

C++ code 7.8.7: Sub-problem (7-8.f): Implementation of SDIRK discrete evolution operator

→ [GITHUB](#)

```
2 // Compute one step of the SDIRK scheme
3 template <typename Func, typename Jac>
4 Eigen::VectorXd DiscEvoSDIRK(Func &&f, Jac &&df, const Eigen::VectorXd &y,
5                               double h, double rtol = 1E-6,
6                               double atol = 1E-8) {
7     // The b weights are in the last row of coeffs.
8     Eigen::MatrixXd coeffs = ButcherMatrix();
9     int n_stages = coeffs.cols();
10    Eigen::VectorXd b = coeffs.row(n_stages);
11
12    // Vector after one step of the SDIRK scheme
13    Eigen::VectorXd Psi;
14
15    Psi = y;
16    // Compute array of stages
17    std::array<Eigen::VectorXd, 5> G = ComputeStages(f, df, y, h, rtol, atol);
18    for (int i = 0; i < n_stages; i++) {
19        Psi += h * b(i) * f(y + G[i]);
20    }
21
22    return Psi;
23 }
```