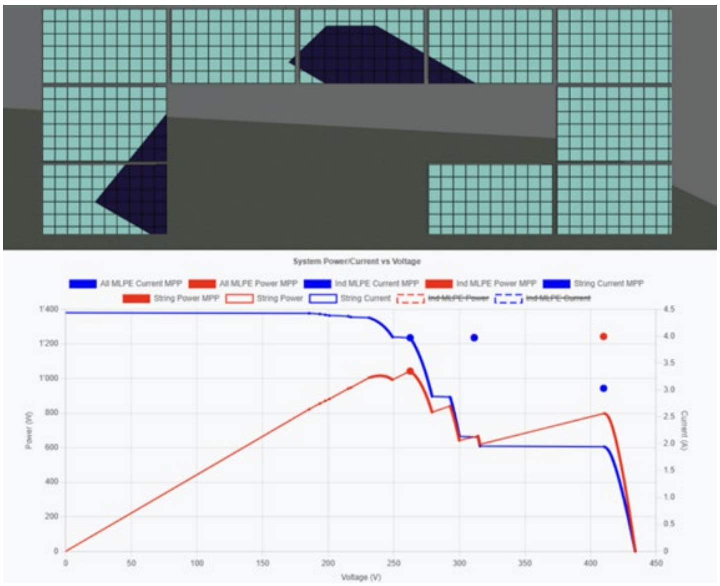
School of Engineering
Institut für Energiesysteme und
Fluid-Engineering (IEFE)

Mit Unterstützung von
energieschweiz

WebPVShade



Contract Number: SH/8100380-02-01-46
Recipient of Subsidy: ZHAW / IEFE
Duration: 01 July 2023 – 31 October 2025

Funding Agency Contact: Dr. Wieland Hintz
Recipient Contact: Prof. Dr. Franz Baumgartner

Date: 28 October 2025
Authors: Markus Klenk, Josias Ribi, Franz Baumgartner

Summary – Background and Achievement of Objectives

The technical premise—that the overall yield of PV systems can be significantly reduced by the shading of individual modules—is immediately clear not only to experts but also to less technically experienced users. Devices designed to minimize such shading losses are therefore in high demand, and the market volume for so-called optimizers or Module-Level Power Electronics (MLPE) has been growing internationally as well as in Switzerland.

The strong acceptance of such devices is also due to the generally positive effects attributed to them. This, in turn, created the expectation of precisely quantifying the achievable loss reduction in practice and led to roughly a dozen publicly funded national projects and internal student theses (project, bachelor's, and master's) carried out at ZHAW in the group of Prof. Dr. Baumgartner. These projects and theses addressed both the investigation of the MLPE hardware components and the relationships between shading, performance, and device behavior in comparison with conventional string inverters (SINV). In addition, mixed systems with MLPE installed only on selected modules (Independent MLPE – IndMLPE) were examined.

In the course of this work, software tools were developed in a predecessor project to simulate and visualize the corresponding yields and losses. These results indicated that the reduction of yield losses by MLPE in typical shading scenarios is, at least in most cases, significantly lower than assumed by MLPE manufacturers and even by many PV planners.

The growing national and international interest in these results led to the decision to further develop and extend the existing software tools. At the core were a MATLAB-based simulation software called "PVShade"—an expert system for simulation—and a web-based interactive tool for the visualization of typical shading scenarios, "WebPVShade." The latter presents the results of the PVShade expert system in a visualized form on a publicly accessible platform, where users can explore and interpret the calculated outcomes in detail.

This further development was carried out within the present project "WebPVShade," which was funded by EnergieSchweiz. From a technical perspective, two main objectives were pursued. On the one hand, it should be possible to capture more complex scenarios of PV generator shading, as the benefits of optimizers only come into full effect in these cases. The original aim was to enable companies interested in selling optimizers to pay to present scenarios that demonstrate the benefits of their devices. On the other hand, the web-based platform was to be expanded to support the public presentation of the results achieved. As the project progressed, it became increasingly clear that the potential for loss reduction through optimizers remained limited, even in complex shading scenarios. This, combined with the general financial situation in the photovoltaic industry—such as the slowdown in inverter sales after the decline in electricity price peaks and the price pressure from Asian suppliers—made the intended participation of industrial partners difficult. The planned contributions from industry were therefore replaced by internal ZHAW funding, allowing the project to continue.

The realization of customer-specific orders originally planned in the project proposal was replaced by the implementation of self-defined, more complex shading scenarios. A substantial portion of the work was also supported through the involvement of students in bachelor's and project theses, as well as IT students with appropriate expertise who contributed on an hourly basis.

It became apparent that the MATLAB tool increasingly reached its limits when handling more complex scenarios and required a fundamental revision. Based on the experiences and existing code base, a significantly improved, more professional, and much faster version was therefore developed in Java, which also provides easier accessibility for future developers.

It should be noted that no commercial PV design software currently available relies on the individual operating points of optimizers within a system. In the research community, however, institutions such as Fraunhofer ISE and TU Delft are investigating this topic and have published relevant results, as shown in IEA PVPS T13-27 (2024).

Overall, the technical objectives set for the project were fully achieved. Through the WebPVShade website, the performance of shaded PV systems can now be demonstrated to practitioners in an accessible and comprehensible way. Although any generalized depiction of the calculated MLPE performance gain based on representative examples will always remain limited, the use of the “Shading Index” as a quantitative measure of system shading enables a nearly systematic representation. On the website, this approach follows the practical rule of thumb: a higher Shading Index corresponds to a greater potential benefit from optimizer systems. The new PVShade tool also enables the rapid setup and simulation of very complex scenarios—annual yield simulations can now be completed in under ten minutes, representing an acceleration by a factor of 10 to 20 compared to the former MATLAB program.

Preview of Key Findings

The results show, on the one hand, that MLPE systems can achieve slightly higher yields even under moderate partial shading and that their benefit increases with the degree of shading. On the other hand, even under very strong partial shading, the benefit remains far below the generally expected effect and, in the cases presented here, lies within the mid single-digit percentage range.

With IndMLPE systems, effects similar to those of fully equipped MLPE systems can be achieved. It is evident that the number and placement of MLPEs in such systems play a major role.

The web-based visualization tool WebPVShade was redeveloped from scratch and now offers a wide range of options for displaying detailed performance results for every simulated time step, including module shading down to cell level and the corresponding module IV curves — which was previously not possible.

The fundamental problem of presenting general results using exemplary scenarios inevitably remains, and the clarity of presentation does not necessarily improve with a larger number of selectable options. Nevertheless, a default selection of representative scenarios, each with identical string configurations but varying Shading Index, now provides a quasi-systematic overview as shading complexity increases. Specific scenarios can always be simulated and added within PVShade.

The faster calculation capability also opens the door for future interactive configuration by users through the web tool. Although the basic structure is already straightforward, further resources would be required to develop a fully user-friendly interface. Overall, the intended objectives of WebPVShade were fully achieved.

Potentially, the PVShade software also allows for future investigations of shade-tolerant modules. For conventional module designs, this is already feasible. However, for newer module types that integrate the bypass diodes directly within the solar cells, the reverse-bias characteristics of the cells would need to be considered, which has not yet been implemented. This would also enable direct comparison between the approaches of MLPE-based and module-integrated shading-loss reduction.

Take-Home Messages / Lessons Learned from the Project

- MLPEs already provide a slightly higher yield under comparatively light partial shading than systems with string inverters. As expected, the advantage of MLPEs increases as shading becomes stronger.
- For large PV systems that are only weakly or not shaded at all, the cumulative DC/DC converter losses of MLPE systems can actually result in slightly lower yields compared with string inverter systems using simple connectors between modules.
- The gain from MLPEs over string inverters remains, even under heavy partial shading, within the mid single-digit percentage range for the investigated scenarios.
- Both software tools — the performance-calculation engine and the web-visualization interface — meet their design expectations. Thanks to the redevelopment, they now also provide the foundation for extended analyses, such as the study of shade-tolerant modules and the comparison of this approach with MLPE effects.
- Both tools were made significantly more efficient, primarily through the targeted collaboration with IT specialists, who contributed their technical expertise in close exchange with experienced PV experts. This underlines the value of interdisciplinary teams.

Summary of the Results of the Subsidized Project

Revised Final Versions of the "PVShade" Program and the Interactive Web Tool ("WebPVShade")

At the outset, it should be emphasized—since this has often been misunderstood—that the PVShade program was never intended to compete with established PV-simulation tools such as PVSyst. Although it performs sophisticated physical calculations, certain effects—such as series resistances in the string cabling or surface reflections at the module front glass—are intentionally not considered. Including such effects would distort the results related to partial shading, which are the central focus of the tool. Nevertheless, the simulations produce physically meaningful and realistic yield estimates.

The simulations are based on published data (e.g., module datasheets) from which the characteristic IV curves are derived, and on experimentally determined properties of hardware components such as the MLPEs.

Throughout earlier projects and within this one, the findings were continuously incorporated into the development of the software. A detailed chronological reconstruction of the approach can be found in previous publications and reports; to avoid repetition, the following focuses on the revised and final versions of PVShade and WebPVShade.

The results shown up to the last interim report were obtained using a MATLAB-based PVShade code, which implemented relatively simple shading scenarios. However, this was not fully satisfactory for evaluating MLPE performance, since their positive effect was expected to appear primarily in complex shading conditions.

As the MATLAB program evolved over time, it accumulated structural complexity that made it unsuitable for flexible adaptation or the inclusion of more advanced scenarios. The scenarios and components were hard-coded, variations required saving the entire program as a single block. Improvements to the general code flow had to be manually propagated to every stored scenario to maintain consistency. The MATLAB-based implementation, together with its organically grown architecture, resulted in cumbersome computational pathways, long calculation times, and difficulties in setting up shading configurations.

To address these issues, it was decided to retain the core computational concepts but rebuild the entire structure in Java, a widely used and more accessible programming language. This allowed the program to be reorganized in a clean, modular way that removed bottlenecks and weaknesses identified in the MATLAB version. Where possible, the new implementation makes use of efficient, well-established computational libraries. Computation speed was improved by a factor of 10 to 20, with typical annual simulations now taking around ten minutes. Once the results are transferred to the web frontend, no further calculation is required: users can interactively explore data, shading maps, and characteristic curves for every simulated time step via intuitive controls.

Further details regarding the modifications to the new program code are documented in Appendix 1.

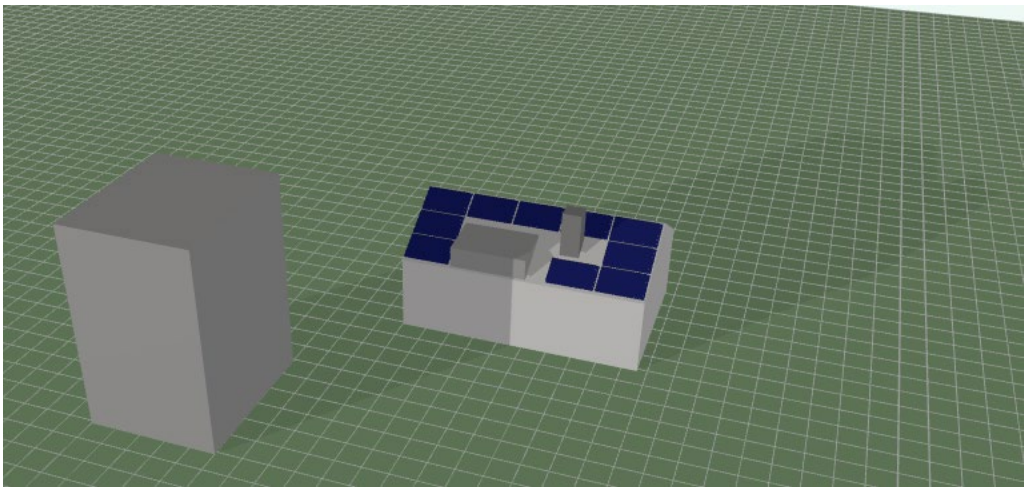


Figure 1: Example illustration of the shadow cast by an arbitrarily placed object, such as a dormer window or chimney, on a rooftop PV system..

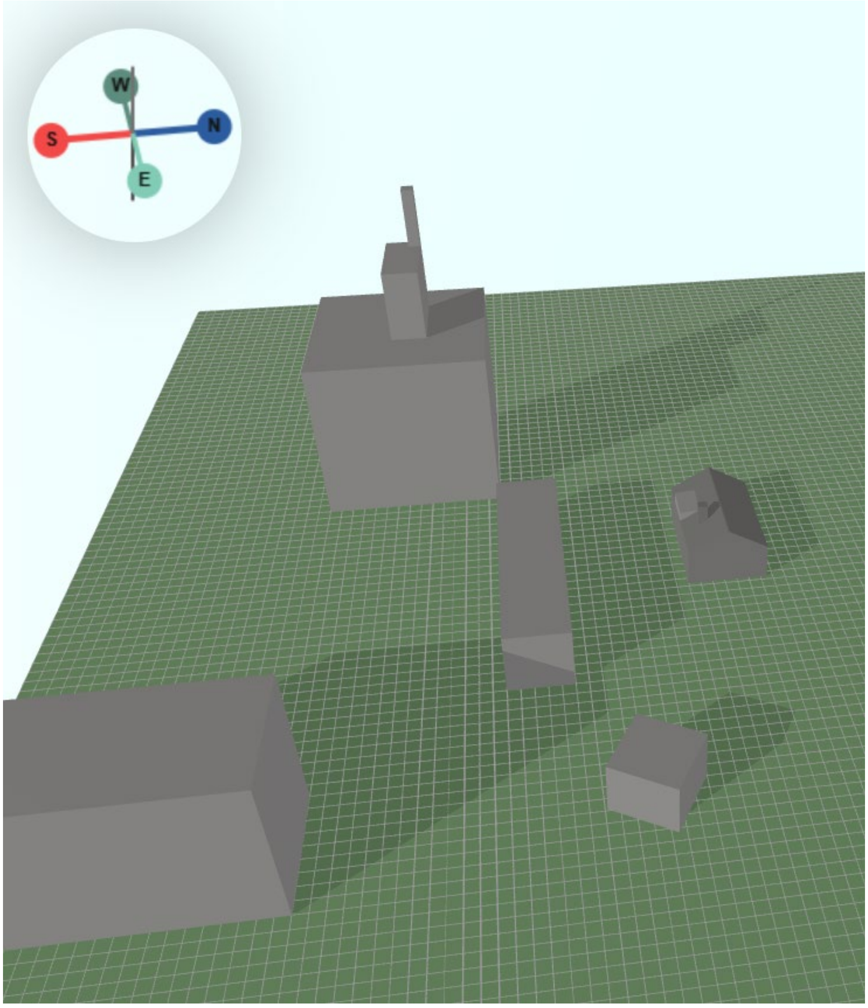


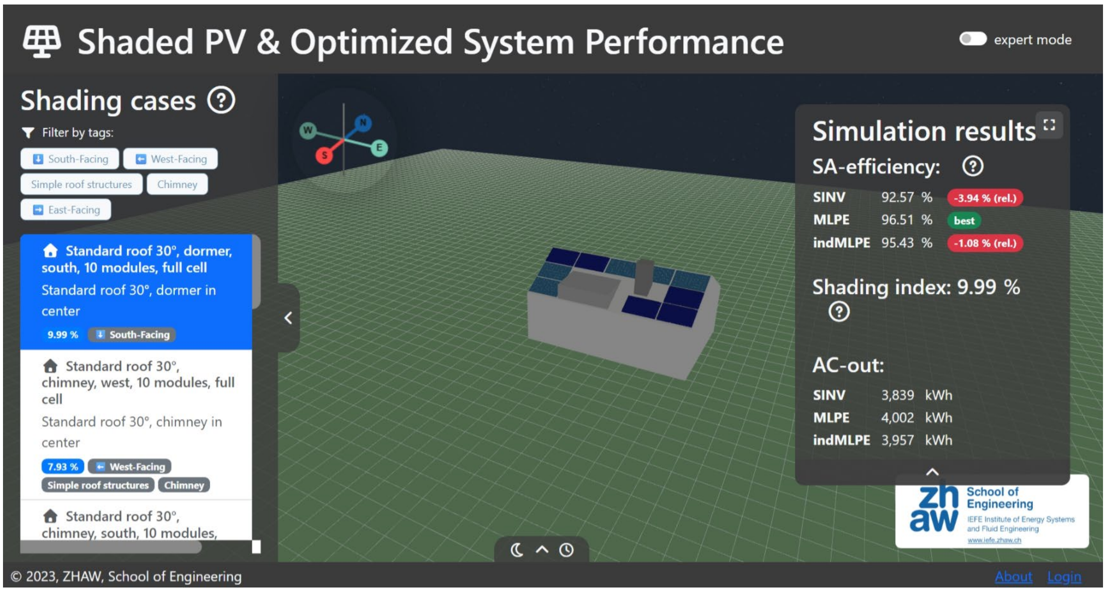
Figure 2: Shading in an urban environment. Combining multiple objects is easy to do and, in principle, also allows a horizon to be inserted to represent distant shading.

The creation of the new implementation was labor-intensive, but it now allows easy selection of components and the construction of complex shading scenarios at much higher computational speed. Figure 1 illustrates, as an example, the shadow cast by an arbitrarily positioned object—such as a dormer or a chimney—on a PV rooftop installation.

By virtue of the simplified structure, objects can now be freely combined and positioned. Figure 2 demonstrates this with an imaginary shading scene in an urban environment. The software can, in principle, also include a horizon line to account for distant shading.

As before, the program provides output of simulated scenarios through a user interface (UI) in which typically an entire year is simulated. By default, weather data for the year 2018 at the Zurich location are used, although other sites and meteorological datasets can easily be integrated.

Figure 3 shows the output of the annual simulations within the WebPVShade tool for a typical PV roof with chimney and dormer. The display compares three variants: String inverter (SINV), fully optimized (allMLPE), and partially optimized (IndMLPE) systems. The right-hand panel, labeled "Simulation Results," presents the best-performing configuration as well as the relative losses of the other two options. It also lists the Shading-Adapted Efficiency and the corresponding AC yield of each variant. On the left, the interface provides a scrollable selection of scenarios with tags for quick navigation.



Standard roof 30°, dormer, south, 10 modules, full cell
Standard roof 30°, dormer in center
9.99 % South-Facing

Standard roof 30°, chimney, west, 10 modules, full cell
Standard roof 30°, chimney in center
7.93 % West-Facing Simple roof structures Chimney

Standard roof 30°, chimney, south, 10 modules,

SA-efficiency:

SINV	92.57 %	-3.94 % (rel.)
MLPE	96.51 %	best
indMLPE	95.43 %	-1.08 % (rel.)

Shading index: 9.99 %

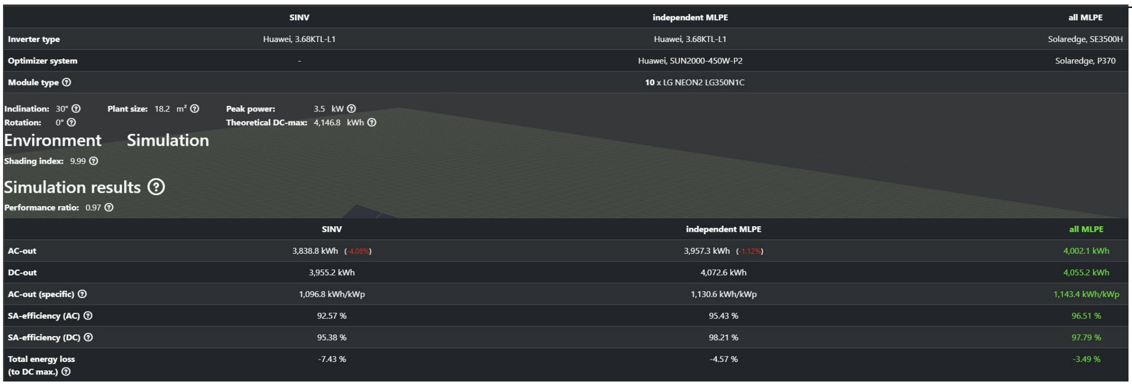
AC-out:

SINV	3,839 kWh
MLPE	4,002 kWh
indMLPE	3,957 kWh

© 2023, ZHAW, School of Engineering

Figure 3: Output of the results (SINV MLPE and IndMLPE) of an annual simulation for the PV roof shown. The right-hand window shows the best variant, as well as the relative losses of the other variants, "Shading Index" and yield. The left-hand window shows the scenario selection window.

Modules shown in lighter color indicate those equipped with IndMLPE devices. Clicking on such a module opens a detailed window (Figure 4) with additional information about the layout, components, and system yields.



	SINV	independent MLPE	all MLPE
Inverter type	Huawei, 3.68KTL-L1	Huawei, 3.68KTL-L1	Solaredge, SE3500H
Optimizer system	-	Huawei, SUN2000-450W-P2	Solaredge, P370
Module type ?		**10** x LG NEON2 LG350N1C	

Inclination: 30° ? **Plant size:** 18.2 m² ? **Peak power:** 3.5 kW ?
Rotation: 0° ? **Theoretical DC-max:** 4,146.8 kWh ?

Environment Simulation

Shading index: 9.99 ?

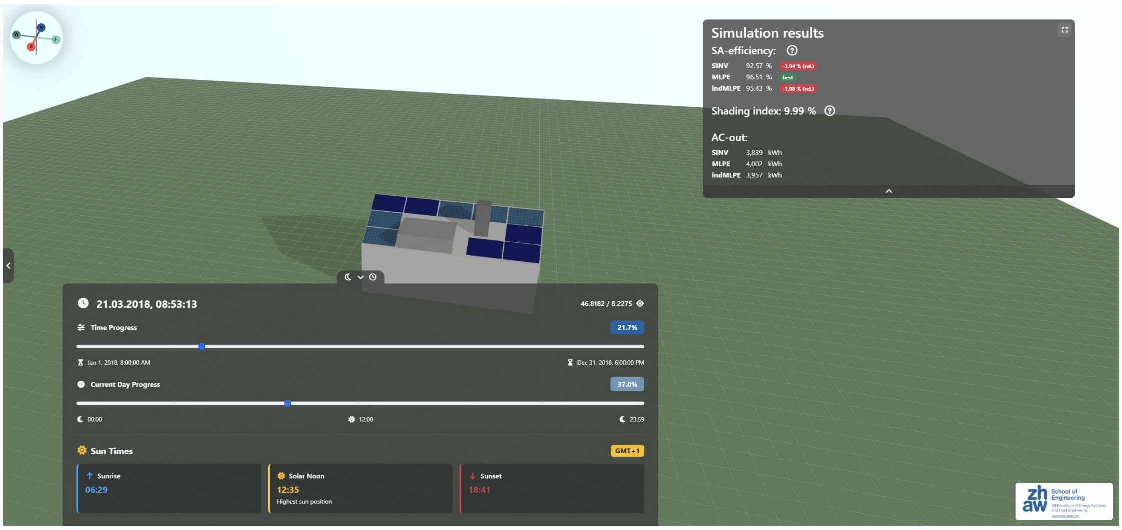
Simulation results ?

Performance ratio: 0.97 ?

	SINV	independent MLPE	all MLPE
AC-out	3,838.8 kWh (-4.08%)	3,957.3 kWh (-1.12%)	4,002.1 kWh
DC-out	3,955.2 kWh	4,072.6 kWh	4,055.2 kWh
AC-out (specific) ?	1,096.8 kWh/kWp	1,130.6 kWh/kWp	1,143.4 kWh/kWp
SA-efficiency (AC) ?	92.57 %	95.43 %	96.51 %
SA-efficiency (DC) ?	95.38 %	98.21 %	97.79 %
Total energy loss (to DC max.) ?	-7.43 %	-4.57 %	-3.49 %

Figure 4: Details on components, layout, and yields can be accessed directly.

At the bottom of the user interface, a control element allows users to adjust date and time continuously via sliders and to observe the resulting sun position and shadow projection (Figure 5)



Simulation results

SA-efficiency: ?

SINV	92.57 %	-3.94 % (rel.)
MLPE	96.51 %	best
indMLPE	95.43 %	-1.08 % (rel.)

Shading index: 9.99 % ?

AC-out:

SINV	3,839 kWh
MLPE	4,002 kWh
indMLPE	3,957 kWh

21.03.2018, 08:53:13 46.8182 / 8.2275

Time Progress 21.7%

Jan 1, 2018, 8:00:00 AM Dec 31, 2018, 6:00:00 PM

Current Day Progress 37.0%

00:00 12:00 23:59

Sun Times GMT+1

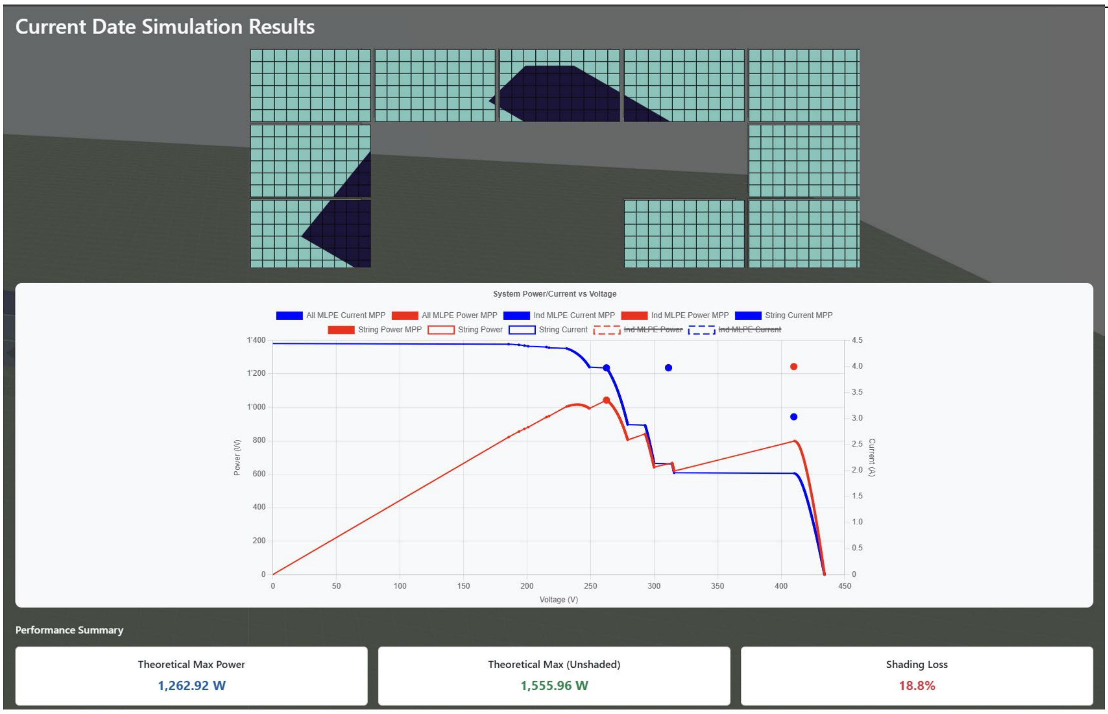
Sunrise 06:29

Solar Noon 12:35 Highest sun position

Sunset 18:41

Figure 5: Date and time adjustable via sliders with resulting shadow cast.

In the “Expert Mode”, additional sliders enable continuous visualization of the module-field shading at any given moment and of the corresponding system I-V and power curves (Figure 6).



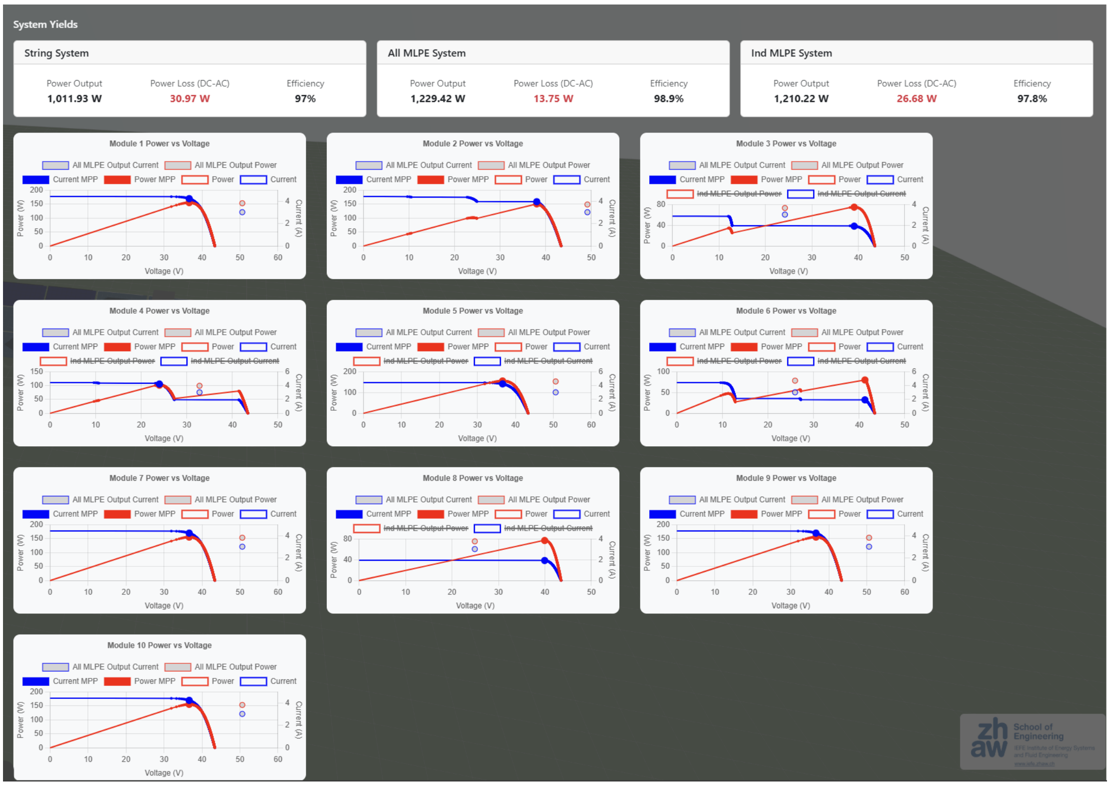
Current Date Simulation Results

Performance Summary

Theoretical Max Power	Theoretical Max (Unshaded)	Shading Loss
1,262.92 W	1,555.96 W	18.8%

Figure 6: At any selected point in time, the shading pattern on the modules and the system characteristic curve (IV and power) can be displayed.

For each time step, the I-V curves of individual modules can also be displayed (Figure 7). By moving the time and date sliders, the user can “play through” the shading situations and resulting curves like a film, tracing effects step by step at the module or even cell level.

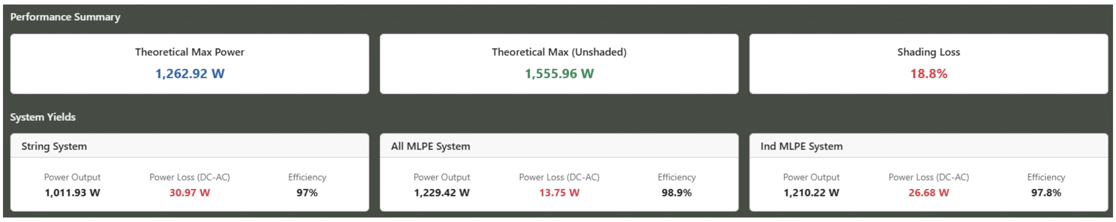


System Yields

	String System	All MLPE System	Ind MLPE System
Power Output	1,011.93 W	1,229.42 W	1,210.22 W
Power Loss (DC-AC)	30.97 W	13.75 W	26.68 W
Efficiency	97%	98.9%	97.8%

Figure 7: Characteristic curves of the individual modules can be output for any point in time.

Moreover, characteristic system-level performance parameters can be retrieved for every simulated instant (Figure 8).



Performance Summary

Theoretical Max Power	Theoretical Max (Unshaded)	Shading Loss
1,262.92 W	1,555.96 W	18.8%

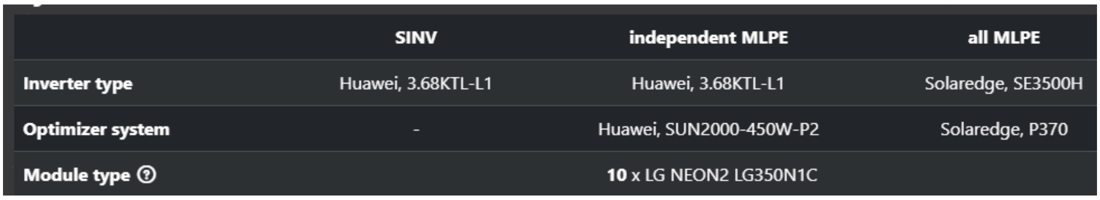
System Yields

	Power Output	Power Loss (DC-AC)	Efficiency
String System	1,011.93 W	30.97 W	97%
All MLPE System	1,229.42 W	13.75 W	98.9%
Ind MLPE System	1,210.22 W	26.68 W	97.8%

Figure 8: Characteristic system data at each point in time.

Example Demonstration – Variation of Shading

In the new tool, comparative scenarios can now be created and simulated easily and quickly. As an example, a configuration of PV modules, inverters, and MLPEs with identical basic setup was analyzed under several shading scenarios using annual simulations. Ten modules (350 Wp each) were simulated in three configurations: String inverter (SINV), fully optimized (allMLPE), and partially optimized (IndMLPE). The components used correspond to those shown in Figure 9.



	SINV	independent MLPE	all MLPE
Inverter type	Huawei, 3.68KTL-L1	Huawei, 3.68KTL-L1	Solaredge, SE3500H
Optimizer system	-	Huawei, SUN2000-450W-P2	Solaredge, P370
Module type ?		**10** x LG NEON2 LG350N1C	

Figure 9: Components used in the example variation of shading situations shown below.

String length and the number of IndMLPE units were chosen such that all three configurations were technically realizable with the available components. This selection is not necessarily optimal for every setup; for instance, the string length in the all-MLPE case is rather short, and the number of IndMLPEs might be chosen differently in a real installation. Nevertheless, keeping the number of components constant was intentional to avoid quantity-related effects.

As an example, Figure 11 shows a south-facing roof with a 30° pitch and the transition to a flat roof with varying shading indices. The shading index can be used to assign a factor to the scenarios to quantify the degree of shading, which allows for a certain degree of systematization.

$$SI_{DC,Max}[\%] = 1 - \frac{\sum_{t=0}^{T=1yr.} \sum_{i=1}^{k=nModules} P_{Mod,i,MPP,shaded}}{\sum_{t=0}^{T=1yr.} \sum_{i=1}^{k=nModules} P_{Mod,i,MPP,unshaded}} \cdot 100$$

Figure 1: Shading Index, which quantifies the degree of partial shading as the ratio between the sum of all modules' maximum possible DC contributions in the shaded system and those of the unshaded reference system (see IEA PVPS T13-27, 2024 in References).

Figure 11 shows systems arranged from top to bottom with decreasing Shading Index, while keeping the string layout identical. Only the position of the IndMLPE-equipped modules (highlighted in lighter blue) changes slightly between the flat-roof simulations; the total number remains the same.

The Shading Index ranges from about 16 % — representing very heavy shading under real conditions — to 0 % for the unshaded system, whose AC yield (with SINV) serves as the reference for the others.

On the right-hand side, the relative difference in “Shading-Adapted Efficiency” compared with the best configuration (highlighted in green) is indicated. This allows the user to identify the best-performing system and the relative losses of the alternative options. Enabling such flexible scenario definitions was a central goal of the new PVShade simulation tool.

Under strong to extreme partial shading (upper examples in Figure 11), systems with MLPEs perform, as expected, better than string-inverter systems.

In contrast, in unshaded or only slightly shaded cases, the string-inverter configuration performs best, since the advantages of MLPEs do not manifest there, while their individual DC/DC conversion losses slightly reduce overall efficiency. For longer strings, this effect — and thus the advantage of string inverters — would be even more pronounced.

The crossover point where MLPE systems begin to outperform SINV systems occurs, in these examples, at a Shading Index of roughly 1.5 %. Although the Shading Index provides a useful indicator for assessing shading severity, it cannot yield a universal threshold valid for all possible configurations. Considering previous investigations (e.g. IEA PVPS T13-27, Section 7.1), favorable conditions for MLPE or SINV configurations are typically found for Shading Index values between ≈ 2 % and 5 %.

It is worth noting that the topic of shading optimization predates the widespread adoption of MLPEs. System design — especially careful module placement — has a decisive influence, and experienced PV planners are well aware of this. In practice, most installations are likely to fall within the lower portion of this Shading Index range.

The relative difference between IndMLPE and allMLPE configurations remains remarkably small: below 1 % loss for mild shading and under 3 % even for heavy shading. This minimal deviation demonstrates that equipping only the critical modules in a PV system with MLPEs can deliver substantial advantages over SINV configurations, without needing full-system deployment.

It should again be noted that the IndMLPE layouts here were constrained by the fixed component sets and thus not fully optimized; the benefit depends strongly on the number and placement of MLPEs.

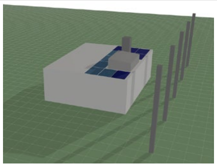
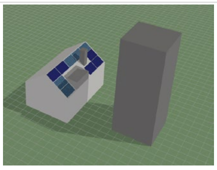
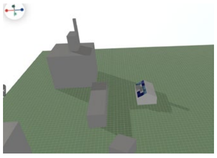
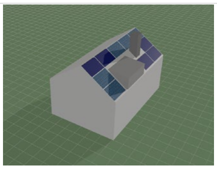
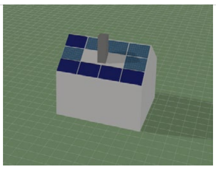
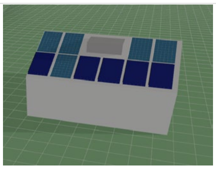
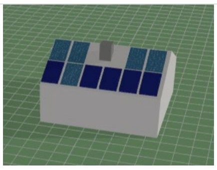
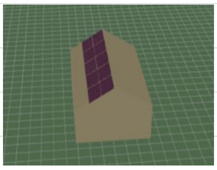
Flachdach mit Flaggenmasten (Süd) und Aufbauten				
	**SI: 15.6 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	90.4	95.5	93.0
	SA-η rel. [%]	-5.5	0.0	-2.5
	AC-kWh rel. zu Ref. [%]	57.7	61.1	59.5
Grosses Verschattungsobjekt nahe vor Süd-Dach 30° mit Kamin und Gaube				
	**SI: 12.7 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	91.3	95.9	94.1
	SA-η rel. to best [%]	-4.6	0.0	-1.8
	AC-kWh rel. zu Referenz [%]	73.9	77.6	76.2
Städtische Umgebung vor Süd-Dach 30° mit Kamin und Gaube				
	**SI: 10.2 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	93.1	96.4	95.3
	SA-η rel. to best [%]	-3.3	0.0	-1.1
	AC-kWh rel. zu Referenz [%]	83.5	86.5	85.5
Süd-Dach 30° mit Kamin und Gaube				
	**SI: 10.0 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	93.1	96.4	95.3
	SA-η rel. to best [%]	-3.9	0.0	-1.0
	AC-kWh rel. zu Referenz [%]	84.7	88.3	87.4
Süd-Dach 30° mit Kamin zentral				
	**SI: 7.9 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	94.1	96.5	96.1
	SA-η rel. to best [%]	-2.4	0.0	-0.5
	AC-kWh rel. zu Referenz [%]	88.7	90.9	90.5
Süd-Dach 30° mit Gaube in Giebelnähe				
	**SI: 1.5 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	96.8	96.7	96.5
	SA-η rel. to best [%]	0.0	-0.1	-0.3
	AC-kWh rel. zu Referenz [%]	97.7	97.6	97.4
Süd-Dach 30° mit Kamin in Giebelnähe				
	**SI: 1.3 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	97.4	96.7	96.5
	SA-η rel. to best [%]	0.0	-0.3	-0.5
	AC-kWh rel. zu Referenz [%]	98.1	97.7	97.6
Süd-Dach 30° (Referenz mit SINV)				
	**SI: 0 %**	**SINV**	**allMLPE**	**indMLPE**
	SA- η [%]:	97.6	96.7	96.5
	SA-η rel. to best [%]	0.0	-0.9	-1.0
	AC-kWh rel. zu Referenz [%]	100.0	99.1	99.0

Figure 11: Example of an annual simulation with varying shading for the same string configuration. The relative difference in shading-adapted efficiency compared to the best of the three variants (green) is highlighted in color (right).

As expected, the advantage of MLPE configurations increases with growing shading complexity. Nevertheless, even under the strongest simulated shading (highest Shading Index values), the MLPE advantage remains around 5 % relative to the SINV system. The benefit is therefore real but still far below the often-advertised “up to 30 % gain.”

A key goal of the project was to make the effect of MLPEs compared with standard inverters intuitive and visually accessible, allowing users to relate the simulated outcomes to their own or comparable systems. Through the default scenarios implemented in WebPVShade, users can now click through the variants shown above, view the simulations, shading progressions, and intermediate results directly. Even though it is impossible to cover every conceivable case, these variations already provide a solid foundation for conveying the MLPE benefit to potential customers, practitioners, and experts. Extensions can easily be created and added at any time.

Further Application Options

The primary purpose of the PVShade software was — and still is — the comparative analysis of systems using SINV, MLPE, and IndMLPE configurations.

However, with the new PVShade tool, other analytical tasks can also be addressed quickly and efficiently. For instance, Figure 12 illustrates the shadow progression for specific times of year and day, which can now be visualized even for highly complex configurations.

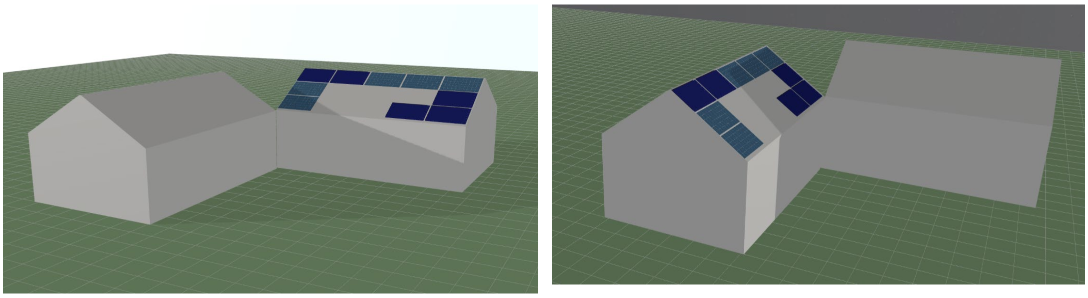


Figure 12: Analysis of shadow progression for specific arrangements, using the example of an L-shaped arrangement of two buildings (PV south-facing on the left, PV east-facing on the right).

The tool’s high spatial resolution (Figure 13) makes it particularly suitable for comparing different module types, including the so-called shade-tolerant modules.

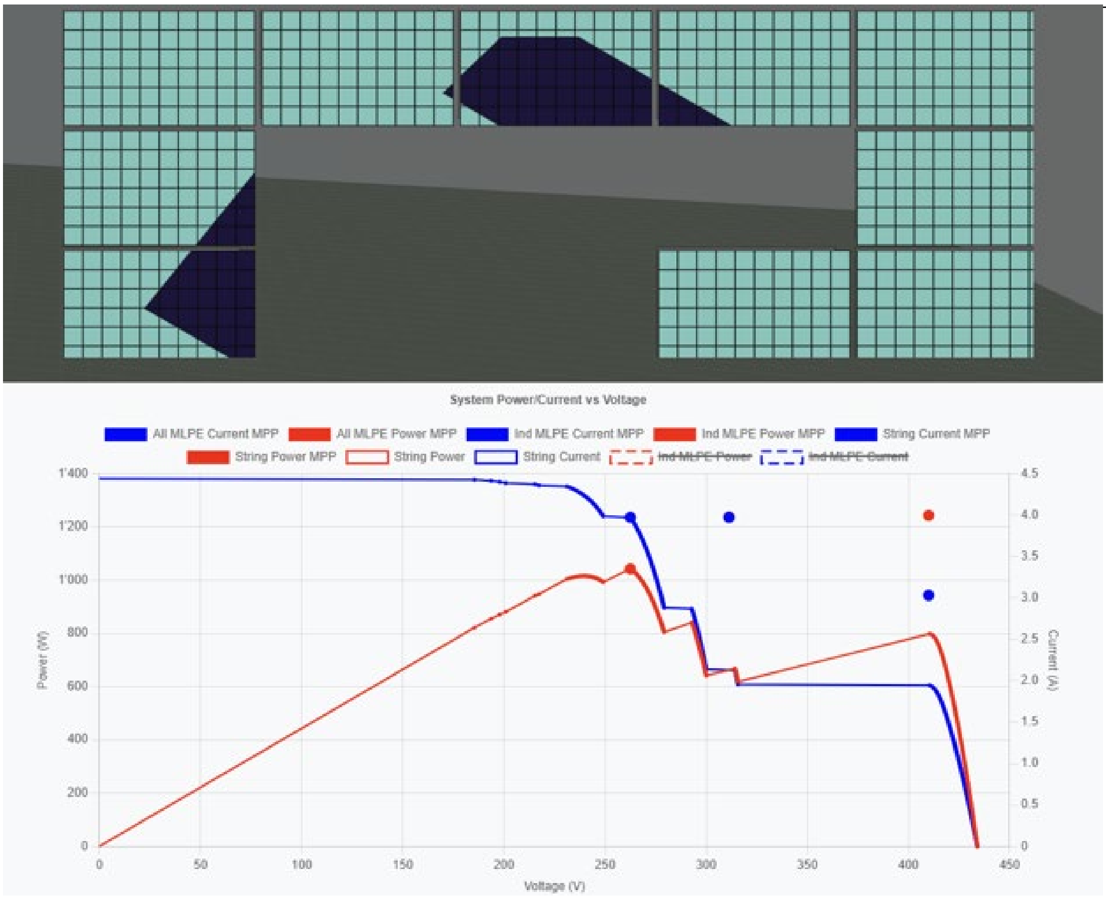


Figure 13: (Zoom from Figure 6) Simulation of individual modules with spatial resolution at cell level. This makes it ideal for simulating the effect of different module types, including shade-resistant types, in partially shaded systems.

This capability has already been used to a limited extent to simulate today's common "Butterfly" module configuration, which is promoted for its reduced sensitivity to shading.

Thematically, this naturally connects to previous research, as both MLPE and shade-tolerant module designs aim to minimize losses caused by partial shading. Given the strong interest in this topic — reflected especially in the market share of MLPE devices — it is logical to extend these analyses to shade-tolerant modules and even to hybrid combinations of both approaches.

With the presented tool, this is already possible for certain module types, though not yet systematically implemented for all designs. For example, the simulation of the mentioned "Butterfly" layout is already supported.

In that case, the reverse-bias characteristic of the solar cells is simplified — resulting in slightly smoother I-V curves instead of the distinct plateaus typical between diode steps (as shown in Figure 13). For the currently most relevant shade-tolerant modules, based on solar cells with "integrated bypass diodes", the reverse-bias behavior would still need to be modeled explicitly. While this was previously impractical due to computational overhead, it should now be feasible with the new, more efficient software generation.

Finally, it should be noted that the systematic presentation in Figure 11 was designed to capture typical systems, but the now-achievable level of complexity extends far beyond those examples. The program can now also simulate unconventional or advanced PV configurations, such as façade-integrated systems (Figure 14) or solar trees deployed in alpine environments.

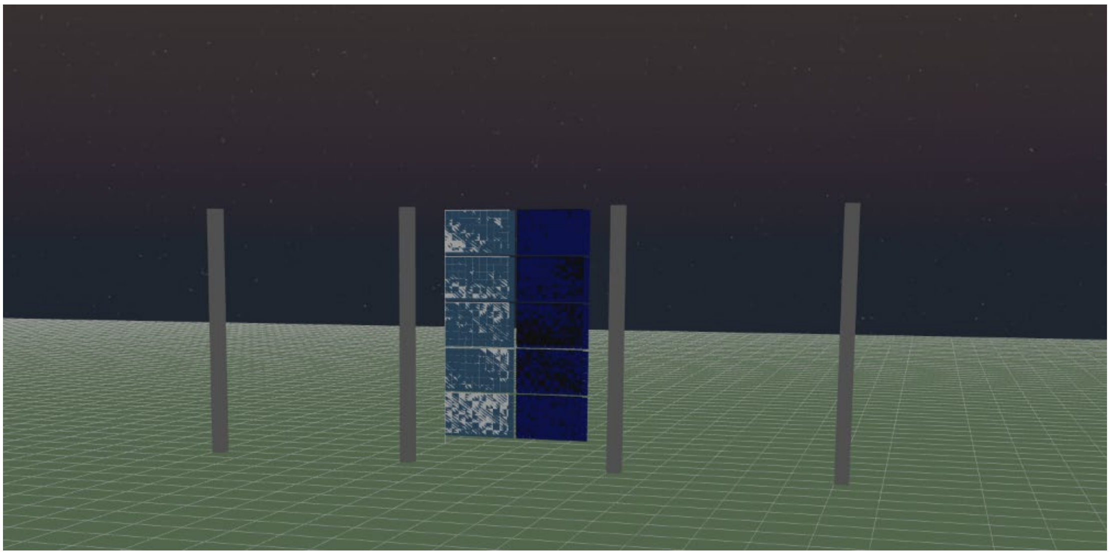


Figure 14: The program now also allows the simulation of more complex or unusual systems than those shown in Figure 11, which depicts typical applications.

Conclusion and Outlook

One of the project's objectives was to calculate and present application-oriented examples of shaded PV systems on a publicly accessible website. To achieve this, it was necessary to improve the existing tools developed at ZHAW.

In line with the expectation that PV systems with stronger shading would gain a greater advantage from MLPE compared with string inverters, numerous illustrative examples with varying and even severe shading conditions were created. A key outcome, however, was that even under realistically strong shading, the annual energy gain achieved through MLPE remains within the mid single-digit percentage range. In most practical cases, module layout can be designed such that the Shading Index remains below roughly 2 – 5 %, where no significant yield advantage for MLPE exists.

Accurate quantitative comparison of systems using SINV, MLPE, and IndMLPE configurations in shading scenarios of different complexity required the improved PVShade tool. Although this was already possible in principle with the MATLAB-based version, the objective was only fully achieved with the new Java implementation, which enables much faster and more flexible simulation of complex scenarios. Additional components and configurations can now be added at any time, and the code — written in the widely used Java language — is readily accessible to future developers. At present, one remaining limitation is that all modules of a string are still assumed to lie in a single plane.

Another major goal of the project was to present comparative results of SINV, MLPE, and IndMLPE systems on a publicly accessible platform (WebPVShade) in a visually comprehensible way.

Fundamentally, the challenge remains that such results must be shown in the form of examples, since this is the only way to communicate them clearly to users. Consequently, outcomes depend on specific shading situations, making direct transfer to individual systems more difficult. Nevertheless, by displaying scenarios with varying Shading Index — as demonstrated in this report — it is possible to provide an approximate systematic overview.

The illustrated examples are available on the public WebPVShade platform as default options and can be accessed interactively. The summarized presentation shown in this report could also be offered as a downloadable PDF document to give users an easy overview, while still allowing them to explore simulation details and inspect shadows or I-V curves for any chosen time. Ultimately, the relative performance differences between system types will remain the most relevant outcome — especially for non-specialist users.

The original project concept envisioned that manufacturers or distributors of MLPEs could commission the creation of complex shading scenarios to demonstrate the benefit of their products. Although there was some interest, this cooperation did not materialize. Economic conditions in the PV industry played a role, but another reason was that even under strong shading the results were not compelling enough to support the typically advertised yield gains. This is particularly true given that competitors are promoting similar figures.

The dissemination of project results was ensured through extensive activities by Prof. Dr. Franz Baumgartner and Cyrill Allenspach, including numerous scientific publications, presentations, and contributions to IEA Task 13. As a result, the core findings became well known both in the professional community and among the broader public. The work received wide attention and has been broadly discussed in the field. Its relevance remains high, as reflected by the significant market share of MLPEs, especially in Switzerland. Therefore, continued publication and dissemination of these insights are strongly recommended.

Looking ahead, the tool could be further developed and expanded. A particularly promising direction is the analysis of shade-tolerant modules, which, like MLPEs, aim to minimize partial-shading losses. Because the PVShade tool operates with spatially resolved cell-level shading analysis, it is ideally suited for such investigations. Some module types can already be modeled, while for the currently most relevant designs — those “with integrated bypass diodes” at cell level — further adaptations are required. This would allow direct comparison between optimizer-based and module-integrated mitigation approaches, including potential combinations of both.

Publications

Publications of student work at the ZHAW

Baumann, Linus; Widler, Alain; Projektarbeit ZHAW SoE SG EU im Herbstsemester 2023; *Maximum Power Point-Regelung von PV-Optimizer*

Baumann, Linus; Widler, Alain; Bachelorarbeit ZHAW SoE SG EU im Frühjahrsemester 2024; *Optimales Photovoltaik-Systemdesign mit Optimizer oder String-Inverter*

Iseni, Rijad; Martinelli, Robin; Projektarbeit ZHAW SoE SG EU im Herbstsemester 2024; *Das exakte Winterstrompotential von Solarbäumen in alpinen Photovoltaik Kraftwerken*

Iseni, Rijad; Martinelli, Robin; Bachelorarbeit ZHAW SoE SG EU im Frühjahrsemester 2024; *Performance Analyse von verschatteten Photovoltaik Kraftwerken*

Remark:

Project work 6 credits - workload per student 180 hours

Bachelor's thesis 12 credits - workload per student 360 hours

Publicly available publications

Baumgartner, Franz 2025, *Mehr Elektronik aufs Dach?*; Electrosuisse Bulletin.ch, 2025 No 6;

Baumgartner Franz; Allensbach Cyril; et al; 2024. *Performance of Partially Shaded PV Generators Operated by Optimized Power Electronics*. International Energy Agency Report IEA-PVPS T13-27:2024; ISBN 978-3-907281-64-2

Allenspach, Cyril Armand; Carigiet, Fabian; Bänziger, Arturo; Schneider, Andrin; Baumgartner, Franz, 2022. *Power conditioner efficiencies and annual performance analyses with partially shaded photovoltaic generators using indoor measurements and shading simulations*. Solar RRL. 7(8), S. 2200596. Verfügbar unter: https://doi.org/10.1002/solr.202200596

Baumgartner, Franz P.; Klenk, Markus; Widler, Adrian; Baumann, Linus, 2024. *MPP tracking losses of module level power electronics at partial module shading*. In:Proceedings of the 41st European Photovoltaic Solar Energy Conference and Exhibition. (EU PVSEC), Vienna, Austria, 23-27 September 2024. Munich: WIP. S. 020289-001-020289-006. Verfügbar unter: https://doi.org/10.4229/EUPVSEC2024/3EO.1.5

Baumgartner, Franz; Baumann, Linus; Widler, Alain; Allenspach, Cyril; *Weniger Komplexität und mehr Zuverlässigkeit hinter jedem Solarmodul*; 18. Symposium Energieinnovation, 14.-16.02.2024, an der Technischen Universität Graz/Austria

Baumgartner, Franz; Vogt, Roman; Allenspach, Cyril Armand; Carigiet, Fabian, 2021. *Performance analysis of shaded PV module power electronic systems*. In:Proceedings of the

38th EUPVSEC. 38th European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC), online, 6-10 September 2021. München: WIP.S. 650-654. Verfügbar unter: https://doi.org/10.4229/EUPVSEC20212021-4CO.3.1

Allenspach, Cyril Armand; Gonzalez de Echavarri Castro, Victor; Richter, Samuel; Meier, Christoph; Carigiet, Fabian; Baumgartner, Franz, 2020. *Module-level power electronics under indoor performance tests*. In: Proceedings of the 37th European Photovoltaic Solar Energy Conference and Exhibition. Online, 7-11 September 2020. WIP Renewable Energies. S. 1188-1194. Verfügbar unter: https://doi.org/10.4229/EUPVSEC20202020-4AV.3.8

Allenspach, Cyril Armand; Baumgartner, Franz, 2023. *Performance of power optimizer versus string inverter systems*. In: 21. Schweizer Photovoltaik-Tagung, Bern, Schweiz, 20.-21. März 2023. ZHAW Zürcher Hochschule für Angewandte Wissenschaften. Verfügbar unter: https://doi.org/10.21256/zhaw-27169

BFE-Projekt: EFFPVSHADE – *Effizienzanalyse von dezentraler Photovoltaik-Leistungselektronik bei Teilbeschattung* (SI/502247-01).

Appendix 1

Technical Report “WebPVShade V2 Beta” 20.10.2025

By Josias Ribi
WebPVShade V2 Beta

Summary

This appendix documents a comprehensive revision and modernization of the WebPVShade application after a year of intensive further development. The project has successfully made the transition from a complex, fragmented technology stack to a uniform, modern, and highly maintainable architecture.
Key result: Complete standardization of the backend. The most important result of this development cycle is the complete migration of the PV simulation logic from the old MATLAB code to the Java Spring Boot backend. This architectural decision eliminates technical hurdles and complexity in operation. The previous architecture was fragmented, complex, and difficult to maintain. It was based on outdated approaches that caused significant problems in program execution:

Examples of problems:

- The simulation was based on outdated and poorly maintained MATLAB code.
- Programming quality: The existing MATLAB code was difficult to understand, adapt, and extend.
- Performance issues: The simulations were extremely slow and had massive deficiencies in terms of memory usage.
- System crashes: Longer-running simulations often crashed due to memory problems.
- No clean integration: The MATLAB code was separate from the main application. Complex integration layers were necessary.

Problems with frontend development
Outdated AngularJS: The frontend was based on an old version of AngularJS without modern, signal-based state management. This made maintenance and extensibility difficult.
Complexity in build and deployment

- Java “wrapped around” Node: The front end was launched via a Java wrapper that controlled Node.js—unnecessary complexity.

Complex developer setups: Developers needed specific versions of MATLAB, Java, and Node.js locally.
Not operating system independent: Some dependencies and build steps were platform-specific, which made cross-platform development difficult.

- Infrastructure limitations
- Limited scalability: The architecture was not cloud-friendly.
- Poor DevOps experience: Setup and maintenance were based on many custom-built, non-standardized solutions.

Most of the problems were eliminated by a modernized, unified architecture.

Basic ideas

- Consolidation: Merge all simulation logic into a single, uniform code base.
- Modernization: Use current best practices and modern frameworks.
- Simplification: Reduce complexity at every level.
- Performance: Eliminate known bottlenecks and memory leaks.
- Maintainability: Design the code base so that future developers can easily understand and extend it.

Complete migration of the simulation engine to Java resulted in a huge improvement. The entire PV simulation logic was ported from the old MATLAB code to the Java Spring Boot backend. There are no longer any external dependencies on MATLAB or other separate runtime environments. All computing logic now runs within the application itself.
The performance improvements had a significant impact

- Annual simulations with a high level of detail previously resulted in many hours of computing time and frequent crashes. Now they take about 10 minutes, without crashes.
- Current performance gain: A factor of 10 to 100, depending on simulation complexity and scene size.
- Future potential: The new architecture enables further optimizations through

  o Multithreading: Parallel use of multiple CPU cores (potential ~5×)
  o Intelligent caching: Intermediate results for repeated partial calculations.
  Optimizations not implemented, but are potentially possible

Technical implementation

- Efficient Java algorithms implemented, MATLAB simulation logic
- Integrated into the Spring Boot application lifecycle
- Added comprehensive error handling and validation
- Built in asynchronous job processing for long-running simulations
- Cleanly implemented memory management and resource release
- Introduced unit tests to ensure simulation accuracy and prevent regressions.

Modernization of the frontend to the latest Angular
Switch from AngularJS to modern Angular (v19+) and use of Angular Signals for reactive state management.

- Better maintainability: Good additional tools and documentation. Widely used.
- Better developer experience: Improved IDE support, debugging tools, and clearer error messages.
- Future-proof: The current framework version is actively maintained and further developed.
- Signal-based state management via Angular Signals.
- Strict typing mode: TypeScript in “strict mode” enabled project-wide.
- Modern toolchain: Modern toolchain for faster builds and better developer experience.

Containerisierung und DevOps

- Complete Docker containerization of frontend and backend for production operation.
- No more specific Java or Node.js versions that developers have to maintain locally.
- Easy scaling and migration: Containers can be run on virtually any modern infrastructure.
- Simplified onboarding: Developers can basically start with just Java and Node.js.

Further improvements

- In addition to the major architectural changes, numerous improvements have been implemented:
- Improved visualization of simulation results: The display and interactive analysis of simulation results in the application have been significantly enhanced.
- Support for complex scenes: Simulation capabilities have been expanded to allow for more complex and realistic scenarios with greater depth of detail.
- Tag management: A robust tagging system has been introduced to better structure and organize simulations and projects.
- Further refinements: Various UI/UX improvements, performance optimizations, and bug fixes throughout the system.

Current limitations and trade-offs

As with any redesign, there are deliberate trade-offs and features that have not yet been implemented in this beta version. This section documents these openly.

Features not yet available in V2

Decorative house models in the web interface. The web interface of version 1 contained decorative house models for visualization. The new version now directly displays the real 3D model that is also used by the simulation engine. This means that the user's view and the physical basis of the simulation are 1:1 identical. Previously, there was a “fancy” visualization on the web that did not exactly match the MATLAB model.
Trade-off: The decorative (simplified) visualization was removed in favor of simulation accuracy and consistency. Future: Such decorative overlays can be added back later as an option.

Limited selection of PV hardware types. Not all PV module hardware types from V1 are currently supported in V2: During migration, the focus was on the correct core physics of the simulation. Additional hardware variants were given lower priority in order to meet the beta deadline. Approach: The new code base is highly modular and extensible, making it much easier to retrofit old (and new) hardware types than it was previously in MATLAB code. Next steps: Missing hardware types can be added gradually based on user needs and requirements.