AIP | Proceedings

Quantitative Spectroscopy in 3D

Lars Koesterke

Quantitative Spectroscopy in 3D

Lars Koesterke

Texas Advanced Computing Center, University of Texas at Austin, Austin, TX 78758, USA

Abstract. The Advanced Spectrum Synthesis 3D Tool ASSET is introduced. ASSET allows for the accurate and fast calculation of spectra from 3D hydrodynamical models. To achieve the highest numerical accuracy 3rd-order Bezier interpolations are employed and all available information from the model grid with respect to the spacial and frequency resolution is exploited.

ASSET is fully parallelized with OpenMP and MPI and highly optimized to run at about 25% of peak speed on workstations and clusters. The emergent flux for a single spectral line can be calculated from dozens of snapshots within a few minutes, and the whole spectrum ($2 \cdot 10^6$ frequencies) can be calculated on a small cluster (a few hundred threads) within a day.

The numerical methods, the serial optimization and the parallel implementation are described in some detail.

INTRODUCTION

New models for the photosphere and the upper convection zone of the Sun and other cool stars became recently available. These models are based on hydrodynamical simulations in 3 dimensions which carefully track the convective motions in the simulation domain. The major advantages of these new models are threefold. First, the fudge-parameters micro- and macro-turbulence (used in any 1D analysis) vanish, secondly, observed asymmetrical line profiles can be accuratey reproduced, and thirdly, cold regions which are crucial for the formation of molecules can co-exist with normal, i.e. warm regions.

A careful re-examination of the solar spectrum by means of these Hydro-models led to a downward revision of the solar abundance of oxygen and carbon by about 40% and 30%, respectively (cf. Allende Prieto et al. 2001, 2002). Naturally, this revision sparked a huge debate since any drastic change of the solar standard has severe implications in many fields of astronomy. The biggest controversy comes from the fact that solar interior models with a lower solar metalicity cannot be reconciled with helioseismological observations. For now it seems that the opacity in the solar interior is too low. This is extensively discussed in the contribution by Anil Pradhan in these proceedings.

There is ample evidence of a lower solar oxygen and carbon abundances from forbidden and allowed C I and O I lines, molecular OH and CH lines (Asplund et al. 2004, 2005). Moreover, a lower metalicity is in line with studies of oxygen abundances in H II regions, the interstellar medium, and O and B-stars in the solar neighborhood. However, many more detailed studies are needed to settle the debate and to conclusively determine the solar abundances from spectroscopic analysis by means of 3D Hydro-models of the Sun.

The dicussion on whether the lower abundances are correct or not relates to two principal questions. How accurate are the underlying Hydro-models and how accurately can the spectra be calculated in 3D? Obviously the two questions are tightly coupled. With-

out a precise method for the calculation of spectra in 3D, the validity of the 3D Hydro-models cannot be answered. In addition I argue that a method for the spectrum synthesis in 3D has to be precise *and* fast to allow for the investigation of many individual lines and large spectral regions in the same way as it is done in 1D.

In the present paper I introduce the Advanced Spectrum Synthesis 3D Tool ASSET and discuss its parallel implementation and performance on ordinary desktop computers and on RANGER at the Texas Advanced Computing Center (TACC), which is one of today's largest supercomputers. This is followed by a summary. In the appendix the methods applied in ASSET are compared to a different approach with respect to accuracy and speed.

ASSET: ADVANCED SPECTRUM SYNTHESIS 3D TOOL

The project started in 2004 with the idea of writing new software that would allow for the calculation of spectra in LTE from 3D Hydro models in a fashion that would be very similar to comparable codes in 1D. The functional requirements were set as follows:

- opacities from atomic and molecular species, with an arbitrary number of overlapping transitions,
- frequency-dependent continuum opacities to allow for the calculation of large frequency ranges,
- accounting for scattering,
- higher-order interpolation schemes,
- radiation transfer without any short-cuts,
- low memory requirements (< 1GB) to enable the use of medium-sized workstations,
- sufficient execution speed to allow for the calculation of large spectral windows on reasonably short time scales.

Hydro Models

The calculations presented in this paper are based on the solar time-series computed by Asplund (et al. 1999) and by Ludwig (et al. 2007). A typical input model has about 200,000 grid points ($50 \cdot 50 \cdot 80$ or $47 \cdot 47 \cdot 100$, respectively) which describe the upper part of the original hydro-model (photosphere down to $\tau \approx 200$) at a reduced horizontal resolution. Spectra, i.e. intensities and fluxes, are calculated separately for individual snapshots and then averaged.

Frequency Grid and Opacity Calculation

Two frequency grids with different resolutions (coarse and fine) are established. Both grids are equidistant in $\log(\lambda)$. Full opacities and emissivities from all sources (continua

and lines) are calculated on the fine grid. Subsequently the intensities are calculated in observer's-frame on exactly the same grid. The purpose of the coarser grid at 0.1 - 5 Å resolution is to provide continuum opacities for the calculation of the scattering (see next section).

The calculation of opacities for all 200,000 grid points is quite a tremendous task, if done individually for all grid points. Fortunately, in LTE the effort can be greatly reduced by precomputing a so-called opacity grid. The opacity grid consists of relatively few (300-500) points that cover equidistantly the model grid points in a log-temperature, log-density plane. Opacities are calculated only for these few opacity grid points in a pre-processing step. For this a modified version of SYNSPEC (Hubeny and Lanz 1995) is used, which does the calculation of the opacities on an equidistant $\log(\lambda)$ scale, but does *not* solve for the equation of radiative transfer. The opacities are stored in a file. When the spectrum is synthesized the opacities at the grid points (200,000 points) are derived by 3rd-order interpolations from the opacities on the file (300-500 points). That way the computational effort is reduced by a large factor while the accuracy is not tarnished and remains better than 0.1%.

Basic Setup

A long-characteristic scheme is applied for the calculation of the emergent intensities and fluxes. The outgoing intensities are computed along rays which start at all grid points of the top layer in different directions in μ and ϕ. The flux is typically integrated from 21 angles; the vertical ray, three μ angles and 8 ϕ angles, except for the shallowest μ-angle with only 4 ϕ angles. The flux integration utilizes a Gauss-Radau scheme. Both, opacities and source functions along the ray are derived via interpolation from opacities provided at the grid points of the hydro-mesh.

Scattering

The mean radiation field for all grid points is calculated separately by a different code based on short-characteristics. This code is not described in detail in the present paper, but many of the underlying principles, i.e. opacity grid, 3rd-order interpolation schemes, radiation transfer methods, etc. are the same. The mean radiation field enters the spectrum synthesis code and is used for the calculation of the scattering emissivity. Thomson and Rayleigh (H I) scattering is accounted for. To reduce the computational effort the scattering emissivity is usually calculated from continuum opacities on the coarser grid (cf. previous section) and interpolated to the fine frequency grid. Test with coherent scattering derived from the full opacities at the higher resolution reveal very small differences in the synthetic solar spectrum.
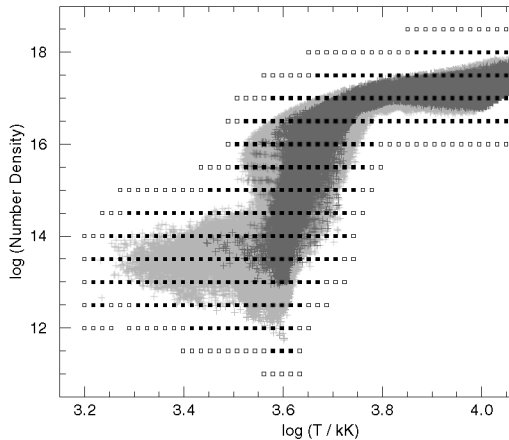
FIGURE 1. Opacity grid covering the individual grid points of the solar Hydro-simulation. The smaller grid (filled squares) allows for linear interpolation while the grid with additional padding (open squares) is sufficient for 3rd-order interpolations in both dimensions. The original grid points from all snapshots are shown in light-grey, one snapshot is highlighted (dark-grey). Note that for increased readability the number of density points has been reduced from the usual choice of 4 points to 2 points per decade. The resolution in temperature is the typical 250 K at 6000 K.

Computational Effort

Compared to the 1D case the calculation of a spectrum from a 3D model is much more costly. For a given frequency typically three rays for three different μ-angles have to be accounted for in 1D. In 3D the effort is higher by at least a factor of 1,000,000 which results from the number of azimuthal angles times the number of points in x and y times the number of snapshots that are used to time-average, i.e. $N_A(8) \cdot N_x(50) \cdot N_y(50) \cdot N_S$ (50), with typical numbers given in brackets.

On top of that, multi-dimensional interpolations are needed to derive opacities and source function along the ray. Since the data are 4D, i.e. 3 spacial dimensions and a frequency shift due to the velocity field, all interpolations are generally 3D. To perform one 3rd-order interpolation in 3D, the nearest 64 datapoints are used and 21 interpolations in 1D are executed. Piecewise cubic Bezier polynomials that do not introduce artificial extrema are used (see, e.g., Auer 2003).

76

Ray Setup

Rays start at the grid points of the top layer and are followed downwards at arbitrary angles (μ, ϕ) until a sufficient optical depth of $\tau \approx 20$ is reached. As illustrated in Fig. 2, the rays do not hit grid points in deeper layers, but intersect with the mesh at arbitrary points. Multi-dimensional interpolations are needed to derive opacities and source functions from the input provided at the grid points. The rays intersect with the mesh when a z-plane, an x or y-plane, or a frequency-plane is crossed. The interpolations are generally 3D: 2D in space and 1D in frequency for the first two types, and 3D in space for the third type.

Accounting for the points of intersection with the frequency grid ensures that the opacities are sampled at the full frequency resolution of the input data and that therefore line opacities are always fully resolved to the extend of the resolution of the underlying frequency grid. As a side effect of this approach, the number of points on a ray between the outer boundary and the inner boundary at $\tau = \tau_{max}$ depends on the frequency resolution of the input. At a higher resolution more points are inserted and at a lower resolution fewer (or no) points are inserted. Solving the equation of radiation transfer along the ray depends mildly on the number of points. Consequently, the absolute level of the intensity (and flux) depends to some degree on the chosen frequency resolution.

Note that in other codes (LTE: Nordlund & Stein 1990, CO5BOLD: Ludwig & Steffen 2007) only intersections with the z-plane are considered. This limits the accuracy of the calculation. First, shallow rays may cross many cells in x and y before a z-plane is intersected, and secondly there is no control over the frequency shift between adjacent points on the ray. For large velocity gradients line opacities may not be completely resolved. This is discussed in more detail in the Appendix.
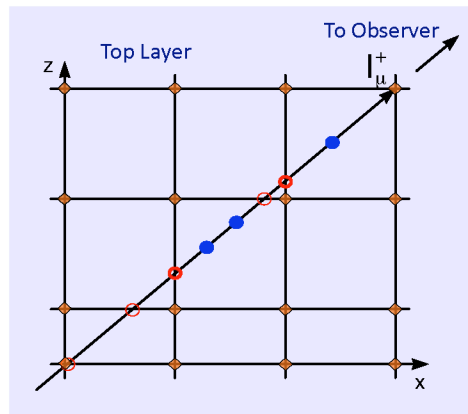
FIGURE 2. Setup of a ray starting at a grid point of the top layer. Opacities and source functions (including thermal and scattering emissivity) are provided at the grid points (yellow diamonds). Opacities and source functions along the ray are derived from interpolations in 3D whenever the ray intersects the mesh. Three different kinds of intersection are identified. 1) intersection in the z-plane (thin, red circles), 2) intersection in the x/y-plane (thick, red circles), and 3) intersection with the ν-plane (filled, blue circle). The first two kinds require a 2D interpolation in space and a 1D interpolation in frequency, the last kind requires a 3D interpolation in space. Note that the spatial grid is regular in x and y but not in z.

Radiation Transfer

Once a ray is established a standard method for the solution of the radiation transfer equation is used. A direct (integral) method of 1st or 2nd order, or the Feautrier method may be applied. The absolute level of intensity (and flux) depends mildly (0.1 to 0.3%) on the method. However, normalized profiles are much less affected. A 1st-order direct method is selected as the default, since it provides the highest computational performance in the current setup.

HIGH-PERFORMANCE AND PARALLEL COMPUTING

The development of the serial version of the code was completed in 2006/07 and the first results have already been published (Koesterke et al. 2006, 2008). At that time the code was used primarily on a Mac Mini dual-core computer with 2GB of memory. It took one core of this machine about 10 seconds to calculate the flux (21 angles) for one frequency. Utilizing both cores, the spectrum of a single line (180 frequencies) could be calculated for 100 snapshots within 1 day.

In the fall of 2007, I joined the High-Performance Computing (HPC) group at the Texas Advanced Computing Center (TACC) at the University of Texas at Austin. This new appointment allowed me to improve my parallel programming skills and to learn/develop advanced optimization techniques. In the first half of 2008 the code was rewritten to exploit OpenMP and MPI parallelism and to dramatically improve the serial execution speed.

Serial Optimization and OpenMP

This section explains the performance gains from serial optimization and the implementation of OpenMP. Four different types of performance measures are summarized in Table 1 and explained in the following sections. The total performance gain is about a factor of 400.

TABLE 1.   Overview of performance measures and corresponding gains.

Speedup	Measure
2.5	Hardware: Laptop CPU $\rightarrow$ High-end CPU (Intel Penryn, AMD Barcelona) Compiler: g95 $\rightarrow$ Intel, PGI
3.7	OpenMP on 4 cores
5	Optimization of memory access (stride-1 access) Trading memory for operations (400 MB $\rightarrow$ 800 MB)
8	Advanced serial optimization
400	Total speedup 10$\times$: Compiler, multicore hardware and OpenMP 40$\times$: Serial optimization

Hardware and Compilers

Switching from a Mac Mini, which uses laptop technology with a slower CPU and memory (compared to desktops), to high-end hardware like INTEL PENRYN or AMD BARCELONA CPUs and changing from the free g95 FORTRAN compiler to the INTEL or PGI compiler gives a performance gain of about 2.5.

OpenMP

OpenMP is implemented in two different variants. Both, the Angles loop and the loop over the starting points in y are parallelized. Parallelization of the angle loop is advantageous because it is positioned further outside in the loop nest (cf. Table 2), but suffers from a low loop count and does not scale well to core counts larger than 4. For the calculation of the flux the number of angles is typically set to 21 compared which is lower than the typical number of points in y direction ($N_y \approx 50$). Moreover for some applications only one ray/angle (typically the vertical ray) may be calculated. For both approaches the parallel efficiency is higher than 90% on four cores, resulting in a performance gain of about 3.7 on four cores.

Memory Access

The execution speed of most scientific codes is severely impacted by the limited capabilities of the memory subsystem. Consequently, improving memory access gives usually the highest performance gains. The bandwidth of the data transfer between memory and CPU is generally highest when array elements are accessed in a regular pattern. The stride length identifies the distance (in memory) between two array elements accessed in a loop. A stride length of 1 means that there are no gaps. Regular access patterns and low stride lengths allow for an increased memory bandwith by maximizing the use of the cache (a tiny, but very fast memory buffer) and by allowing for efficient prefetching of the data from memory into the cache.

Based on these considerations, ASSET was rewritten so that all inner loops facilitate stride-1 memory access. Prior to the rewrite the loops were set up as described in Table 2, second column. A five-times nested loop was used to calculate the intensities for all frequencies and angles and for all rays starting at the $N_y \cdot N_x$ points of the top layer. The frequency loop was chosen as the outermost loop to minimize the cost of reading and interpolating the opacity data.

The original (serial) setup did not ensure stride-1 access at the innermost loop level because the rays move through the 4D datacube in an arbitrary fashion. Unfortunately none of the loops, except for the outermost frequency loop enables efficient stride-1 memory access. To establish stride-1 access, the frequency loop had to become the innermost loop. Since it is impossible to store the opacities for all (potentially millions) frequencies, the frequency loop was split into an outermost loop over frequency blocks and an innermost loop over the frequencies within the block. Numerical experiments

79

revealed that a block size of 192 (inbetween unfavorable powers of 2) gives the best performance.

TABLE 2. Loop structure in the first, serial version and the optimized version with parallel programming.

Loop nest	serial	optimized	parallel programming
(outermost) 1.	Frequency	Frequency block	MPI
2.	Angles	Angles	OpenMP (2nd variant)
3.	Starting point y	Starting point y	OpenMP (main variant)
4.	Starting point x	Starting point x	
5.	Along the ray	Along the ray	
(innermost) 6.		Frequencies in block	

In the course of this modification more quantities were stored rather than being recalculated. This was possible since the high-end hardware has much more memory than the previously used platform on which ASSET was originally developed. The total speedup of the measures described in this section is about a factor of 5. Moreover the rearrangement of the loop nest allowed for the application of more advanced optimization techniques described in the next section.

Advanced Serial Optimization

A variety of techniques is applied to improve the serial performance of ASSET:

- Customization of compiler options for individual subroutines with emphasis on the decision of whether to inline a routine or not,
- arrangement of arrays in memory,
- bundling of 1D interpolation routines for 2D interpolations,
- reverse inlining (beefing-up innermost routines),
- manual loop unrolling and loop blocking to help the compiler,
- verification of the assembly code produced by the compiler,
- arrangement of data in memory so that SIMD (Single Instruction Multiple Data) instructions could be maximally exploited,
- custom-build exponential function that exploits the fact that the solution of the radiation transfer along a long-characteristic scheme requires only arguments of a limited range between -24 and 0.,
- etc.

The total (serial) speed-up of these measures is a factor of about 8.

Ranger at TACC

In the spring of 2008, the new SUN Constellation cluster RANGER, went into production at the Texas Advanced Computing Center (TACC). RANGER is the first system funded by the National Science Foundation's *Path to Petascale* program and pro-

vides unprecedented capabilities to researchers at US Institutions and their collaborators around the world. RANGER consists of 3936 quad-socket quad-core (16 cores with 32 GB of memory) compute nodes, that are linked via a high-speed INFINIBAND interconnect. With a theoretical peak performance of 579.4 TFlops, RANGER ranks currently 8th on the TOP 500, JUNE 2009 list (www.top500.org). ASSET is optimized to run efficiently on RANGER, any other shared- or distributed-memory cluster, or any single/multicore workstation.

Parallelization with MPI

MPI parallelism is implemented by distributing the frequency blocks of the outermost loop (cf. Table 2) among the compute nodes/CPUs/cores of a cluster. ASSET can be executed in pure OpenMP on a desktop or one compute blade or in pure MPI mode on a cluster (or a single blade), or in hybrid mode. The distribution of work requires very little communication and the amount of overhead is small as well. Scaling on RANGER is very good as shown in Fig. 3.
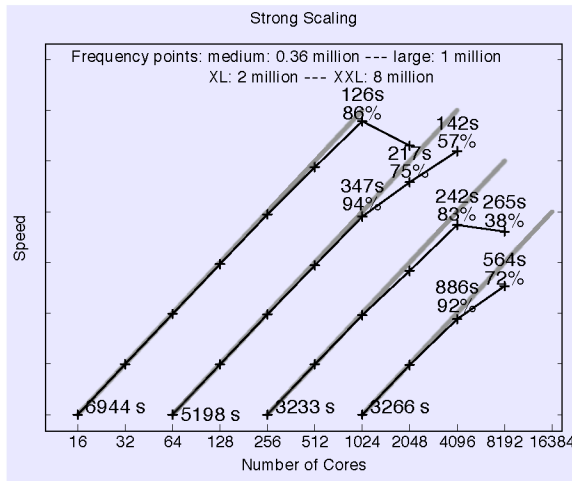
The calculation of the full solar spectrum for 1 snapshot with the spectral lines fully resolved (steps of ~ 0.2 km s^{-1}) takes about 1 hour on 16 blades, i.e. 256 compute cores, which scales almost perfectly to 4 minutes on 256 blades. With the extensive serial optimization ($40\times$) in place, the calculation of the whole solar spectrum is no longer large enough to require all of RANGER'S compute nodes. By using 100×16 blades, which is not even half of RANGER, the time-avaraged spectrum can be calculated from 100 snapshots within 1 hour as well.

SUMMARY

ASSET is a very powerful tool for the calculation of LTE spectra from 3D hydrodynamical simulations of the Sun and other cool stars. Extensive comparisons (LTE: Ramirez et al. 2009, Asplund & Collet priv. comm.; CO5BOLD: Ludwig priv. comm.) have shown excellent agreement with both, the spectrum synthesis code LTE (Asplund et al. 2000) that is part of the Nordlund/Stein hydro package (Nordlund & Stein 1990), and with the spectrum synthesis code that is part of the CO5BOLD package (Ludwig & Steffen 2007).

Considerable effort has been taken to ensure high numerical accuracy. All interpolations are 3rd-order and the complete input data with respect to its spacial and frequency resolution is taken into account for the spectrum synthesis.

ASSET is fully parallelized and employs OpenMP and MPI to exploit both shared and distributed memory computers/clusters. It is highly optimized and runs at about 25% of peak performance (4 floating-point operations per clock cycle) of modern INTEL and AMD CPUs. On a modern multi-core workstation 30 frequencies per second can be calculated (flux integration from 21 angles on a $50 \times 50 \times 82$ grid, one snapshot). The time-averaged profile of a single line can be calculated within a few minutes and the calculation of the whole spectrum with about $2 \cdot 10^6$ frequencies can be easily achieved on a small cluster in a day.

FIGURE 3. Strong scaling experiment on RANGER for 4 different problem sizes. The ideal speed up is indicated by the straight diagonal bars. The calculation of the full solar spectrum from 2,000 Å to 20,000 Å with about 2 million frequency points is the third test. Scaling is excellent up 4096 cores where the compute time decreases below 5 minutes. In the largest test the number of frequency points is artifcially increased.

APPENDIX: COMPARISON WITH OTHER APPROACHES

In spectrum synthesis codes that are part of the hydro simulation packages a different approach is being used. In these codes the spacial interpolations (cf. Fig. 2) that are necessary during the radiation transfer in ASSET are avoided by rotating first the input model to the angles (μ, ϕ) of the ray under consideration. This approach still requires spacial interpolations in 2D, but they are performed only once when the model is rotated, and not for every frequency during the radiation transfer. Both methods are compared in Fig. 4.

At a first glance the alternative procedure seems to be much more efficient. However, a close comparison of both approaches reveals a different and somewhat surprising result. In the following sections I compare both accuracy and speed of the two methods.

Comparison of the Accuray

The alternative approach suffers from two deficiencies. First, only intersections of the ray with z-planes are considered. Secondly, the number of interpolations in the v-space is higher.
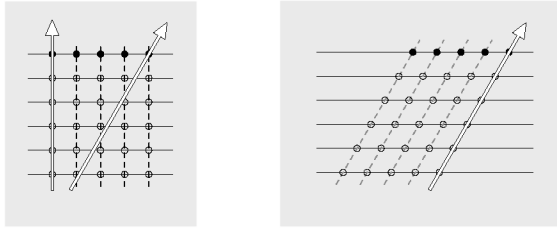
FIGURE 4. Left: Setup of the vertical ray and a second ray with arbitrary angles (μ, ϕ) in ASSET. The calculation of the latter ray requires multi-dimensional interpolations in space. Right: Setup for an arbitrary ray in the alternative approach. The hydro model is sheared to match the angles μ and ϕ. This is accomplished by rearranging the points of a given z-plane within the plane. The calculation of the intensity along the ray does not require interpolations in space.

Points of intersection: A model is sheared by rearranging the points of a given z-plane within the plane, so that the ray hits exactly the rearranged mesh points (Fig. 4, right panel). The radiation transfer is solved along the ray based on the points of intersection of the ray and the z-planes, but not on points of intersection with the x- or y-planes. Between the two approches there is no difference for the vertical ray, but particularly for more shallow rays, information is lost. This may affect studies of the center-to-limb variation (CLV) when very shallow rays are considered. However, the effect on the flux is small since the flux integration weights quickly decrease with inclination (μ) angle.

Number of Interpolations: In both approaches, 1D frequency interpolations are performed where the ray intersects the mesh. This is necessary to offset the Doppler effect and to obtain the opacities and source functions at the frequency (observer's frame) under consideration. The difference is that in the alternative approach, interpolations of the (projected) velocity field are needed when the model is sheared to match the inclination of the ray. These are 2D interpolations. To summarize, ASSET requires 1D interpolations in frequency, while the other approach requires 3D interpolations in frequency/velocity. This may be important when verly small line shifts of the order of 10 m/s are investigated.

Comparison of the Execution Speed

The two approaches differ in the sense that in ASSET multi-dimensional interpolations of the opacities are done on-the-fly for all frequencies and angles. In the other approach the input model is sheared for each angle once by means of very similar interpolations, which renders the on-the-fly interpolations unecessary. At first glance the latter strategy seems much more efficient since the costs of the interpolation in ASSET are quite high. However, a closer look reveals a different story.

The interpolations on the ray are avoided, but at the price of having to interpolate the opacities from the opacity grid onto the hydro model for every ray. In ASSET these interpolations are done once per frequency, but in the alternative approach every angle uses its own individually sheared model and therefore requires its own interpolations of

83

the opacities. Table 3 summarizes the costs for a flux calculation based on 21 angles. The alternative approach is almost twice as costly as the method on which ASSET is based.

Note that it is not a coincidence that the cost for the interpolation of the opacities is about the same as the cost for the calculation of the intensity of one inclined ray. Both calculations require about the same number of interpolations.

TABLE 3. Costs of a flux calculation with 21 angles for ASSET and the alternative approach based on sheared input models. Arbitrary time units are used. The cost for the interpolation of the opacities is 5 units. The setup of a ray and the integration of the intensity costs 4 and 5 time units, respectively, depending on whether the ray is vertical or inclined. In the alternative approach all rays are vertical (central) rays.

Operation (per frequency)	ASSET	Alternative approach
Interpolation from Opacity grid to model	1×5	21×5
Central ray	1×4	21×4
Inclined rays	20×5	N/A
Total	109	189

REFERENCES

1. Allende Prieto, Lambert, D. L., Asplund, M. 2001, ApJ, 556, L63
2. Allende Prieto, C., Lambert, D. L., & Asplund, M. 2002, ApJ, 573, L137
3. Asplund, M., Grevesse, N., Sauval, A. J., Allende Prieto, C., & Blomme, R. 2005, A&A, 431, 693
4. Asplund, M., Grevesse, N., Sauval, A. J., Allende Prieto, C., & Kiselman, D. 2004, A&A, 417, 751
5. Asplund, M., Nordlund, Å, Trampedach, R., & Stein, R. F. 1999, A&A, 346, L17
6. Auer, L., 2003, Formal Solution: EXPLICIT Answers in *Stellar Atmosphere Modeling*, ASP Conference proceedings, 2003, Vol. 288, Hubeny I., Mihalas D., Werner K. (eds)
7. Hubeny, I., Lanz, T. 1995 "SYNSPEC — A Users's Guide" http://nova.astro.umd.edu/Tlusty2002/pdf/syn43guide.pdf
8. Koesterke, L., Allende Prieto, C., Lambert, D.L. 2006, proceedings of the 14th Cambridge Workshop on Cool Stars, Stellar Systems, and the Sun, G. van Belle, ed. (Pasadena, Nov. 2006)
9. Koesterke, L., Allende Prieto, C., Lambert, D.L. 2008, ApJ, 680, 764
10. Ludwig, H.-G., Steffen, M. 2007, in press (arXiv: 0704.1176)
11. Nordlund, Å, & Stein, R. F., 1990, Comp. Phys. Comm., 59, 119
12. Ramirez, I., Allende Prieto, C., Koesterke, L., Lambert, D.L., Asplund, M. 2009, ApJ in press (ArXiv: 0905.3101)