

```

! Initialize the derivatives at the surface; the surface is treated as a
single
! clear-sky layer so we only need to put values in region 1.
lw_derivatives_g_reg = 0.0_jprb
lw_derivatives_g_reg(:,1) = flux_up_surf / sum(flux_up_surf)
lw_derivatives(icol, nlev+1) = 1.0_jprb

! Move up through the atmosphere computing the derivatives at each half-level
do jlev = nlev,1,-1
! Compute effect of overlap at half-level jlev+1, yielding
! derivatives just above that half-level
lw_derivatives_g_reg = singlemat_x_vec(ng,ng,nreg,u_matrix(:,jlev+1),
lw_derivatives_g_reg)

! Compute effect of transmittance of layer jlev, yielding
! derivatives just below the half-level above (jlev)
lw_derivatives_g_reg = transmittance(:,jlev) * lw_derivatives_g_reg

lw_derivatives(icol, jlev) = sum(lw_derivatives_g_reg)
end do

```



```

...
! Move up through the atmosphere computing the derivatives at each half-level
do jlev = nlev,1,-1
! Inline everything in one loop over g-points
lw_deriv_old = lw_derivatives_g_reg
sum_tmp = 0.0_jprb
associate(A=>u_matrix(:,jlev+1), b=>lw_deriv_old)
!$omp simd reduction(+:sum_tmp)
do jg = 1, ng
! Compute effect of overlap at half-level jlev+1, yielding derivatives just above that
! half-level (matrix-vector multiply)
! both inner and outer loop of the matrix loops j1 and j2 unrolled
! inner loop:
j2=1 j2=2 j2=3
lw_derivatives_g_reg(jg,1) = A(1,1)*b(jg,1) + A(1,2)*b(jg,2) + A(1,3)*b(jg,3)
lw_derivatives_g_reg(jg,2) = A(2,1)*b(jg,1) + A(2,2)*b(jg,2) + A(2,3)*b(jg,3)
lw_derivatives_g_reg(jg,3) = A(3,1)*b(jg,1) + A(3,2)*b(jg,2) + A(3,3)*b(jg,3)

! Compute effect of transmittance of layer jlev, yielding
! derivatives just below the half-level above (jlev)
lw_derivatives_g_reg(jg,1) = lw_derivatives_g_reg(jg,1) * transmittance(jg,1,jlev)
lw_derivatives_g_reg(jg,2) = lw_derivatives_g_reg(jg,2) * transmittance(jg,2,jlev)
lw_derivatives_g_reg(jg,3) = lw_derivatives_g_reg(jg,3) * transmittance(jg,3,jlev)

sum_tmp = sum_tmp + lw_derivatives_g_reg(jg,1) + lw_derivatives_g_reg(jg,2) + &
& + lw_derivatives_g_reg(jg,3)
end do
end associate

lw_derivatives(icol, jlev) = sum_tmp
end do

```

Figure 1: Reference (top) and optimized (bottom) version of the longwave derivatives kernel used by TripleClouds.