```
def forward(model, source, receiver,
     space_order=2):
    m, eta = model.m, model.damp
    # Allocate wavefield and auxiliary fields
    u = TimeFunction(name='u', grid=model.grid,
         time_order=2,
                     space_order=space_order)

    # Derive stencil from symbolic equation
    eqn = m * u.dt2 - u.laplace + eta * u.dt
    stencil = solve(eqn, u.forward)
    update_u = Eq(u.forward, stencil)

    # Source injection and receiver interpolation
    src = source.inject(field=u.forward,
         expr=src * dt**2 / m)
    rec = receiver.interpolate(expr=u)

    op = Operator([update_u] + src + rec,
         subs=model.spacing_map)
```