

```

2  Eigen::VectorXd Radau3MOLTimestepper::discreteEvolutionOperator(
3      double time, double tau, const Eigen::VectorXd &mu) const {
4      Eigen::VectorXd discrete_evolution_operator(dofh_.NumDofs());
5      // Dimension of finite element space
6      const If::uscalfe::size_type N_dofs(dofh_.NumDofs());
7      LF_VERIFY_MSG(N_dofs == mu.size(),
8          "Dimension mismatch between the number of degrees of freedom "
9          "and the dimension of the argument vector.");
10
11     // Building the linear system for the implicitly defined increments
12     // Assembling the right hand side using block initialization
13     Eigen::VectorXd linSys_rhs(2 * N_dofs);
14     Eigen::VectorXd rhs_subtraction_term = A_ * mu; // precomputation
15     linSys_rhs << rhsVectorheatSource(dofh_, time + c_[0] * tau) -
16         rhs_subtraction_term,
17         rhsVectorheatSource(dofh_, time + tau) - rhs_subtraction_term;
18
19     // Implicit Runge-Kutta methods lead to systems of equations that must be
20     // solved in order to obtained the increments.
21     Eigen::SparseMatrix<double> linSys_mat;
22     // Assembling the system right hand side matrix using the (unfortunately
23     // officially not supported) Eigen Kronecker product
24     linSys_mat = M_Kp_ + tau * A_Kp_;
25     LF_VERIFY_MSG(linSys_mat.rows() == linSys_mat.cols(),
26         "The linSys_mat Eigen matrix is not squared.");
27     Eigen::SparseLU<Eigen::SparseMatrix<double>> solver;
28     solver.compute(linSys_mat);
29     LF_VERIFY_MSG(solver.info() == Eigen::Success, "LU decomposition failed");
30
31     // Solve linear system using Eigen's sparse direct elimination
32     Eigen::VectorXd k_vec = solver.solve(linSys_rhs);
33     LF_VERIFY_MSG(solver.info() == Eigen::Success, "Solving LSE failed");
34
35     // Compute action of the discrete evolution operator on argument vec
36     discrete_evolution_operator = mu + tau * (b_[0] * k_vec.topRows(N_dofs) +
37         b_[1] * k_vec.bottomRows(N_dofs));
38     return discrete_evolution_operator;
39 }

```