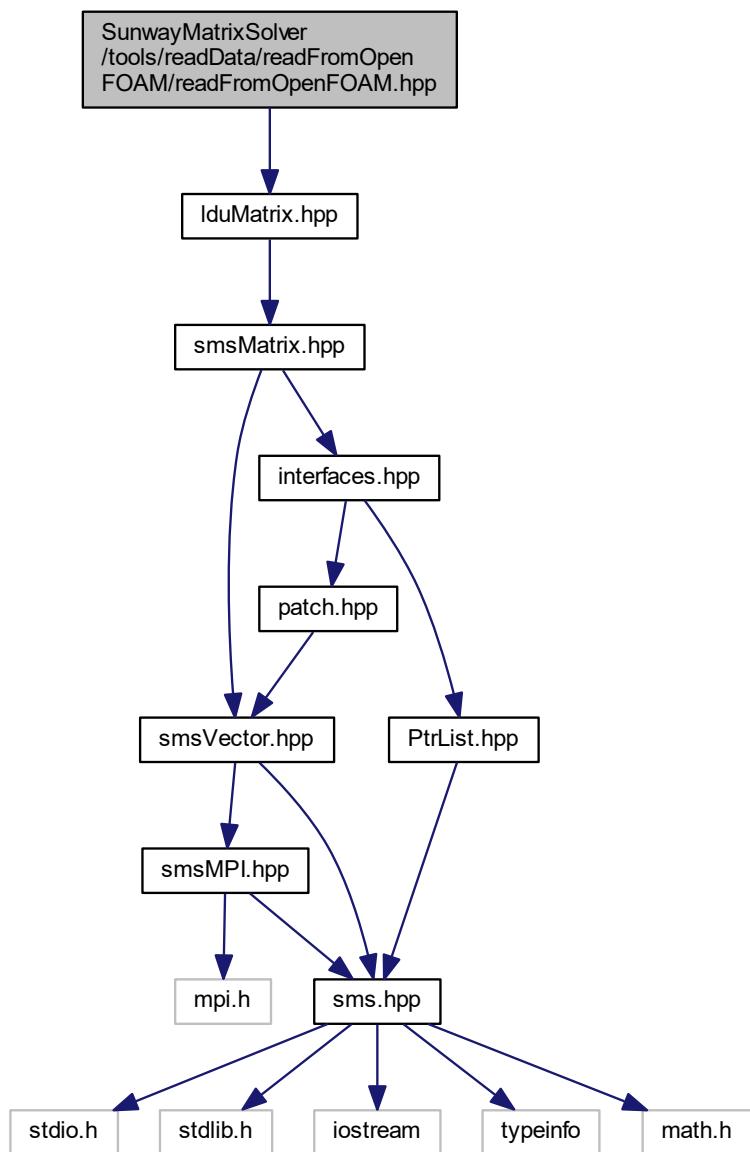
```
#include <sstream>
Include dependency graph for readFromOpenFOAM.cpp:
```



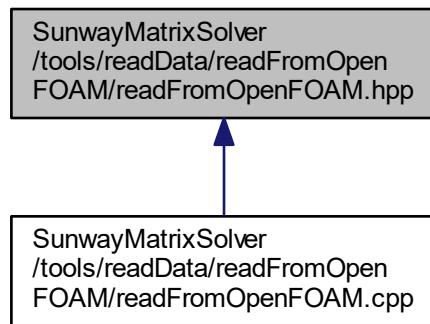
7.76 SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.hpp File Reference

```
#include "lduMatrix.hpp"
```

Include dependency graph for readFromOpenFOAM.hpp:



This graph shows which files directly or indirectly include this file:



Namespaces

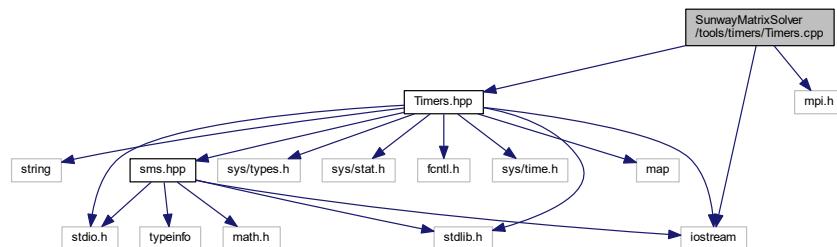
- [SMS](#)

Functions

- void [SMS::constructMatrixFromOpenFOAM](#) (lduMatrix &lDU, const char *fileName)
- void [SMS::constructVectorFromOpenFOAM](#) (scalarField &b, const char *fileName)
- void [SMS::constructInterfacesFromOpenFOAM](#) (lduMatrix &lDU, const char *fileName)

7.77 SunwayMatrixSolver/tools/timers/Timers.cpp File Reference

```
#include "Timers.hpp"
#include <iostream>
#include "mpi.h"
Include dependency graph for Timers.cpp:
```



Functions

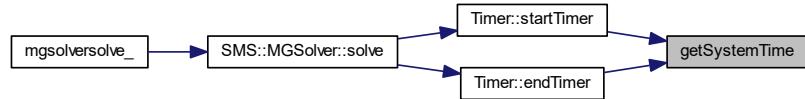
- double [getSystemTime \(\)](#)

7.77.1 Function Documentation

7.77.1.1 getSystemTime()

```
double getSystemTime ( ) [inline]
```

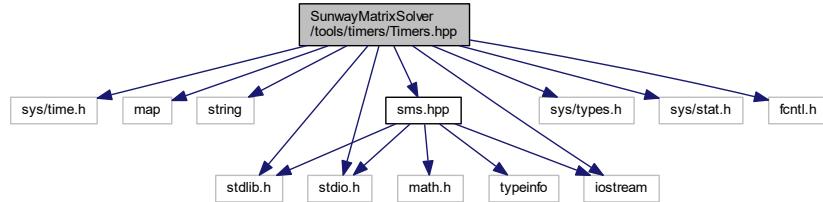
Here is the caller graph for this function:



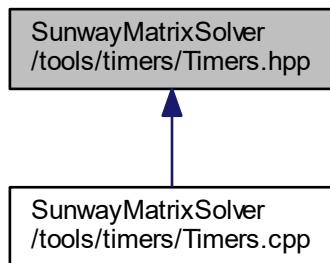
7.78 SunwayMatrixSolver/tools/timers/Timers.hpp File Reference

```
#include <sys/time.h>
#include <map>
#include <string>
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include "sms.hpp"
```

Include dependency graph for Timers.hpp:



This graph shows which files directly or indirectly include this file:



Classes

- class [Timer](#)

Index

_OpenTimer
 Timer, 136

~MGPrecond
 SMS::MGPrecond, 91

~MGSolver
 SMS::MGSolver, 94

~PBiCGStab
 SMS::PBiCGStab, 109

~PCG
 SMS::PCG, 112

~PtrList
 SMS::PtrList< T >, 116

~Timer
 Timer, 134

~Vector
 SMS::Vector< T >, 139

~agglomeration
 SMS::matrix::agglomeration, 26

~interfaces
 SMS::interfaces, 39

~lduAgglomeration
 SMS::lduAgglomeration, 47

~lduDICPrecond
 SMS::lduDICPrecond, 59

~lduDILUPrecond
 SMS::lduDILUPrecond, 63

~lduDiagPrecond
 SMS::lduDiagPrecond, 55

~lduGaussSeidelSmoothen
 SMS::lduGaussSeidelSmoothen, 66

~lduMatrix
 SMS::lduMatrix, 71

~matrix
 SMS::matrix, 87

~patch
 SMS::patch, 101

~preconditioner
 SMS::matrix::preconditioner, 114

~smoother
 SMS::matrix::smoother, 119

~solver
 SMS::matrix::solver, 124

agglomerate
 SMS::lduAgglomeration, 47, 48
 SMS::matrix::agglomeration, 26

agglomerateLduAddressing
 SMS::lduAgglomeration, 48

agglomerateMatrix
 SMS::lduAgglomeration, 49

 SMS::matrix::agglomeration, 26

agglomeration
 SMS::matrix::agglomeration, 26

agglomeration_
 SMS::MGSolver, 99

ALLOCATE_POINTER
 sms.hpp, 153

allocateSym2D
 SMS::eigenDiagPCG, 34

APtr_
 SMS::lduDICPrecond, 60
 SMS::lduDILUPrecond, 64
 SMS::MGPrecond, 92

begin
 SMS::Vector< T >, 140

boostFactorCheby_
 SMS::chebySmoothen, 32

cacheAgglomeration_
 SMS::MGSolver, 99

calcLosort
 SMS::lduMatrix, 72

calcLosortStart
 SMS::lduMatrix, 72

calcOwnerStart
 SMS::lduMatrix, 72

calcReciprocalD
 SMS::lduDICPrecond, 59
 SMS::lduDILUPrecond, 63

chebySmoothen
 SMS::chebySmoothen, 31

CHECK_POINTER
 sms.hpp, 153

checkConvergence
 SMS::matrix::solverPerformance, 129

checkSingularity
 SMS::matrix::solverPerformance, 129

closeTimer
 Timer, 134

coarseMatrixLevels
 SMS::lduAgglomeration, 50
 SMS::matrix::agglomeration, 26

coarseMatrixLevels_
 SMS::lduAgglomeration, 53

combineLevels
 SMS::lduAgglomeration, 51

compactLevels
 SMS::lduAgglomeration, 51

compassInterface.cpp

PTR2OBJ, 208
 computeMaxEig
 SMS::eigenDiagPCG, 34
 computeValueForMatrix
 SMS::eigenDiagPCG, 34
 constructInterfacesFromOpenFOAM
 SMS, 11
 constructMatrixFromOpenFOAM
 SMS, 11
 constructVectorFromOpenFOAM
 SMS, 12
 continueAgglomerating
 SMS::lduAgglomeration, 51
 contract_solver_mg_
 SMS, 12
 contract_solver_pbicgstab_
 SMS, 13
 contract_sw_matrix_
 SMS, 13
 contract_sw_matrix_interfaces_
 SMS, 13
 converged
 SMS::matrix::solverPerformance, 130
 converged_
 SMS::matrix::solverPerformance, 132
 coo2ldu
 SMS, 14
 coo2ldumatrixcreat_
 fortranInterface.cpp, 212
 fortranInterface.h, 217
 count
 Timer, 137
 COUT
 sms.hpp, 154
 csr2ldu
 SMS, 15
 csr2ldumatrixcreat_
 fortranInterface.cpp, 212
 fortranInterface.h, 217
 DELETE_OBJECT_POINTER
 sms.hpp, 154
 DELETE_POINTER
 sms.hpp, 154
 deletePrecondPtr_
 SMS::PBiCGStab, 110
 SMS::PCG, 113
 deleteSym2D
 SMS::eigenDiagPCG, 35
 destRank_
 SMS::interfaces, 41
 determineEigRange
 SMS::eigenDiagPCG, 35
 diag
 SMS::lduMatrix, 73
 SMS::matrix, 88
 diagPCGLoops
 SMS::eigenDiagPCG, 35
 diagPtr_
 SMS::lduMatrix, 85
 dot
 SMS, 15
 eigenDiagPCG
 SMS::eigenDiagPCG, 34
 eigFirstTimeComputed_
 SMS::chebySmoothen, 32
 eigRatioCheby_
 SMS::chebySmoothen, 32
 eigRatioCoarest_
 SMS::chebySmoothen, 32
 end
 SMS::Vector< T >, 140
 ENDL
 sms.hpp, 154
 endTimer
 Timer, 134
 ERROR_EXIT
 sms.hpp, 154, 155
 EXIT
 sms.hpp, 155
 exitMPI
 SMS::smsMPI, 121
 faceCells
 SMS::patch, 101, 102
 faceCellsPtr_
 SMS::patch, 106
 facel
 SMS::labelPair, 43
 facel_
 SMS::labelPair, 45
 faceRestrictAddressing
 SMS::lduAgglomeration, 51
 SMS::matrix::agglomeration, 27
 SMS::patch, 102, 103
 faceRestrictAddressing_
 SMS::lduAgglomeration, 53
 faceRestrictAddressingPtr_
 SMS::patch, 106
 FILECLOSE
 printSMS.hpp, 226
 FILEOPEN
 printSMS.hpp, 226
 fill_sw_matrix_coefficients_
 SMS, 16
 fill_sw_matrix_interfaces_coefficients_
 SMS, 16
 finalResidual
 SMS::matrix::solverPerformance, 130
 finalResidual_
 SMS::matrix::solverPerformance, 132
 finestMatrix_
 SMS::lduAgglomeration, 53
 SMS::MGSSolver, 99
 first
 SMS::labelPair, 44
 first_

SMS::labelPair, 45
forAll
 sms.hpp, 155
fortranInterface.cpp
 coo2ldumatrixcreat_, 212
 csr2ldumatrixcreat_, 212
 ldumatrixcreat_, 213
 matrixinterfacescreat_, 213
 mgsolversolve_, 213
 pbicgstabsolversolve_, 214
 pcgsolversolve_, 215
 RETURN_VALUE, 211
fortranInterface.h
 coo2ldumatrixcreat_, 217
 csr2ldumatrixcreat_, 217
 ldumatrixcreat_, 217
 matrixinterfacescreat_, 218
 mgsolversolve_, 218
 pbicgstabsolversolve_, 218
 pcgsolversolve_, 219
forward_
 SMS::lduAgglomeration, 53

getSystemTime
 Timers.cpp, 230
GREAT
 sms.hpp, 155

h14Sturm
 SMS::eigenDiagPCG, 36

init
 SMS::chebySmoothen, 31
 SMS::lduGaussSeidelSmoothen, 66
 SMS::matrix::smoother, 120
 SMS::smsMPI, 121
initialResidual
 SMS::matrix::solverPerformance, 131
initialResidual_
 SMS::matrix::solverPerformance, 133
initInterfaces
 SMS::lduMatrix, 73
 SMS::matrix, 88
initMatrixInterfaces
 SMS::interfaces, 39
initSmoothers
 SMS::MGSolver, 94
initVcycle
 SMS::MGSolver, 94
interfaces
 SMS::interfaces, 39
interfacesPtr_
 SMS::lduMatrix, 85
isEmpty
 SMS::PtrList< T >, 117

labelField
 SMS, 11
labelPair
 SMS::labelPair, 43
lduAgglomeration
 SMS::lduAgglomeration, 47
lduDiagPrecond
 SMS::lduDiagPrecond, 55
lduDICPrecond
 SMS::lduDICPrecond, 59
lduDILUPrecond
 SMS::lduDILUPrecond, 63
lduGaussSeidelSmoothen
 SMS::lduGaussSeidelSmoothen, 65
lduMatrix
 SMS::lduMatrix, 71
ldumatrixcreat_
 fortranInterface.cpp, 213
 fortranInterface.h, 217
length_
 SMS::Vector< T >, 146
locPosition_
 SMS::interfaces, 42
losortAddr
 SMS::lduMatrix, 74
losortPtr_
 SMS::lduMatrix, 85
losortStartAddr
 SMS::lduMatrix, 74
losortStartPtr_
 SMS::lduMatrix, 85
lower
 SMS::lduMatrix, 74
lowerAddr
 SMS::lduMatrix, 75
lowerAddrPtr_
 SMS::lduMatrix, 86
lowerPtr_
 SMS::lduMatrix, 86
mag
 SMS, 17
matrixInterfaces
 SMS::lduMatrix, 75, 76
 SMS::matrix, 88
matrixinterfacescreat_
 fortranInterface.cpp, 213
 fortranInterface.h, 218
matrixLevel
 SMS::lduAgglomeration, 51
max
 SMS, 17
maxEigenValue
 SMS::eigenDiagPCG, 36
maxEigenValue_
 SMS::eigenDiagPCG, 37
maxEigPCG_
 SMS::chebySmoothen, 33
maxIter
 SMS::matrix::solver, 124
maxIter_
 SMS::matrix::solver, 127

maxLevels_
 SMS::lduAgglomeration, 53
 maxTimeSum
 Timer, 135
 mergeLevels_
 SMS::lduAgglomeration, 53
 MGPrecond
 SMS::MGPrecond, 91
 MGSolver
 SMS::MGSolver, 94
 MGSolverPtr_
 SMS::MGPrecond, 92
 mgsolversolve_
 fortranInterface.cpp, 213
 fortranInterface.h, 218
 min
 SMS, 18
 minlter
 SMS::matrix::solver, 124
 minlter_
 SMS::matrix::solver, 127
 MPI_LABEL
 smsMPI.hpp, 149
 MPI_SCALAR
 smsMPI.hpp, 149
 MYID
 smsMPI.hpp, 149
 myProcNo
 SMS::patch, 103
 SMS::smsMPI, 121
 myProcNo_
 SMS::patch, 106
 SMS::smsMPI, 122
 nCells
 SMS::lduMatrix, 76
 nCells_
 SMS::lduAgglomeration, 53
 SMS::lduMatrix, 86
 nCellsInCoarsestLevel_
 SMS::lduAgglomeration, 53
 nCreatedLevels_
 SMS::lduAgglomeration, 54
 nDiagPCGs_
 SMS::chebySmoother, 33
 neighbProcNo
 SMS::patch, 103, 104
 neighbProcNo_
 SMS::patch, 107
 nFinestSweeps_
 SMS::MGSolver, 99
 nIterations
 SMS::matrix::solverPerformance, 131
 nIterations_
 SMS::matrix::solverPerformance, 133
 normFactor
 SMS::matrix::solver, 124
 nPostSweeps_
 SMS::MGSolver, 99
 nPreSweeps_
 SMS::MGSolver, 99
 NPROCS
 smsMPI.hpp, 149
 nProcs
 SMS::smsMPI, 121
 nProcs_
 SMS::smsMPI, 122
 nVcycles_
 SMS::MGPrecond, 92
 openTimer
 Timer, 135
 operator *
 SMS::Vector< T >, 141
 operator *=
 SMS::Vector< T >, 141
 operator+
 SMS::Vector< T >, 141
 operator+=
 SMS::Vector< T >, 141
 operator-
 SMS::Vector< T >, 141
 operator-=
 SMS::Vector< T >, 142
 operator=
 SMS::Vector< T >, 142
 operator==
 SMS::labelPair, 44
 operator[]
 SMS::PtrList< T >, 117
 SMS::Vector< T >, 142
 ownerStartAddr
 SMS::lduMatrix, 76
 ownerStartPtr_
 SMS::lduMatrix, 86
 PARRUN
 smsMPI.hpp, 149
 parRun
 SMS::smsMPI, 121
 parRun_
 SMS::smsMPI, 122
 patch
 SMS::patch, 101
 patchCoeffs
 SMS::patch, 104, 105
 patchCoeffsPtr_
 SMS::patch, 107
 patches_
 SMS::interfaces, 42
 patchFaceRestrictAddressing_
 SMS::lduAgglomeration, 54
 patchList
 SMS::interfaces, 39, 40
 PBiCGStab
 SMS::PBiCGStab, 108
 pbicgstabsolversolve_
 fortranInterface.cpp, 214

fortranInterface.h, 218
PCG
 SMS::PCG, 112
pcgsolversolve_
 fortranInterface.cpp, 215
 fortranInterface.h, 219
precondition
 SMS::lduDiagPrecond, 56
 SMS::lduDICPrecond, 59
 SMS::lduDILUPrecond, 63
 SMS::matrix::preconditioner, 114
 SMS::MGPrecond, 91
precondPtr_
 SMS::PBiCGStab, 110
 SMS::PCG, 114
preSmoothUsing_
 SMS::chebySmoothen, 33
previousResidual
 SMS::matrix::solverPerformance, 132
previousResidual_
 SMS::matrix::solverPerformance, 133
printInterfaces
 SMS, 18
printLDUMatrix
 SMS, 19
printSMS.hpp
 FILECLOSE, 226
 FILEOPEN, 226
printTimer
 Timer, 135
printVector
 SMS, 20
prolongField
 SMS::matrix::agglomeration, 27
PTR2OBJ
 compassInterface.cpp, 208
PtrList
 SMS::PtrList< T >, 116
ptrs_
 SMS::PtrList< T >, 118
rD_
 SMS::lduDiagPrecond, 56
 SMS::lduDICPrecond, 60
 SMS::lduDILUPrecond, 64
recvBuffer_
 SMS::interfaces, 42
reduce
 SMS, 20
relTol
 SMS::matrix::solver, 124
relTol_
 SMS::matrix::solver, 127
removeLevel
 SMS::PtrList< T >, 117
restrictAddressing
 SMS::lduAgglomeration, 52
 SMS::matrix::agglomeration, 27
restrictAddressing_
 SMS::lduAgglomeration, 54
restrictFaceField
 SMS::matrix::agglomeration, 27
restrictField
 SMS::matrix::agglomeration, 28
RETURN_VALUE
 fortranInterface.cpp, 211
scalarField
 SMS, 11
scaleCorrection_
 SMS::MGSolver, 99
scalingFactor
 SMS::MGSolver, 95
second
 SMS::labelPair, 44, 45
second_
 SMS::labelPair, 45
sendBuffer_
 SMS::interfaces, 42
sendRecvRequests_
 SMS::interfaces, 42
SET_cacheAgglomeration
 SMS::MGSolver, 95
SET_diag
 SMS::lduMatrix, 77
SET_lower
 SMS::lduMatrix, 77, 78
SET_lowerAddr
 SMS::lduMatrix, 78
SET_maxIter
 SMS::matrix::solver, 125
SET_maxLevels
 SMS::lduAgglomeration, 52
 SMS::matrix::agglomeration, 28
SET_minIter
 SMS::matrix::solver, 125
SET_nCellsInCoarsestLevel
 SMS::lduAgglomeration, 52
 SMS::matrix::agglomeration, 29
SET_nFinestSweeps
 SMS::MGSolver, 96
SET_nPostSweeps
 SMS::MGSolver, 96
SET_nPreSweeps
 SMS::MGSolver, 96
set_nVcycles
 SMS::MGPrecond, 92
SET_preconditioner
 SMS::PBiCGStab, 109
SET_relTol
 SMS::matrix::solver, 125
SET_scaleCorrection
 SMS::MGSolver, 96
SET_size
 SMS::PtrList< T >, 117
 SMS::Vector< T >, 143
SET_tolerance
 SMS::matrix::solver, 126

SET_upper
 SMS::lduMatrix, 78, 79
 SET_upperAddr
 SMS::lduMatrix, 79
 SET_zero
 SMS::Vector< T >, 143
 setLevel
 SMS::PtrList< T >, 117
 setMatrixCoefficients
 SMS::lduMatrix, 79, 80
 setMatrixTopology
 SMS::lduMatrix, 80, 81
 setSymm
 SMS::lduMatrix, 81
 singular
 SMS::matrix::solverPerformance, 132
 singular_
 SMS::matrix::solverPerformance, 133
 size
 SMS::interfaces, 40
 SMS::lduAgglomeration, 52
 SMS::lduMatrix, 81, 82
 SMS::matrix, 88
 SMS::matrix::agglomeration, 29
 SMS::patch, 105, 106
 SMS::PtrList< T >, 118
 SMS::Vector< T >, 143
 size_
 SMS::patch, 107
 SMS::PtrList< T >, 118
 sleep
 SMS, 21
 SMALL
 sms.hpp, 155
 smooth
 SMS::chebySmoother, 31
 SMS::lduGaussSeidelSmoother, 66
 SMS::matrix::smoother, 120
 smoother
 SMS::matrix::smoother, 119
 smoothers_
 SMS::MGSolver, 99
 SMS, 9
 constructInterfacesFromOpenFOAM, 11
 constructMatrixFromOpenFOAM, 11
 constructVectorFromOpenFOAM, 12
 contract_solver_mg_, 12
 contract_solver_pbicgstab_, 13
 contract_sw_matrix_, 13
 contract_sw_matrix_interfaces_, 13
 coo2ldu, 14
 csr2ldu, 15
 dot, 15
 fill_sw_matrix_coefficients_, 16
 fill_sw_matrix_interfaces_coefficients_, 16
 labelField, 11
 mag, 17
 max, 17
 min, 18
 printInterfaces, 18
 printLDUMatrix, 19
 printVector, 20
 reduce, 20
 scalarField, 11
 sleep, 21
 sortData, 21
 sw_matrix_destroy_, 21
 sw_solve_mg_, 22
 sw_solve_pbicgstab_, 22
 sw_solver_destroy_mg_, 22
 sw_solver_destroy_pbicgstab_, 22
 sw_solver_mg_set_maxiter_, 22
 sw_solver_mg_set_miniter_, 22
 sw_solver_mg_set_nfinesweeps_, 23
 sw_solver_mg_set_npostsweeps_, 23
 sw_solver_mg_set_npresweeps_, 23
 sw_solver_mg_set_reltol_, 23
 sw_solver_mg_set_tol_, 23
 sw_solver_pbicgstab_set_maxiter_, 23
 sw_solver_pbicgstab_set_miniter_, 24
 sw_solver_pbicgstab_set_precond_, 24
 sw_solver_pbicgstab_set_reltol_, 24
 sw_solver_pbicgstab_set_tol_, 24
 sms.hpp
 ALLOCATE_POINTER, 153
 CHECK_POINTER, 153
 COUT, 154
 DELETE_OBJECT_POINTER, 154
 DELETE_POINTER, 154
 ENDL, 154
 ERROR_EXIT, 154, 155
 EXIT, 155
 forAll, 155
 GREAT, 155
 SMALL, 155
 VGREAT, 155
 VSMALL, 155
 SMS::chebySmoother, 30
 boostFactorCheby_, 32
 chebySmoother, 31
 eigFirstTimeComputed_, 32
 eigRatioCheby_, 32
 eigRatioCoarest_, 32
 init, 31
 maxEigPCG_, 33
 nDiagPCGs_, 33
 preSmoothUsing_, 33
 smooth, 31
 SMS::eigenDiagPCG, 33
 allocateSym2D, 34
 computeMaxEig, 34
 computeValueForMatrix, 34
 deleteSym2D, 35
 determineEigRange, 35
 diagPCGLoops, 35
 eigenDiagPCG, 34

h14Sturm, 36
maxEigenValue, 36
maxEigenValue_, 37
SMS::interfaces, 37
~interfaces, 39
destRank_, 41
initMatrixInterfaces, 39
interfaces, 39
locPosition_, 42
patches_, 42
patchList, 39, 40
recvBuffer_, 42
sendBuffer_, 42
sendRecvRequests_, 42
size, 40
updateMatrixInterfaces, 41
SMS::labelPair, 42
faceI, 43
faceI_, 45
first, 44
first_, 45
labelPair, 43
operator==, 44
second, 44, 45
second_, 45
SMS::lduAgglomeration, 46
~lduAgglomeration, 47
agglomerate, 47, 48
agglomerateLduAddressing, 48
agglomerateMatrix, 49
coarseMatrixLevels, 50
coarseMatrixLevels_, 53
combineLevels, 51
compactLevels, 51
continueAgglomerating, 51
faceRestrictAddressing, 51
faceRestrictAddressing_, 53
finestMatrix_, 53
forward_, 53
lduAgglomeration, 47
matrixLevel, 51
maxLevels_, 53
mergeLevels_, 53
nCells_, 53
nCellsInCoarsestLevel_, 53
nCreatedLevels_, 54
patchFaceRestrictAddressing_, 54
restrictAddressing, 52
restrictAddressing_, 54
SET_maxLevels, 52
SET_nCellsInCoarsestLevel, 52
size, 52
SMS::lduDiagPrecond, 54
~lduDiagPrecond, 55
lduDiagPrecond, 55
precondition, 56
rD_, 56
SMS::lduDICPrecond, 57
~lduDICPrecond, 59
APtr_, 60
calcReciprocalD, 59
lduDICPrecond, 59
precondition, 59
rD_, 60
SMS::lduDILUPrecond, 61
~lduDILUPrecond, 63
APtr_, 64
calcReciprocalD, 63
lduDILUPrecond, 63
precondition, 63
rD_, 64
SMS::lduGaussSeidelSmoothen, 65
~lduGaussSeidelSmoothen, 66
init, 66
lduGaussSeidelSmoothen, 65
smooth, 66
SMS::lduMatrix, 67
~lduMatrix, 71
calcLsosort, 72
calcLsosortStart, 72
calcOwnerStart, 72
diag, 73
diagPtr_, 85
initInterfaces, 73
interfacesPtr_, 85
lduMatrix, 71
lsosortAddr, 74
lsosortPtr_, 85
lsosortStartAddr, 74
lsosortStartPtr_, 85
lower, 74
lowerAddr, 75
lowerAddrPtr_, 86
lowerPtr_, 86
matrixInterfaces, 75, 76
nCells, 76
nCells_, 86
ownerStartAddr, 76
ownerStartPtr_, 86
SET_diag, 77
SET_lower, 77, 78
SET_lowerAddr, 78
SET_upper, 78, 79
SET_upperAddr, 79
setMatrixCoefficients, 79, 80
setMatrixTopology, 80, 81
setSymm, 81
size, 81, 82
spMV, 82
symm, 83
updateInterfaces, 83
upper, 84
upperAddr, 84
upperAddrPtr_, 86
upperPtr_, 86
SMS::matrix, 87

~matrix, 87
 diag, 88
 initInterfaces, 88
 matrixInterfaces, 88
 size, 88
 spMV, 88
 symm, 89
 updateInterfaces, 89
SMS::matrix::agglomeration, 25
 ~agglomeration, 26
 agglomerate, 26
 agglomerateMatrix, 26
 agglomeration, 26
 coarseMatrixLevels, 26
 faceRestrictAddressing, 27
 prolongField, 27
 restrictAddressing, 27
 restrictFaceField, 27
 restrictField, 28
 SET_maxLevels, 28
 SET_nCellsInCoarsestLevel, 29
 size, 29
SMS::matrix::preconditioner, 114
 ~preconditioner, 114
 precondition, 114
SMS::matrix::smoother, 119
 ~smoother, 119
 init, 120
 smooth, 120
 smoother, 119
SMS::matrix::solver, 123
 ~solver, 124
 maxIter, 124
 maxIter_, 127
 minIter, 124
 minIter_, 127
 normFactor, 124
 relTol, 124
 relTol_, 127
 SET_maxIter, 125
 SET_minIter, 125
 SET_relTol, 125
 SET_tolerance, 126
 solve, 126
 solver, 124
 tolerance, 127
 tolerance_, 127
SMS::matrix::solverPerformance, 128
 checkConvergence, 129
 checkSingularity, 129
 converged, 130
 converged_, 132
 finalResidual, 130
 finalResidual_, 132
 initialResidual, 131
 initialResidual_, 133
 nIterations, 131
 nIterations_, 133
 previousResidual, 132
 previousResidual_, 133
 singular, 132
 singular_, 133
 solverPerformance, 128
SMS::MGPrecond, 90
 ~MGPrecond, 91
 APtr_, 92
 MGPrecond, 91
 MGSolverPtr_, 92
 nVcycles_, 92
 precondition, 91
 set_nVcycles, 92
SMS::MGSolver, 93
 ~MGSolver, 94
 agglomeration_, 99
 cacheAgglomeration_, 99
 finestMatrix_, 99
 initSmoothers, 94
 initVcycle, 94
 MGsolver, 94
 nFinestSweeps_, 99
 nPostSweeps_, 99
 nPreSweeps_, 99
 scaleCorrection_, 99
 scalingFactor, 95
 SET_cacheAgglomeration, 95
 SET_nFinestSweeps, 96
 SET_nPostSweeps, 96
 SET_nPreSweeps, 96
 SET_scaleCorrection, 96
 smoothers_, 99
 solve, 96
 solveCoarsestLevel, 97
 Vcycle, 98
SMS::patch, 100
 ~patch, 101
 faceCells, 101, 102
 faceCellsPtr_, 106
 faceRestrictAddressing, 102, 103
 faceRestrictAddressingPtr_, 106
 myProcNo, 103
 myProcNo_, 106
 neighbProcNo, 103, 104
 neighbProcNo_, 107
 patch, 101
 patchCoeffs, 104, 105
 patchCoeffsPtr_, 107
 size, 105, 106
 size_, 107
SMS::PBiCGStab, 107
 ~PBiCGStab, 109
 deletePrecondPtr_, 110
 PBiCGStab, 108
 precondPtr_, 110
 SET_preconditioner, 109
 solve, 109
SMS::PCG, 111

~PCG, 112
deletePrecondPtr_, 113
PCG, 112
precondPtr_, 114
solve, 112
SMS::PtrList< T >, 115
~PtrList, 116
isEmpty, 117
operator[], 117
PtrList, 116
ptrs_, 118
removeLevel, 117
SET_size, 117
setLevel, 117
size, 118
size_, 118
SMS::smsMPI, 120
exitMPI, 121
init, 121
myProcNo, 121
myProcNo_, 122
nProcs, 121
nProcs_, 122
parRun, 121
parRun_, 122
smsLabel, 122
smsLabel_, 122
smsScalar, 122
smsScalar_, 122
SMS::Vector< T >, 137
~Vector, 139
begin, 140
end, 140
length_, 146
operator *, 141
operator *=, 141
operator+, 141
operator+=, 141
operator-, 141
operator-=, 142
operator=, 142
operator[], 142
SET_size, 143
SET_zero, 143
size, 143
Sum, 144
SumMag, 145
SumSqr, 145
SumSqrt, 145
values, 145
values_, 146
Vector, 138, 139
smsLabel
SMS::smsMPI, 122
smsLabel_
SMS::smsMPI, 122
smsMPI.hpp
MPI_LABEL, 149
MPI_SCALAR, 149
MYID, 149
NPROCS, 149
PARRUN, 149
smsScalar
SMS::smsMPI, 122
smsScalar_
SMS::smsMPI, 122
solve
SMS::matrix::solver, 126
SMS::MGSolver, 96
SMS::PBICGSTab, 109
SMS::PCG, 112
solveCoarsestLevel
SMS::MGSolver, 97
solver
SMS::matrix::solver, 124
solverPerformance
SMS::matrix::solverPerformance, 128
sortData
SMS, 21
spMV
SMS::lduMatrix, 82
SMS::matrix, 88
startTimer
Timer, 136
Sum
SMS::Vector< T >, 144
SumMag
SMS::Vector< T >, 145
SumSqr
SMS::Vector< T >, 145
SumSqrt
SMS::Vector< T >, 145
SunwayMatrixSolver/src/base/mpi/smsMPI.cpp, 147
SunwayMatrixSolver/src/base/mpi/smsMPI.hpp, 147
SunwayMatrixSolver/src/base/PtrList/PtrList.hpp, 150
SunwayMatrixSolver/src/base/sms.cpp, 151
SunwayMatrixSolver/src/base/sms.hpp, 152
SunwayMatrixSolver/src/base/vectors/smsVector.hpp,
156
SunwayMatrixSolver/src/base/vectors/smsVectorOperations.cpp,
157
SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/csrAgglomeration.hpp,
158
SunwayMatrixSolver/src/matrix/csrMatrix/csrAgglomeration/csrAgglomeration.cpp,
158
SunwayMatrixSolver/src/matrix/csrMatrix/csrMatrix.cpp,
158
SunwayMatrixSolver/src/matrix/csrMatrix/csrMatrix.hpp,
158
SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/eigenDiagPCG.hpp,
159
SunwayMatrixSolver/src/matrix/eigenvalue/eigenDiagPCG/eigenDiagPCG.cpp,
159
SunwayMatrixSolver/src/matrix/interfaces/interfaces.hpp,
161

SunwayMatrixSolver/src/matrix/interfaces/interfaces.hpp,	SunwayMatrixSolver/src/preconditioners/lduPreconditioners/lduDILU/lduDILU.h	162
SunwayMatrixSolver/src/matrix/interfaces/labelPair/labelPair.h,	SunwayMatrixSolver/src/preconditioners/MGPreconditioner/MGPreconditioner.h	164
SunwayMatrixSolver/src/matrix/interfaces/labelPair/labelPair.h,	SunwayMatrixSolver/src/preconditioners/MGPreconditioner/MGPreconditioner.h	164
SunwayMatrixSolver/src/matrix/interfaces/patch/patch.cpp,	SunwayMatrixSolver/src/smoothers/chebySmoothers/chebySmoothers.h	165
SunwayMatrixSolver/src/matrix/interfaces/patch/patch.hpp,	SunwayMatrixSolver/src/smoothers/chebySmoothers/chebySmoothers.h	192
SunwayMatrixSolver/src/matrix/interfaces/patch/patch.h,	SunwayMatrixSolver/src/smoothers/chebySmoothers/chebySmoothers.h	193
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmooth.h	167
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmooth.h	195
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/smoothers/csrSmoothers/csrGaussSeidelSmooth.h	195
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/smoothers/lduSmoothers/lduGaussSeidelSmooth.h	195
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/smoothers/lduSmoothers/lduGaussSeidelSmooth.h	195
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/smoothers/lduSmoothers/lduGaussSeidelSmooth.h	196
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/scalingFactor.h	198
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/solvers/MG/MG.h	199
SunwayMatrixSolver/src/matrix/lduMatrix/lduAgglomeration.h,	SunwayMatrixSolver/src/solvers/MG/solveCoarsestLevel.cpp	200
SunwayMatrixSolver/src/matrix/lduMatrix/lduMatrix.cpp,	SunwayMatrixSolver/src/solvers/MG/scalingFactor.cpp	201
SunwayMatrixSolver/src/matrix/lduMatrix/lduMatrix.hpp,	SunwayMatrixSolver/src/solvers/MG/solveCoarsestLevel.cpp	201
SunwayMatrixSolver/src/matrix/lduMatrix/lduMatrixSpMV.cpp,	SunwayMatrixSolver/src/solvers/MG/Vcycle.cpp	202
SunwayMatrixSolver/src/matrix/smsMatrix.hpp,	SunwayMatrixSolver/src/solvers/PBiCGStab/PBiCGStab.h	203
SunwayMatrixSolver/src/matrix/smsMatrixAgglomeration.cpp,	SunwayMatrixSolver/src/solvers/PBiCGStab/PBiCGStab.h	204
SunwayMatrixSolver/src/matrix/smsMatrixPrecond.cpp,	SunwayMatrixSolver/src/solvers/PCG/PCG.h	206
SunwayMatrixSolver/src/matrix/smsMatrixSolver.cpp,	SunwayMatrixSolver/src/solvers/PCG/PCG.h	207
SunwayMatrixSolver/src/matrix/smsMatrixTest.cpp,	SunwayMatrixSolver/tools/compass/compassInterface.cpp	208
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/compass/compassInterface.h	209
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/matrixConversion/coo2csr.cpp	210
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/matrixConversion/coo2ldu.cpp	210
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/matrixConversion/cs2coo.cpp	210
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/matrixConversion/ldu2coo.cpp	210
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/matrixConversion/ldu2csr.cpp	210
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/matrixConversion/printSMS.cpp	220
SunwayMatrixSolver/src/preconditioners/csrPreconditioner.h,	SunwayMatrixSolver/tools/printSMS/printSMS.h	220
SunwayMatrixSolver/src/preconditioners/lduPreconditioner.h,	SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.h	221
SunwayMatrixSolver/src/preconditioners/lduPreconditioner.h,	SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.h	221
SunwayMatrixSolver/src/preconditioners/lduPreconditioner.h,	SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.h	226
SunwayMatrixSolver/src/preconditioners/lduPreconditioner.h,	SunwayMatrixSolver/tools/readData/readFromOpenFOAM/readFromOpenFOAM.h	227

SunwayMatrixSolver/tools/timers/Timers.cpp, 229
SunwayMatrixSolver/tools/timers/Timers.hpp, 230
sw_matrix_destroy_
 SMS, 21
sw_solve_mg_
 SMS, 22
sw_solve_pbicgstab_
 SMS, 22
sw_solver_destroy_mg_
 SMS, 22
sw_solver_destroy_pbicgstab_
 SMS, 22
sw_solver_mg_set_maxiter_
 SMS, 22
sw_solver_mg_set_miniter_
 SMS, 22
sw_solver_mg_set_nfinestsweeps_
 SMS, 23
sw_solver_mg_set_npostsweeps_
 SMS, 23
sw_solver_mg_set_npresweeps_
 SMS, 23
sw_solver_mg_set_reltol_
 SMS, 23
sw_solver_mg_set_tol_
 SMS, 23
sw_solver_pbicgstab_set_maxiter_
 SMS, 23
sw_solver_pbicgstab_set_miniter_
 SMS, 24
sw_solver_pbicgstab_set_precond_
 SMS, 24
sw_solver_pbicgstab_set_reltol_
 SMS, 24
sw_solver_pbicgstab_set_tol_
 SMS, 24
symm
 SMS::lduMatrix, 83
 SMS::matrix, 89
Timer, 133
 _OpenTimer, 136
 ~Timer, 134
 closeTimer, 134
 count, 137
 endTimer, 134
 maxTimeSum, 135
 openTimer, 135
 printTimer, 135
 startTimer, 136
 timeStart, 137
 timeSum, 137
 Vector_Variance, 136
Timers.cpp
 getSystemTime, 230
timeStart
 Timer, 137
timeSum
 Timer, 137