

VieSched++: A new Scheduling Tool in VieVS

Matthias Schartner, Johannes Böhm

Abstract Scheduling is an integral part of every VLBI experiment. The task of a scheduler is to create the sequence of observed sources for each station. Because the amount of possible source sequences is extremely high, selecting the best sequence is a challenging problem and can be exploited for optimization. Typically scheduling software is used to automate this process. With the VGOS era approaching, high demands are posed on scheduling software. We present a new scheduling software called VieSched++ as part of the Vienna VLBI and Satellite Software (VieVS) [3] which is able to fulfill those requirements. This software follows many new ideas concerning the algorithms used to create schedules and support various new features. It is written from scratch, keeping VGOS requirements in mind. First results using the new scheduling software look very promising.

Keywords Scheduling, VieSched++, VieVS

1 Introduction

For the analysis of VLBI sessions several different software packages exist, like the Vienna VLBI and Satellite Software (VieVS), Calc/Solve, OCCAM, C5++, Where, and many more. But, when it comes to scheduling, most of the IVS sessions are scheduled using only one software package called SKED [4], which was developed by NASA/GSFC. Those schedules are usually not cross-validated against results from other software packages. In Vienna, we had developed our own

scheduling tool [11] as part of VieVS which is successfully applied for scheduling geodetic VLBI sessions, especially for the AUSCOPE [5] network. While schedules generated with this software tool were considered good, the software had some limitations considering the flexibility, and – most importantly – it was not capable of scheduling VGOS [8] experiments. Based on the experience we gained with our previous software, we have started to develop a new scheduling tool from scratch as part of VieVS called VieSched++.

2 Concept

VieSched++ is written in C++ using a fully object-oriented software design. The software consists of two parts, the scheduler and a graphical user interface (GUI). It makes use of many modern software development approaches and supports multithreading for increased performance. It is a multi-platform application and is successfully tested on Linux and Windows using different compilers.

The scheduler supports all features required in geodesy and astrometry like optimization of sky coverage (Section 3.4), tagalong mode, fillin mode (see Section 3.2), subnetting, and basic support for twin telescopes. The output is available in both .skd and .vex format. Besides the schedule files, an operation notes file, helpful log files, and other files with statistical information can be generated as well as an empty NGS file which can be directly used in the VieVS VLBI package to simulate the schedule. The scheduler is controlled by a simple .xml document which can be set up by the GUI or by hand. To make

the software more consistent with today's schedules, VieSched++ supports the sked catalogs.

Great care was taken to ensure that the software and the GUI are very intuitive and therefore easy to use. Installing the software should also be easy because it has only a small amount of dependencies. For the scheduler, only the Standards of Fundamental Astronomy (SOFA) [10] libraries need to be linked. Other pre-requirements are the boost C++ header files and OpenMP for the multithreading support. While it is sufficient to only include the boost header files, it is recommended to link the boost libraries for additional features. The Qt libraries are required for the GUI. To further simplify the installation process, a CMake file for the scheduler and a qmake file for the GUI are provided.

While the main purpose of the GUI is the creation of the .xml documents required for the scheduler, it comes with many additional features like a schedule analyzer and comparison tools between different schedules and between schedules and station log files. Although VieSched++ is a standalone tool of the VieVS package, the interaction between VieSched++ and the VieVS VLBI software is straightforward, and results from the scheduler can be directly used.

3 Algorithms

All algorithms used in VieSched++ are redesigned from scratch. In the following subsections, some major differences compared to other software packages are shown and their usefulness is discussed.

3.1 Multi-scheduling

With VieSched++ it is recommended to not only generate one schedule for a session but create many schedules with different parameters. This can be automatically achieved using our multi-scheduling support. There are many parameters which can be used in every scheduling software to fine tune a schedule, such as the maximum allowed slew time per station, the time between observations to the same source, or the maximum allowed idle time for slow antennas. At the same time, it is difficult to decide

which scan is the best at a certain time and therefore should be scheduled. To measure this, usually multiple quantities are combined like the improvement in the sky coverage, the number of expected observations, the duration of the scan, and many more. Additional parameters combine these quantities to decide which scan should be scheduled. Together this leads to a huge variety of parameters which can be used to fine tune a schedule. Different networks with different stations benefit greatly from customized scheduling parameters [9]. The multi-scheduling tool helps in this process to create an optimized schedule with customized parameters. Instead of only scheduling one schedule at a time it creates many schedules automatically by varying some scheduling parameters. The multithread support reduces the processing time significantly. It is possible to compare the huge amount of schedules based on statistics in the GUI to pick the best schedule. Furthermore, the output can be used directly in the VieVS VLBI software to simulate the sessions in order to get a better comparison of the schedules based on repeatability of geodetic parameters. Altogether this increases the quality of the schedules significantly.

3.2 Recursive Scan Selection / Fillin Modes

In geodetic VLBI schedules, a concept named fillin modes is used to minimize the amount of idle time per station. Usually, there are two main reasons why idle time in a schedule occurs. First, most stations have very different slew rates, and therefore some stations have to wait for others to finish slewing before a scan can be started. Second, most stations have different sensitivities and therefore observing times which lead to some stations finishing a scan earlier than others. Thus, a so called fillin mode tries to minimize the idle time by scheduling more scans in between the main scans with a reduced amount of stations which would otherwise be idling. Most scheduling software follows a sequential scan selection, which means they start at the beginning of the session and schedule scan after scan increasing in time. If some stations finish way before others additional fillin mode scans are introduced to reduce the idle time. The downside of this approach is that it only reduces idle time based on shorter ob-

serving times which occur after an observation and not based on different slew times before an observation.

With VieSched++ this is not necessarily the case. Figure 1 illustrates how the recursive scan selection in the standard case works. First, the scheduler decides to



Fig. 1 Recursive scan selection. Red (dark) lines indicate observing times for main scans. Green (light) lines indicate observing times for recursively selected scans. Numbers show the order in which these scans are selected.

schedule scan number 1 at the beginning of the session. The next scheduled scan is scan number 2. Now the software checks if it is possible to squeeze in another scan between 1 and 2. This might be possible for some stations which are faster slewing or are more sensitive and therefore require less observing time. If a new scan is possible it is scheduled as scan number 3. Again the software checks if it is possible to squeeze in another scan, this time between scan 1 and 3. If not, the software checks if it is possible to add a scan between scan 3 and 2. If scan number 4 is possible the software tries to squeeze in another scan between 3 and 4 and 4 and 2. If no more scans are possible it will continue with scan 5 and the process starts over again. Following this algorithm, the whole session is scheduled, and the idle time of both different slew times and different observing times is reduced.

While this is the most basic case it is also possible to start the schedule at any time. Figure 2 illustrates another case. This time the scheduler starts with the most important scans for this session.

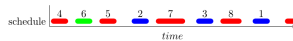


Fig. 2 Recursive scan selection. First, scans which are assumed to have the highest impact (blue: scans 1, 2, and 3) are scheduled. Afterward, the gaps between those scans are filled (see Figure 1).

For example, those could be scans during Intensive sessions that will be close to the edges of the commonly visible sky. It is assumed that these observations have the highest impact on the result [1], [12]. After fixing the most important scans the gaps in between are recursively filled.

Another use is the so-called fillin mode a posteriori, where the schedule is first created without any recur-

sion and the recursive scan selection happens after the schedule reaches the end time. In the case of Figure 1 the scheduler would first select the scans 1, 2, 5 and 7 and afterwards fill the gaps in between. This works particularly well in the case of multiple tagalong stations (see Section 4).

3.3 Station-, Source-, and Baseline-based Parameters

To increase the flexibility of the scheduler it is possible, but not necessary, to give every station, source, or baseline its own optimization parameters. This can be helpful if you have a network with very different stations. For fast slewing, smaller antennas it might bring an improvement to schedule them differently than slower, bigger antennas and give them, for example, a different maximum allowed slew distance. The same is true for sources. If you have a set of calibration sources it might bring a benefit to schedule them differently than your target sources and set a different minimum required signal to noise ratio or a different minimum time between two scans to the same source. On top of that, it is possible that these parameters can change in time. This might be helpful if your session has a special science goal like relativistic experiments and your session consists of multiple phases. With this technique, it is further possible to start and end the tagalong mode of stations during a session. For example, this is done by the CONT campaign for stations participating in Intensive sessions [2]. It is also possible to follow custom scheduling designs like the Austral star mode [6].

3.4 Sky Coverage

One major disadvantage of the previous VieVS scheduling tool was the representation of the sky coverage. There, the sky is divided into thirteen areas. If an observation is inside one of these areas, the area is assumed to be saturated and further observations in a certain time epoch that are in the same area are assumed to have no improvement on the sky coverage. This representation fails if observations are close to the edges of the areas. As with other software packages in VieSched++, the sky is no longer divided into distinct

areas, but instead the angular distance between the new and previous observations is taken into account.

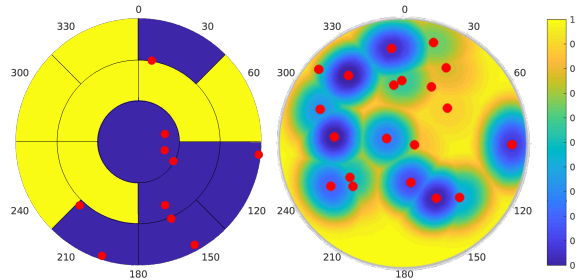


Fig. 3 Comparing the old sky coverage model with 13 areas (left) and the new implementation using angular distance and time as parameters. Red dots mark observations.

Several parameters can be used to describe the model. Figure 3 compares the old representation with the new one when decreasing the saturation of the sky coverage with the distance from the observation and with time.

4 Results

The usefulness of the new scheduling algorithms is discussed using the first CONT17 VGOS schedule CONTB1701. From a scheduling point of view, CONTB1701 is very special. According to personal email communication, four out of eight stations are scheduled in tagalong mode (see Table 1).

Table 1 Example of scheduled CONT17 VGOS network. Four stations were part of a core network, four stations were scheduled in tagalong mode. The schedule is limited by two slow core stations.

CONTB1701		slew rate	idle	scans
GGAO12M	core	300/66	15%	1178
ISHIOKA	tag	720/360	74%	412
KOKEE12M	core	720/300	80%	363
ONSA13NE	tag	720/360	61%	736
ONSA13SW	tag	720/360	61%	735
RAEGYEB	core	720/360	58%	722
WESTFORD	core	200/120	10%	1078
WETTZ13S	tag	720/360	73%	509

While most of the stations are very fast slewing antennas with a slew rate of 720 degrees per minute in

azimuth and 360 degrees per minute slew rate in elevation, two antennas are considerably slower. Both of these antennas are part of the main scheduling network. Additionally, one of those two slower stations, GGAO12M, is given double weight in the scheduling software.

This results in a schedule which is completely limited by the two slow antennas in the core network, which can be seen by comparing the number of scans and amount of idle time in Table 1.

With the recursive scan selection and especially the fillin mode a posteriori it is possible to create a schedule which is better suited for the full network while keeping the results from the core network intact. Using the multi scheduling feature (see Section 3.1), more than 100 different schedules were created and compared to pick the best one. The schedules are automatically created in three phases:

1. a first schedule is created using only the core stations,
2. the other stations are tagged along, and
3. the fillin mode a posteriori (see Section 3.2) is used to minimize idle time and increase number of scans.

Table 2 summarizes results achieved with this new procedure. The number of scans and the number of observations are considerably higher using VieSched++ compared to the submitted CONTB1701 schedule, both by looking only at the core network and by analyzing the full network. Even more observations can be scheduled using the fillin mode a posteriori mode.

Table 2 Comparison of the number of scans and simulated repeatabilities. Different sub-networks are investigated. *Core* means only the four core stations are used, *full* means all stations are used, and *fi a post* stands for fillin mode a posteriori.

	CONTB1701	VieSched++
scans core network	1180	1365
scans + fi a post		2508
obs core network	3267	5007
core + tagalong	12985	18540
core + tag + fi a post		23771
coord core network	9.8	3.7 [mm]
coord full	4.7	2.2 [mm]
xpol core network	831	645 [μ as]
xpol full	357	114 [μ as]
dut1 core network	50.5	31.9 [μ s]
dut1 full	17.0	4.8 [μ s]

Using the VieVS VLBI software, simulations were carried out to compare the effect on geodetic parameters. Table 2 lists repeatability values based on 300 simulations of the submitted CONTB1701 schedule and the schedule created with VieSched++. The simulations were carried out like standard SX observations and are based on a structure constant C_n of $1.8 \cdot 10^{-7} m^{-1/3}$ and a scale height of 2 km for the description of tropospheric turbulence, an Allan Standard Deviation of 10^{-14} after 50 minutes for the clocks, and an additional white noise of 30 picoseconds per observation [7].

Based on the simulations, a significant improvement can be seen by using the schedule created using VieSched++.

5 Conclusion

VieSched++ is a new modern VLBI scheduling software written in C++ and redesigned from scratch. It supports all necessary features to create geodetic and astrometric VLBI schedules. Many new ideas are implemented to increase the quality of created schedules as well as the flexibility of the software. One main goal is to automate the creation of highly optimized schedules. A GUI can be used to set up the schedule and analyze results.

Based on simulations, schedules created with VieSched++ look very promising. While still being developed, the software already runs very stably and can be used easily. It is planned to further optimize the implemented algorithms, as well as to add new features such as a tree-based scan selection and quality of life improvements.

References

1. K. Baver, J. Gipson, M. S. Carter, K. Kingham. Assessment of the First Use of the Uniform Sky Strategy in Scheduling the Operational IVS-INT01 Sessions (2012). International VLBI Service for Geodesy and Astrometry 2012 General Meeting Proceedings, NASA/CP-2012-217504, 251—255.
2. D. Behrend. Continuous VLBI Scheduling: The CONT14 Example (2015). Proceedings of the 22nd European VLBI Group for Geodesy and Astrometry Working Meeting.
3. J. Böhm, S. Böhm, J. Boisits, A. Girdiuk, J. Gruber, A. Hellerschmied, H. Krásná, D. Landskron, M. Madzak, D. Mayer, J. McCallum, L. McCallum, M. Schartner, K. Teke. Vienna VLBI and Satellite Software (VieVS) for Geodesy and Astrometry (2018). Publications of the Astronomical Society of the Pacific, 130(986), 044503, 1-6.
4. J. Gipson. Sked - VLBI Scheduling Software (2016). NASA Goddard Space Flight Center, ftp://gemini.gsfc.nasa.gov/pub/sked/sked_Manual_v2012May09.pdf.
5. J.E.J. Lovell, J.N. McCallum, P.B. Reid, P.M. McCulloch, B.E. Baynes, J.M. Dickey, S.S. Shabala, C.S. Watson, O. Titov, R. Ruddick, R. Twilley, C. Reynolds, S.J. Tingay, P. Shield, R. Adada, S.P. Ellingsen, J.S. Morgan, H.E. Bignall. The AuScope geodetic VLBI array (2013). *J Geod*, 87, 527—538.
6. L. McCallum, D. Mayer, K. Le Bail, M. Schartner, J. McCallum, J. Lovell, O. Titov, S. Fengchun, G. Sergei. Star Scheduling Mode - A New Observing Strategy for Monitoring Weak Southern Radio Sources with the AuScope VLBI Array (2017). Publications of the Astronomical Society of Australia, 34, E063. doi:10.1017/pasa.2017.58.
7. A. Pany, J. Böhm, D. MacMillan, H. Schuh, T. Nilsson, J. Wresnik. Monte Carlo simulations of the impact of troposphere, clock and measurement errors on the repeatability of VLBI positions (2011). *Journal of Geodesy*, 85(1), doi:10.1007/s00190-010-0415-1, 39—50.
8. B. Petrachenko, A. Niell, D. Behrend, B. Corey, J. Böhm, P. Charlot, A. Collioud, J. Gipson, R. Haas, T. Hobiger, Y. Koyama, D. MacMillan, Z. Malkin, T. Nilsson, A. Pany, G. Tuccari, A. Whitney, and J. Wresnik. Design Aspects of the VLBI2010 System - Progress Report of the IVS VLBI2010 Committee (2009). NASA/TM-2009-214180.
9. M. Schartner, J. Böhm, D. Mayer, L. McCallum, A. Hellerschmied. Recent developments in scheduling with VieVS (2017). Proceedings of the 23rd European VLBI Group for Geodesy and Astrometry Working Meeting.
10. IAU SOFA Board, IAU SOFA Software Collection (2018), Issue 2018-01-30, <http://www.iausofa.org>.
11. J. Sun, J. Böhm, T. Nilsson, H. Krásná, S. Böhm, H. Schuh. New VLBI2010 scheduling strategies and implications on the terrestrial reference frames (2014). *J Geod*, 88(5), 449—461.
12. M. Uunila, A. Nothnagel, J. Leek. Influence of Source Constellations on UT1 Derived from IVS INT1 Sessions (2012). International VLBI Service for Geodesy and Astrometry 2012 General Meeting Proceedings. NASA/CP-2012-217504, 395—399.