Minor comments for revised version of:

“A dynamic approach to three-dimensional radiative transfer in subkilometer-scale numerical weather prediction models: the dynamic TenStream solver v1.0”

Could you try to measure the number of floating point operations? This would give a direct measure of the cost. Reporting just the speed against the ten-stream implementation in Libradtran (a non operational code) is not that informative. Whether dynamic TenStream becomes a viable contender for NWP models will largely depend on how fast it eventually is, which depends both on the cost (number of FLOPs) and whether it can effectively exploit hardware (FLOPs per second). I agree with leaving this second aspect for a future paper to address but the number of floating point operations could easily be measured with GPTL (https://github.com/jmrosinski/GPTL) or other timing library built with PAPI.

Regarding the second aspect, I am in fact a bit concerned that the Gauss Seidel implementation, even if divided into subdomains, won't be able to exploit SIMD vectorization on CPU's. (Perhaps GPU’s could be a better fit). You need not address this in the paper but do you think it would be a lot of work to write a Jacobi implementation for dynamic TenStream in the future in order to explore which gives better speed/accuracy trade-off on different hardware? I understand its convergence speed is worse but it could turn out to be a reasonable trade-off if it allows SIMD vectorization and Gauss-Seidel doesn’t, especially as CPU hardware is moving towards longer vector lengths with AVX-512 instructions being supported by newer CPU’s.

Line 66. SPARTACUS was not designed for NWP specifically, in the original Hogan and Shonk (2013) paper the authors actually talk more about climate models. The proliferation of (sub-) kilometer-scale NWP models arguably makes SPARTACUS more relevant for climate rather than NWP.

Line 67. This speed comparison is slightly out of date since SPARTACUS was recently sped-up by ~3x via code optimization. A better comparison might be to TripleClouds, it’s fully 1D-counterpart, and to say it’s 3-5 times more slower than TripleClouds (citing Fig. 3 in Ukkonen and Hogan, 2024). However, this is nitpicking sightly and it’s also OK to leave the older comparison (optimized SPARTACUS against an optimized McICA could still still be ~6x slower for all I know).

References:

Hogan, R. J., & Shonk, J. K. (2013). Incorporating the effects of 3D radiative transfer in the presence of clouds into two-stream multilayer radiation schemes. *Journal of the Atmospheric Sciences*, *70*(2), 708-724.

Ukkonen, P., & Hogan, R. J. (2024). Twelve Times Faster yet Accurate: A New State-Of-The-Art in Radiation Schemes via Performance and Spectral Optimization. *Journal of Advances in Modeling Earth Systems*, *16*(1), e2023MS003932.