

Synthesis of Solar Photovoltaic Systems: Optimal Sizing Comparison^{*}

Alessandro Trindade¹ and Lucas Cordeiro²

¹ Federal University of Amazonas, Brazil, alessandrotrindade@ufam.edu.br

² University of Manchester, UK, lucas.cordeiro@manchester.ac.uk

Abstract. In the current scenario, energy demand rises by 1.3% each year to 2040, and photovoltaic (PV) systems have emerged as an alternative to the fossil or nuclear fuel energy generation. The use of formal methods for PV systems is a new subject with significant research spanning only five years. Here we develop and evaluate an automated synthesis technique to obtain optimal sizing of PV systems based on Life Cycle Cost (LCC) analysis. The optimal solution is the lowest cost from a list of equipment that meets the electrical demands from a house, plus the replacement, operation, and maintenance costs over 20 years. We propose a variant of the counterexample guided inductive synthesis (CEGIS) approach with two phases linking the technical and cost analysis to obtain the PV sizing optimization. We advocate that our technique has various advantages if compared to off-the-shelf optimization tools available in the market for PV systems. Experimental results from seven case studies demonstrate that we can produce an optimal solution within an acceptable run-time; different software verifiers are evaluated to check performance and soundness. We also compare our approach with a commercial tool specialized in PV systems optimization. Both results are validated with commercial design software; furthermore, some real PV systems comparison are used to show our approach effectiveness.

Keywords: Formal Synthesis · Software Verification · Solar Photovoltaic Systems · Cyber-Physical Systems.

1 Introduction

Lack of access to clean and affordable energy is considered a core dimension of poverty [19]. Progress has been made worldwide; in particular, in 2017, the number of people without electricity access fell below 1 billion for the first time [20]. The share of people without access to electricity from Africa is 58%, while 19% of the share comes from developing Asia, and 31% from Latin America [20]. Numbers from Brazil show the aim to electrify 270 isolated areas and 2.7 million people by 2023 [13]. There exists a close relationship between the lack of energy and the low HDI (Human Development Index) of those localities [11]. It follows that increased access to energy allows economic growth and poverty

^{*} Supported by Newton Fund (ref. 261881580) and FAPEAM (Amazonas State Foundation for Research Support, calls 009/2017 and PROTI Pesquisa 2018).

alleviation [21]. To provide electricity for all, decentralized systems led by solar photovoltaic (PV) in off-grid and mini-grid systems will be the lowest-cost solution for three-quarters of the connections needed [19].

To evaluate or to obtain the optimal sizing of a PV system, there exist various specialized tools, e.g., RETScreen [25] and HOMER [28]; and even general-purpose tools, e.g., MATLAB/Simulink [16]. However, these tools are based on simulation; they have the drawback of incomplete coverage of design-space since verification of all possible combinations, and potential failures of a system are not feasible [10].

However, the industry demands the design solution to be the optimum, considering equipment manufacturers and models available on the market and not just minimum or maximum values of current or power for the optimized items. We need to evaluate the electrical compatibility among the equipment, which can only be achieved with specialized PV optimization software. Therefore, the optimal solution is the lowest cost from a list of equipment that meets the house’s electrical demands. Our analysis is based on Life Cycle Cost (LCC) [4], where the acquisition and replacement cost are considered over a specific period.

Optimization of PV systems is not a recent topic; since the 1990s, different techniques using different criteria to find ultimate combinations for design parameters, based on intuitive, numerical, and analytical methods, were proposed and developed [4]. Concerning the use of automated verification or synthesis to electrical systems, we can mention that in 2015, an automated simulation-based verification technique was applied to verify the correctness of the power system protection settings [27]. In 2017, Abate suggested the application of formal methods to verify and control the behavior of devices in a smart grid (e.g., [2]). In 2018, a verification methodology was applied to PV panels and its distributed power point tracking [12]. Lastly, in 2019, an automated verification methodology was proposed to validate the sizing of stand-alone solar PV systems [31]. However, *formal methods and its application to synthesize PV systems are not explored in literature, mainly because it requires background and experience in computer science and PV systems, which is not common.*

Here we developed a variant of the counterexample guided inductive synthesis (CEGIS) [3] technique for synthesizing optimal sizing of stand-alone PV systems. If we provide a correctness specification σ , our method uses that as a starting point and then iteratively produces a sequence of candidate solutions that satisfy σ related to power reliability. In particular, on each iteration, we synthesize the sizing of stand-alone PV systems, but that may not achieve the lowest cost. The candidate solution is then verified via software model checking with a lower bound that serves as the minimum cost of reference. If the verification step does not produce a counterexample, then the lower bound is adjusted; otherwise, we have achieved an optimal PV sizing.

Our work makes three significant contributions to advance the state-of-the-art in PV optimal sizing. First, the use of automated software verification in PV systems was uncommon in recent prior studies; we have shown that formal methods can detect various errors in PV systems designed by existing commercial tools [31]. The application of software verification to synthesize PV sizing is novel, thereby leading to more accurate results than existing commercial tools.

Second, we evaluate our approach using different state-of-the-art software verifiers to obtain the best performance in our verification backend for synthesizing optimal PV systems. We compare that to HOMER Pro optimization, with the results being validated with accurate commercial design software called PVsyst. Lastly, we discuss the challenges of applying software verification to PV systems and reflect on lessons learned from this experience.

2 Background

Fig. 1 illustrates how to obtain the optimal sizing of a stand-alone PV system using the manual or simulation techniques and the proposed synthesis technique. Note that the input information is the same for all the methods: weather data, price information, design requirements, as load curve and power demand, and design assumptions. However, when using automated synthesis, we also define the bound k to restrict the design-space search. Here we start with a low bound k and incrementally increase it to avoid time and memory constraints on our verification engine. Therefore, the proper choice of k is essential to this method.

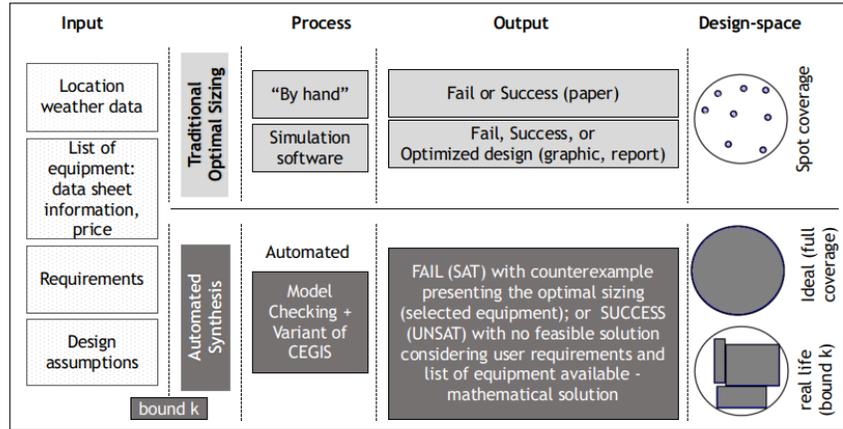


Fig. 1. Comparison of optimal sizing methods.

The techniques (manual, simulation, and automated synthesis) produce either a SUCCESS or FAIL result, thereby considering a feasible technical solution with the lowest cost. On the one hand, when done by simulation, we get a report or graphical result; on the other hand, the automated synthesis technique, which is mathematical reasoning about a model, presents a counterexample with the optimal solution stored in variables. Furthermore, the design-space coverage during the optimal sizing search is sound and complete when using synthesis.

2.1 Program Synthesis

The basic idea of program synthesis is to automatically construct a P program that satisfies a correctness specification σ . In particular, program synthesis is automatically performed by engines that use a correctness specification σ , as a starting point, and then incrementally produce a sequence of candidate solutions that partially satisfy σ [1]. As a result, a given candidate program p is iteratively refined to match σ more closely. Figure 2 illustrates the underlying architecture.

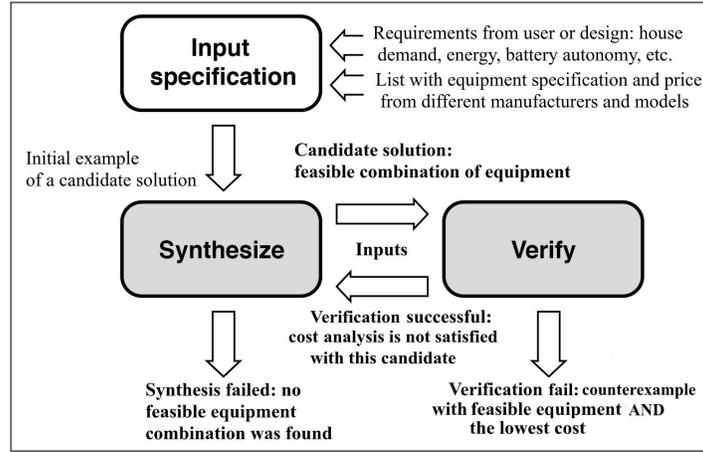


Fig. 2. CEGIS in PV system sizing.

The correctness specification σ provided to our synthesizer is of the form $\exists \mathbf{F}.\forall \mathbf{x}.\sigma(\mathbf{x}, \mathbf{F})$, where \mathbf{F} ranges over functions, \mathbf{x} ranges over ground terms, and σ is a quantifier-free (QF) formula typically supported by SMT solvers. The ground terms are interpreted over some finite domain \mathcal{D} , where \mathcal{D} can be encoded using the SMT's bit-vectors part. Our specification includes house demand, energy, and battery autonomy; we also provide equipment specifications and prices from different manufacturers and models.

In Figure 2, the phases **SYNTHESIZE** and **VERIFY** interact via a finite set of test vectors **INPUTS**, which is incrementally updated. Given the correctness specification σ , the **SYNTHESIZE** procedure tries to find an existential witness \mathbf{F} satisfying the specification $\sigma(\mathbf{x}, \mathbf{F})$, for all \mathbf{x} in **INPUTS** (as opposed to all $\mathbf{x} \in \mathcal{D}$). If **SYNTHESIZE** succeeds in finding a witness \mathbf{F} , the latter is a candidate solution to the full synthesis formula, which is passed to **VERIFY** to check whether it is a proper solution (*i.e.*, \mathbf{F} satisfies the specification $\sigma(\mathbf{x}, \mathbf{F})$ for all $\mathbf{x} \in \mathcal{D}$). If this is the case, then the algorithm terminates.

One may notice that each iteration of the traditional CEGIS loop adds a new input to the finite set **INPUTS**, which is then used for synthesis. Given that the full set of inputs \mathcal{D} is finite because we use bit-vector expressions, the refinement

loop can only iterate over a finite number of times. However, SYNTHESIZE may conclude that no candidate solution obeying σ for the finite set $INPUTS$ exists.

In our CEGIS variant, there exist four differences related to the traditional one: (1) there exists no test vector, and every candidate is generated in the SYNTHESIZE phase and sent to the VERIFY phase; (2) if the VERIFY phase is unsuccessful, then a new candidate is generated by SYNTHESIZE and (3) the lower bound of the VERIFY phase is incremented to search for the lowest cost; as a result, (4) there exists no refinement from the VERIFY phase back to the SYNTHESIZE phase. In particular, a new counterexample is not added to the INPUT set since a failure during the VERIFY phase will only discard a given candidate, which could be feasible in the next iteration with a new lower bound.

In summary, our proposal is a technique based on CEGIS, which aims to synthesize the optimal solution of a PV system; therefore, our technique addresses an optimization problem.

2.2 Sizing Stand-alone Solar PV Systems

A PV system is illustrated in Fig.3. It employs the PV generator (*panel or an array*), a semiconductor device that can convert solar energy into DC electricity. For night hours or rainy days, we hold *batteries*, where power can be stored and used. The use of batteries as a storage form implies the presence of a *charge controller* [17]. The PV arrays produce DC, and therefore when the PV system contains an AC load, a DC/AC conversion is required (*inverter*). The *AC load* dictates the behavior of the AC electrical load from the house that will be fed by the system.

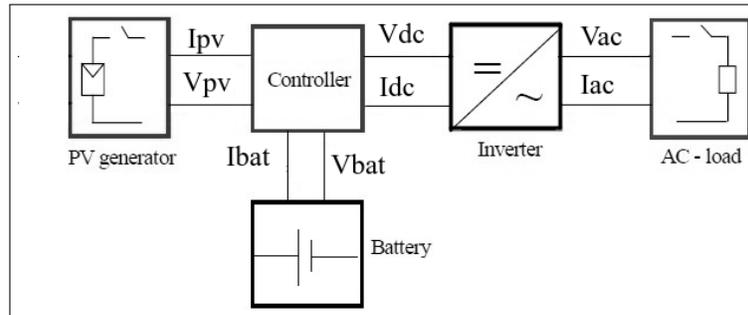


Fig. 3. Block diagram for a typical stand-alone PV system [17].

The sizing check stage can ensure that the system meets the standard project steps related to the critical period method (worst month) for solar energy system sizing [24]. It adopts an MPPT (Maximum Power Point Tracking) charge controller, which is the most common in use.

Since this paper's audience is targeted to be from the software verification area, we decided to use a higher-level explanation about the PV sizing. The sizing

process involves a set of eighteen equations related to the electrical engineering area, which is detailed online.³ Fig. 4 illustrates the overview of the steps that must be taken to size a stand-alone PV system.

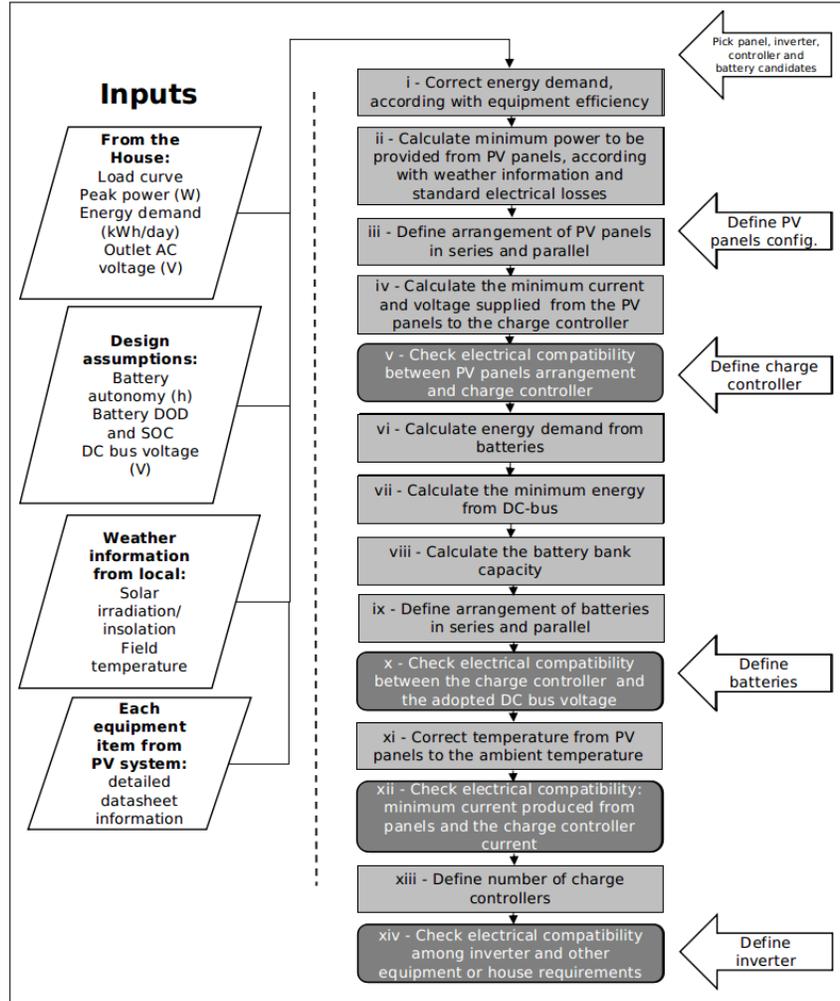


Fig. 4. High level description of stand-alone PV system sizing process.

On the left side of Fig. 4, we describe the needed **inputs** to size a PV system. There exist requirements from the house to be electrified; in particular, we have the design assumptions, weather information from the targeted local of the PV system deployment, and a possible initial list of equipment to be used, covering

³ <https://tinyurl.com/yck7dfxt>

all the items listed in Fig.3. Concerning the equipment list, the designer can use a few pieces of equipment or a vast one. We can also decide not to adopt the commercial equipment list. However, the result is a PV sizing of specific power, current, or voltage values, and usually just close to original equipment, which can be found in the market. This possibility has the drawback of possible incompatibility among equipment when the real one is bought and deployed.

There exist specific steps that aim the calculation of some variables and others related to the electrical compatibility validation among equipment items, both enumerated from **i** to **xiv**, as illustrated in Fig. 4; those represent different shades of gray of the rectangle boxes. The start point is usually a candidate list of PV panels, charge controller, battery, and inverter, as indicated at the top of the flowchart. The arrows, on the right side, show a point where some specific item is validated. The diagram does not show the returning location. However, if the candidate item is not compatible with others or does not meet some requirement, it must be changed to follow the indicated flow. The last rectangle box checks the inverter electrical compatibility with the DC-bus voltage, with the required AC voltage from the outlet. Besides, the inverter specified power must be lower than the charge controller power to avoid it from burning by overcharge. At the end of the flowchart, all the items are defined, and the PV sizing is finished.

These equations model the continuous-time behavior of the PV system; they produce real numbers except for the batteries and panels, where real numbers must be converted into integer ones, considering the minimum or maximum according to each equation. The verification and simulation tools need to handle non-linear real arithmetic to produce the correct result. Our mathematical model uses floating-point arithmetic. It is just an approximation of the real numbers. However, in this work, we are not concerned with calculating the rounding error, which is negligible when considering the size of the physical quantities and the variables adopted [15].

3 Synthesizing Optimal Sizing of Stand-alone Solar Photovoltaic Systems

The best compromise between two objectives makes the optimal sizing of PV systems: *power reliability* and *system cost* [4]. This study will rely on the critical period solar energy method [24], as described in Section 2.2. Our study will use an adapted Life Cycle Cost (LCC) analysis, where the acquisition cost of every item of equipment is considered, plus the installation cost, the operational and maintenance costs [4], given as,

$$LCC = C_{PV} + C_{bat} + C_{charger} + C_{inv} + C_{installation} + C_{batrep} + C_{PWO\&M} \quad (1)$$

where C_{PV} is the PV array cost, C_{bat} is the initial cost of batteries, $C_{charger}$ is the cost of the charger, C_{inv} is the inverter cost, $C_{installation}$ is the installation cost, C_{batrep} is battery replacement cost at current prices, and $C_{PWO\&M}$ is operation and maintenance costs at current rates. In this study, we will use a $C_{installation}$ equivalent to 5% of total equipment cost and a $C_{PWO\&M}$ equal

to US\$ 289.64/year according to Amazon State literature data [29]; and a LCC lifetime analysis of 20 years.

Algorithm 1 describes our pseudo-code to synthesize stand-alone PV systems. It uses software model checking as a backend verification engine [30]. The analytical method of optimization was adopted.

Algorithm 1 Synthesis algorithm

Input: weather data (temperature, solar irradiance); data from panels, controllers, batteries, and inverters; design requirements (load curve, peak demand, load surge power, energy consumption, battery autonomy, AC voltage); design assumptions (SOC, DOD, criteria and objectives for technical and cost analysis)

Output: FAIL (SAT) with counterexample showing the optimal sizing; SUCCESS (UNSAT), saying that the project has no feasible solution considering the requirements and the list of equipment

- 1: Initialize variables
- 2: Declare the maximum possible cost $MaxCost$
- 3: Calculate min possible Cost $MinCost$, based on the equipment list
- 4: **for** $HintCost = MinCost$ to $MaxCost$ **do**
- 5: Declare non-deterministic variables to select PV Panel, Controller, Battery, and Inverter from list
- 6: Calculate Steps i and ii of Fig. 4
- 7: Define PV panels arrangement: Step iii of Fig. 4
- 8: Calculate Step iv of Fig. 4
- 9: Enforce electrical compatibility in Step v of Fig. 4 with statement **assume**
- 10: Calculate Steps vi to viii of Fig. 4
- 11: Define battery arrangement according Step ix of Fig. 4
- 12: Enforce electrical compatibility in Step x of Fig. 4 with statement **assume**
- 13: Correct variables to ambient temperature: Step xi of Fig. 4
- 14: Enforce electrical compatibility in Step xii of Fig. 4 with statement **assume**
- 15: Define number of charge controllers: Step xiii of Fig. 4
- 16: Enforce electrical compatibilities in Step xiv of Fig. 4 with statement **assume** and define the inverter
- 17: Non-deterministic variables hold feasible equipment and cost
- 18: $F_{obj} \leftarrow N_{TP} * Panel_{Cost} + N_{TB} * Battery_{Cost} + Controller_{Cost} + Inverter_{Cost} + Installation_{Cost} + batrep_{Cost} + PWO \& M_{Cost}$
- 19: Violation check with **assert**($F_{obj} > HintCost$)
- 20: **end for**
- 21: **return** ()

Our synthesis algorithm will synthesize constant values; it starts with the input of the manufacturer's data and prices of PV panels, batteries, charge controllers, and inverters. Moreover, we define design (house) requirements and design assumptions. The *for*-loop started in line 4 controls the lowest cost of the PV solution. In particular, it begins with a value $MinCost$ and stops when the algorithm finds a feasible solution in which the value breaks the *assertion* stated in line 18. If that happens, then our algorithm has found an optimal solution, thereby indicating that the VERIFY phase reached a satisfiable condition (*SAT*).

The *MaxCost* value in line 2 is just a very high value inserted as a limit to the *for*-loop, that meets one of the following requirements. (1) It will never be reached because the optimal solution will be found first (SAT result); or (2) it will be achieved when the search engine did not find a feasible solution for the optimization (UNSAT result).

Our synthesis algorithm uses non-deterministic variables to choose one specific constant from a given list of PV panels, controllers, batteries, and inverters (line 5). This procedure ensures that our synthesis engine checks all combinations of items from each equipment, and combines them to assemble a viable (candidate) PV solution, which meets user requirements. A list of forty equipment from ten different manufacturers was provided (as INPUT) to our synthesis engine to allow the choice of every item of PV sizing. Datasheet from each item was necessary to collect technical information. Moreover, the price of each item was obtained from available quotations in the market, and if the currency was not in US dollars, then it was used the exchange rate of the day to convert it to US dollars. All this data is available online.⁴

Next, we use a set of equations to calculate the sizing variables (lines 6, 7, 8, 10, 11, 13, 15). The statements *assume* (lines 9, 12, 14, and 16) ensures compatibility of the items chosen from the list of equipment: the VERIFY phase uses only items (among all the possible ones) that satisfy the statements of those lines. Line 12 is for the battery bank. Lines 9 and 14 are the charge controller voltage check. Line 16 does the inverter check and ensures the power demand and the surge power of the inverter. Therefore, our synthesis algorithm reaches line 17 with one feasible solution, and the cost of that solution is calculated in F_{obj} (line 18). This cost is the equivalent to Eq. (1).

If our algorithm does not find a feasible solution among the item of equipment that was provided for our SYNTHESIZE phase, then the result is unsatisfiable (UNSAT). In particular, the program finishes without finding a solution, indicating that it was unable to combine the specific items of equipment to create a feasible solution. The main challenge for the SYNTHESIZE phase is to find a feasible candidate solution for the constraints and user requirements. The problem for the VERIFY phase is to find the lowest acquisition cost from a list of equipment and components provided by the SYNTHESIZE phase.

Summarizing: We use four non-deterministic variables to index four matrices with complete datasheet information from an equipment item. We have four variables and four matrices: one to PV panels, one to batteries, one to the inverter, and one to the charge controller. Those non-deterministic variables are used during the search for the feasible solution and controlled by the statements **assume**. The candidate solution is proposed in the phase SYNTHESIZE of the technique. For the VERIFY phase, we performed a linear search for the lowest cost, and the result is based on FAIL or SUCCESS check from a cost that is linearly incremented in this model. The model checking is used during this VERIFY phase. If the process produces FAIL, then an optimal solution is found. If SUCCESS is presented, i.e. the property holds, there is no feasible solution from the equipment list provided by the technique. Note that the process described

⁴ <https://tinyurl.com/ycgbsgkp>

here is completely automated and that validation is performed by our VERIFY phase to ensure that the solution is sound. The verification engines transform the Algorithm 1 into the Boolean expressions that are passed to the solver to verify $(C \wedge \neg P)$, as described online.⁵

4 Results and Discussion

4.1 Description of the Case Studies

The proposed synthesis approach was evaluated in seven stand-alone PV system case studies. These case studies were defined based on the usual electrical load found in riverside communities in the Amazonas State, Brazil [31,29], except for case 7, which was idealized to support a few lamps and a 12k BTUs air-conditioner solution. Here we report each case study as a 4-tuple $\{power\ peak\ (W); power\ surge\ (W); energy\ consumption\ (Wh/day); battery\ autonomy\ (hours)\}$ as follows: **1:** {342; 342; 3,900; 48}; **2:** {814; 980; 4,880; 48}; **3:** {815; 980; 4,880; 12}; **4:** {253; 722; 3,600; 48}; **5:** {263; 732; 2,500; 48}; **6:** {322; 896; 4,300; 48}; **7:** {1,586; 2,900; 14,000; 48}. This 4-tuple represents the Algorithm 1 inputs. For all cases, an estimated load curve (kWh) was defined based on the electronics consumers in each house. Our synthesis algorithm was fed with data and costs of forty equipment items from ten different manufacturers of PV systems. Three state-of-the-art verifiers, CBMC⁶ v5.11 with MiniSat 2.2.1 [23], ESBMC⁷ v6.0.0 [14] with the Boolector 3.0.1 solver [8], and CPAchecker⁸ v1.8 [6] with MathSAT 5.5.3 [9], were used as verification engines to compare the proposed approach effectiveness and efficiency.

4.2 Optimization/Simulation Tools and Assumptions

Concerning the off-the-shelf optimization/simulation tools, only HOMER Pro performs an off-grid system with battery backup analysis and includes economical analysis. Here we used HOMER Pro version 3.13.1 as a state-of-the-art optimization tool for comparison purposes. In particular, HOMER Pro has the following characteristics: (a) it is available only for MS-Windows, its annual standard subscription costs US\$ 504.00 [18]; (b) it has two optimization algorithms: one algorithm simulates all of the feasible system configurations defined by the search space, and additionally, a proprietary derivative-free algorithm to search for the least-costly system; (c) it does not have LCC cost in its reports, only Net Present Cost (NPC); however, we can obtain LCC from NPC; (d) the optimization analysis defines a load curve and temperature according to data collected from online databases. However, to allow a correct comparison, the curve load and the temperature were defined the same as our synthesis approach; (e) it

⁵ <https://tinyurl.com/yajfmav1>

⁶ Command-line: `$ cbmc --unwind 100 file.c --trace`

⁷ Command-line: `$ esbmc filename.c --incremental-bmc --boolector`

⁸ Command-line: `$ scripts/cpa.sh -heap 64000m -config config/bmc-incremental.properties -spec config/specification/sv-comp-reachability.spc file.c`

does not have a charge controller. During the tests, we have chosen the “load-following” option, which produces enough power to meet the demand [18] and (usually) presents a non-overestimated solution; (f) it was assumed 95% availability of the PV system. By definition, “availability” is the percentage of time at which a power system can feed the load requirements [22]. For an ordinary house electrical load, 95% is considered acceptable; (g) it was assumed a string of two batteries to match the voltage of the 24 V DC system, which was used for our automated synthesis tool; (h) it was included a generic flat-plate PV of 1 kW and generic lead-acid batteries of 1 kW (83.4 Ah capacity). During run-time, HOMER decides the size in kW of each one based on feasibility and lower cost.

To validate and compare the optimal sizing solution produced by our approach and by HOMER Pro, we use a simulation tool, called PVsyst version 6.86 [26], with plenty of commercial equipment in its data bank. We have considered a comparison for an entire year’s weather data of simulation to guarantee that the proposed sizing meets the electrification requirements. PVsyst is a PC software package developed by a Swiss company used for the study, sizing, simulation and data analysis of solar PV systems. PVsyst contains design, sizing, 3D shading scene, simulation, grid, and off-grid features. It uses comprehensive irradiation data from Meteonorm,⁹ and aging analysis [5]. However, it does not perform optimization; therefore, PVsyst needs the system sized to validate it. Furthermore, PVsyst does not have commercial inverter equipment and, as a result, does not consider surge power demand as the ones produced by air conditioners and refrigerators for a few seconds. PVsyst is commercial software with a 30-day test possibility and runs only in MS-Windows.

4.3 Objectives and Setup

Our evaluation aims to answer three experimental goals: [EG1] (**soundness**) Does our automated synthesis approach provide correct results?; [EG2] (**performance**) How do the software verifiers compare to each other for synthesizing PV systems?; and [EG3] (**state-of-the-art**) how does our formal synthesis tool compare to a specialized simulation tool?

All experiments regarding the verification tools were conducted on an otherwise idle Intel Xeon CPU E5-4617 (8-cores) with 2.90 GHz and 64 GB RAM, running Ubuntu 16.04 LTS 64-bits. For HOMER Pro, we have used an Intel Core i5-4210 (4-cores) with 1.7 GHz and 4 GB RAM running Windows 10. The ideal scenario would be to use the same hardware configuration for the experiments. However, we faced restrictions concerning the license for the HOMER Pro tool; besides, we did not have the autonomy to change the Linux VM machine installed in our university’s servers due to the internal policy. This setup has an impact on performance, which is less favorable to HOMER Pro. PVsyst used the same configuration as HOMER Pro. We perform the experiments with a predefined *time out* of 660 minutes.

⁹ <https://meteonorm.com/en/>

4.4 Results

The results are presented in Table 1. The violation (SAT result) indicated in

Table 1. Case studies and results: optimization of stand-alone PV systems.

Tools	CBMC 5.11 (MiniSat 2.2.1)	ESBMC 6.0.0 (Boolector 3.0.1)	CPAchecker 1.8 (MathSAT 5.5.3)	HOMER Pro 3.13.1
Specification	Result	Result	Result	Result
Case Study 1 Peak:342W Surge:342W E:3,900Wh/day Autonomy:48h	OM	SAT (620 min) NTP:6×330W (2P-3S) NBT:16×105Ah (2S-8P) Controller 35A/145V Inverter 700W/48V LCC: US\$ 10,214.04	SAT (548 min) NTP:6×330W (2P-3S) NBT:16×105Ah (2S-8P) Controller 35A/145V Inverter 700W/1600W/48V LCC: US\$ 10,214.04	(Time: 0.33 min) 2.53 kW of PV NBT:12×83.4Ah (2S-6P) 0.351kW inverter LCC: US\$ 7,808.04
Case Study 2 Peak:814W Surge:980W E:4,880Wh/day Autonomy:48h	OM	TO	TO	(Time: 0.18 min) 3.71 kW of PV NBT:20×83.4Ah (2S-10P) 0.817kW inverter LCC: US\$ 12,861.75
Case Study 3 Peak:815W Surge:980W E:4,880Wh/day Autonomy:12h	OM	SAT (63 min) NTP:14×150W (7P-2S) NBT:6×105Ah (2S-3P) Controller 35A/145V Inverter 1,200W/48V LCC: US\$ 9,274.07	TO	Not possible
Case Study 4 Peak:253W Surge:722W E:3,600Wh/day Autonomy:48h	OM	SAT (147 min) NTP:6×330W (2P-3S) NBT:16×105Ah (2S-8P) Controller 35A/145V Inverter 280W/48V LCC: US\$ 9,678.63	SAT (605 min) NTP:6×330W (2P-3S) NBT:16×105Ah (2S-8P) Controller 35A/145V Inverter 280W/48V LCC: US\$ 9,678.63	(Time: 0.23 min) 2.42 kW of PV NBT:12×83.4Ah (2S-6P) 0.254kW inverter LCC: US\$ 7,677.95
Case Study 5 Peak:263W Surge:732W E:2,500Wh/day Autonomy:48h	OM	SAT (36.70 min) NTP:4×330W (2S-2P) NBT:14×80Ah (2S-7P) Controller 35A/145V Inverter 280W/24V LCC: US\$ 8,900.70	SAT (254.25 min) NTP:4×330W (2S-2P) NBT:14×80Ah (2S-7P) Controller 35A/145V Inverter 280W/24V LCC: US\$ 8,900.70	(Time: 0.18 min) 1.59 kW of PV NBT:10×83.4Ah (2S-5P) 0.268kW inverter LCC: US\$ 6,175.57
Case Study 6 Peak:322W Surge:896W E:4,300Wh/day Autonomy:48h	OM	SAT (380.93 min) NTP:6×320W (2P-3S) NBT:18×105Ah (2S-9P) Controller 35A/145V Inverter 400W/24V LCC: US\$ 10,136.61	TO	(Time: 0.22 min) 3.15 kW of PV NBT:14×83.4Ah (2S-7P) 0.328kW inverter LCC: US\$ 9,112.45
Case Study 7 Peak:1,586W Surge:2,900W E:14,000Wh/day Autonomy:48h	OM	UNSAT (0.48 min)	TO	(Time: 0.20 min) 12.5 kW of PV NBT:66×83.4Ah (2S-33P) 1.60kW inverter LCC: US\$ 41,878.11

Legend: OM = out of memory; TO = time out; IF = internal failure; E = energy; NTP = total number of panels, NBtotal = total number of batteries, NPS = number of panels in series; NPP = number of panels in parallel, NBS = number of batteries in series; NBP = number of batteries in parallel; LCC = Life Cycle Cost.

Table 1 is the *assert* of the line 18 in Algorithm 1 and suggests that an optimal solution was found. Here, CBMC was unable to produce any conclusive result; *out of memory* situations occurred in all case studies.

CPAchecker was able to synthesize optimal sizing in three out of the seven case studies (cases 1, 4, and 5): the result was produced within the time limit, which varied from 4 to 9 hours. Fig. 5 illustrates the result of case 5 with the

optimal sizing appearing on the left side as the integer 3 for the solar panel (which is the Canadian CS6U-330P model of 330 W from the manufacturer Victron Energy), battery 0 refers to the model 12MF80 of 80 Ah from Moura, charge controller 0 refers to the model 35A-145V MPPT from Victron Energy. The inverter number 2 refers to the Epever model IP350-11 of 280 W (nominal power) and 750 W of surge power. The variables NTP, NPS, NPP, NBS, NBP, and NBTotal, also presented in the counterexample, shows the number of panels and batteries and how they are connected. Case studies 2, 3, 6, and 7 led to a *time out* result in CPAchecker, i.e., it was not solved within 11 hours.

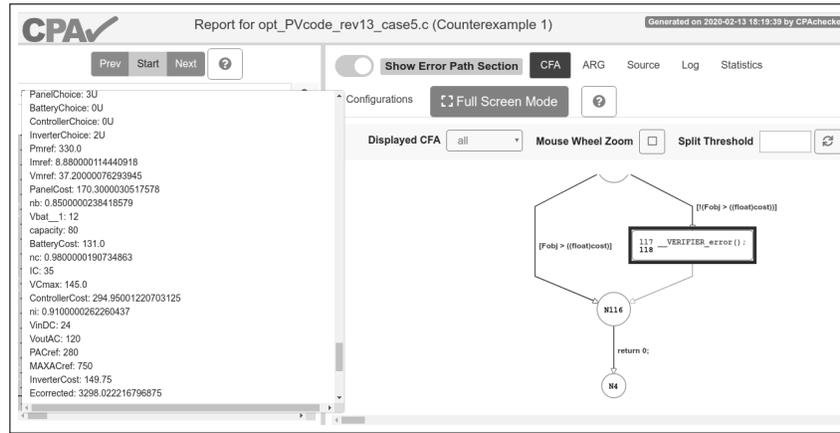


Fig. 5. Counterexample generated by CPAchecker after validation of case 5.

Here we used ESBMC in BMC incremental mode with Boolector; ESBMC was able to reach the optimal sizing of case studies 1, 3, 4, 5, and 6 with a FAIL/ SAT response, varying from 36 minutes to 10 hours. ESBMC, with this configuration, was unable to obtain an optimal solution in cases 2 and 7. Case 2 produced a *time out*. Moreover, case 7 resulted in a UNSAT result, i.e., ESBMC was unable to provide a feasible solution. However, this is not a bug, and it means that the available list of equipment can not produce a feasible solution that satisfies electrical compatibility or design requirements. This UNSAT situation was reached in less than one minute. These experimental results answer the *EG2*.

HOMER Pro was able to evaluate six case studies (cases 1, 2, 4, 5, 6, and 7) under 30 seconds; it was much faster than the proposed automated synthesis tool (cf. *EG3*). Case study 3 could not be simulated since HOMER Pro does not have the battery autonomy adjustment feature, i.e., the tool always tries to feed the given load with electricity 365 days/year. Some HOMER Pro drawbacks were also noted. (1) System equipment does not include an explicit charge controller. HOMER Pro includes a controller automatically to simulate the charge/discharge of batteries and to meet the load requirement. However, without costs or even with electrical characteristics such as maximum current and voltage, which are common during PV sizing. (2) HOMER Pro requires the

inclusion of some battery specification to initiate optimization; however, it does not change the electrical specifications during simulation; the results presented are multiples of the original battery type suggested by the user. For example, it was started with an 83.4 Ah lead-acid battery, and during simulation, HOMER Pro did not try to use other capacities or types. (3) HOMER Pro does not present the optimal solution in terms of connections of PV panel arrays, just the total in terms of power, i.e., it presents neither the models and the power of each PV panel nor the total of panels in series or parallel. The cost of every equipment item used in HOMER Pro is a US-based cost, without adaptation regardless of where the equipment is installed.

We have real PV systems deployed since June 2018 in a riverside community in the State of Amazonas, Brazil, GPS coordinates $2^{\circ}44'50.0''\text{S } 60^{\circ}25'47.8''\text{W}$, with demands of case studies 1, 4, 5, and 6, always with a $3 \times 325 \text{ W}$ (3S, total 975 W) panels and $4 \times 220 \text{ Ah}$ (2S-2P = 440 Ah) lead-acid batteries.

4.5 Comparison Between Formal Synthesis and HOMER Pro

If we compare the formal synthesis results against those of HOMER Pro, we observed some distinct effects in terms of the technical solution and cost (cf. Table 1). Concerning the performance, there exists a vast difference in favor of HOMER Pro that obtained the results in considerably less time: few seconds in the opposite of an average of 4 hours for the automated synthesis technique. Particularly in the case of LCC, the cost varied from 11% to 44%, producing a higher estimation from the automated synthesis technique. However, considering that the cost of individual items of each database used to compose the optimal design is not the same among the tools, it is plausible to obtain distinct results.

On the one hand, concerning the PV panels sizing, the results presented by the automated synthesis were smaller in terms of power than the ones produced by the simulation tool. The difference varied from 19% to 65%. On the other hand, concerning the battery bank, the results were smaller in terms of HOMER Pro capacity. The difference was between 34% to 68%. The mathematical models are different and particular parameters can be tuned for each technique, and that can justify the difference, which was presented in all the case studies.

Those discrepancies are not easy to address without some real systems validation. However, we use the simulation software PVsyst to validate the optimal sizing produced, as shown in Table 2. Note that PVsyst has a pre-sizing feature, which presents a minimum recommended sizing of PV panels and batteries (only) without using manufacturers' data or models for it. This feature was used as reference mainly with HOMER Pro, where there exists no equipment brands or models (only power and capacities specification). PVsyst was used with the field-deployed and the formal synthesis sizing solutions, where brands and models were simulated in PVsyst according to the sized system.

Each simulation with PVsyst took 4 seconds. We were unable to validate the case study 3 using PVsyst. The battery autonomy is less than 24 hours, and only the proposed synthesis technique can perform the optimal sizing (PVsyst and HOMER Pro are limited for a 24 h minimum). Case studies 2 and 7 had only HOMER Pro sizing validation. There is no deployed equivalent system in

Table 2. Optimal sizing validation with PVsyst.

CS	PVsyst (pre-sizing)	Field deployed validation	Formal synthesis sizing validation	HOMER Pro sizing validation
CS 1	P= 1,166 W B= 381 Ah (minimum)	Not correct sizing Avail. < 95% (91.06%)	No error found 100% of avail.	No error found Panels oversized in $2.16 \times$ Batteries oversized in $1.39 \times$
CS 2	P= 1,482 W B= 478 Ah (minimum)	NA There exists no real PV system available for comparison	NA (TO result in Table 1)	No error found Panels oversized in $2.6 \times$ Batteries oversized in $1.74 \times$
CS 3	Not possible to simulate (autonomy < 24h)	NA There exists no real PV system available for comparison	Only technique that produced solution	NA (autonomy < 24h)
CS 4	P= 1,078 W B= 354 Ah (minimum)	No error found 95.76% of avail.	No error found 97.37% of avail.	No error found Panels oversized in $2.24 \times$ Batteries oversized in $1.41 \times$
CS 5	P= 823 W B= 268 Ah (minimum)	No error found 100% of avail.	No error found 100% of avail.	No error found Panels oversized in $1.93 \times$ Batteries oversized in $1.56 \times$
CS 6	P= 1,299 W B= 421 Ah (minimum)	Not correct sizing Avail. < 95% (85.65%)	No error found 100% of avail.	No error found Panels oversized in $2.42 \times$ Batteries oversized in $1.38 \times$
CS 7	P= 4,263 W B= 1,384 Ah (minimum)	NA There exists no real PV system available for comparison	NA (UNSAT result in Table 1)	No error found Panels oversized in $2.9 \times$ Batteries oversized in $1.99 \times$

Legend: CS = case study; NA = sizing not available for validation; B = batteries capacity; P = panels power; Avail.= Availability (expected of 95% or greater as a design requirement).

the field, and the synthesis technique did not present a solution due to *time out* and internal failures in the underlying verification engine.

Overall, those comparisons with our approach, the optimization software, and the deployed systems, with validation through simulation tool, show that the synthesis solution is sound and complete, which answers *EG1* and *EG3*.

Concerning the cost (LCC) present by both tools, HOMER Pro does not use the real cost for PV systems deployed in Brazil; therefore, the optimal solution presented by HOMER Pro is notoriously cheaper than our technique. However, considering that the aim is to present an optimal PV sizing solution that is feasible and closer to the market prices, our technique is more indicated.

Besides that, HOMER Pro suggests a value in kW for the inverters that are very close to the maximum load of every case study, but it is not commercial. The proposed synthesis tool, however, presents inverters that are commercial and can be obtained off-the-shelf. Moreover, our synthesis approach considers surge power demand from the house, which is not viewed by HOMER Pro or PVsyst. This feature is a definite advantage of the formal synthesis method. HOMER Pro does not include charge controllers as a specific item of equipment in its mathematical model; only the synthesis tool presents a commercial controller and includes it during the cost analysis. The formal synthesis method, therefore, presents more reliable results than HOMER Pro.

In summary, our synthesis technique can present a solution that is far more detailed and closer to commercial conditions than the answer given by HOMER Pro. In particular, the automated synthesis method can provide all the details of every component of a PV system solution, with complete electrical information from the manufacturer datasheet, including the model of the component, nominal

current, and voltage. In this respect, even the name of the manufacturer can be cited (in Table 1, it was removed to avoid unauthorized advertising). Moreover, the validation through PVsyst simulation, using the PV sizing produced by HOMER Pro and our synthesis approach, shows that our results are feasible and not as oversized as HOMER Pro results, mainly concerning PV panels.

An optimal solution from a tool is not necessarily the same optimal from other tools, mainly when the database of equipment items (with different costs) is not the same [4]. Therefore, the comparison must take this issue into account.

4.6 Threats to validity

We have reported a favorable assessment of the proposed method. Nevertheless, we have also identified four threats to the validity of our results that constitute future work. First, improvement of the power reliability analysis: to include loss of load probability or loss of power supply probability, which can make the study more accurate. Second, improvement of the system cost analysis by adding operational and maintenance costs to the adopted LCC analysis. Third, increase the equipment and manufacturers database: this will increase the optimization complexity, but the result will also allow improved sizing. Lastly, the underlying software verifiers employed here perform bit-precise verification based on the Floating-Point (FP) theory. We could use a real arithmetic strategy to tackle these equations; however, in this study, we have exploited the FP arithmetic, approximating the real one.

5 Conclusions

Our novelty relies on a practical approach to pursue the optimal PV systems' optimal solution using contemporary formal methods. The use of formal synthesis to design PV systems has no precedent in literature; we show that our approach results, using seven case studies, are promising. Our synthesis tool can present a solution that is far more detailed and closer to commercial reality than the solution given by the commercial tool. The battery autonomy feature, together with the details of every component of a PV system solution, is advantageous for our synthesis approach; these details are essential for the PV system owner. The industry demands proximity between the result presented by optimization tools and the items of equipment for solar systems available on the market.

Our synthesis technique was developed and used open-source software verifiers and the environment, in contrast to the optimization and simulation tools used in this work. We have also observed that state-of-the-art software verifiers are doing an excellent job of solving hard verification conditions based on the underlying SAT/SMT solvers. Lastly, the use of data from real deployed systems in Brazil and the validation through PVsyst was essential to validate the comparison. We have shown that our technique has a promising result for PV system sizing optimization. Our focus for future work consists of improving the search mechanism used in the VERIFY phase of our synthesis technique to speed up the overall time. We plan to use parallel binary search [32] or even a solver that is specific to perform optimization with model checking as νZ [7].

References

1. Abate, A., Bessa, I., Cattaruzza, D., Cordeiro, L., David, C., Kesseli, P., Kroening, D., Polgreen, E.: Automated formal synthesis of digital controllers for state-space physical plants. In: *Comp. Aided Verif. (CAV)*. vol. LNCS 10426, pp. 462–482 (2017)
2. Abate, A.: Verification of networks of smart energy systems over the cloud. In: Bogomolov, S., Martel, M., Prabhakar, P. (eds.) *Num. Soft. Verif.* vol. LNCS 10152, pp. 1–14 (2017)
3. Abate, A., David, C., Kesseli, P., Kroening, D., Polgreen, E.: Counterexample guided inductive synthesis modulo theories. In: Chockler, H., Weissenbacher, G. (eds.) *Computer Aided Verification*. pp. 270–288. Springer International Publishing, Cham (2018)
4. Alsadi, S.; Khatib, T.: Photovoltaic power systems optimization research status: A review of criteria, constrains, models, techniques, and software tools. *Appl. Sci.* **8**(1761), 1–30 (2018)
5. Barua, S., Prasath, R.A., Boruah, D.: Rooftop solar photovoltaic system design and assessment for the academic campus using PVsyst software. *International Journal of Electronics and Electrical Engineering* **5**(1), 76–83 (2017)
6. Beyer, D., Keremoglu, M.E.: CPAchecker: A tool for configurable software verification. In: Gopalakrishnan, G., Qadeer, S. (eds.) *Lecture Notes in Computer Science*. vol. LNCS 6806, pp. 184–190. Springer, Berlin, Heidelberg (2011)
7. Bjørner, N., Phan, A., Fleckenstein, L.: *vz* - an optimizing SMT solver. In: Baier, C., Tinelli, C. (eds.) *21st International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*. LNCS, vol. 9035, pp. 194–199. Springer (2015)
8. Brummayer, R., Biere, A.: Boolector: An efficient SMT solver for bit-vectors and arrays. In: *Tools and Alg. for the Const. and An. of Sys. (TACAS)*. vol. LNCS 5505, pp. 174–177 (2009)
9. Cimatti, A., Griggio, A., Schaafsma, B., Sebastiani, R.: The MathSAT5 SMT Solver. In: Piterman, N., Smolka, S. (eds.) *Proceedings of TACAS*. LNCS, vol. 7795, pp. 93–107. Springer (2013)
10. Clarke, E.M., Henzinger, T.A., Veith, H.: Introduction to model checking. In: *Handbook of Model Checking*, pp. 1–26. Springer (2018)
11. Coelho, S., Sanches-Pereira, A., Tudeschini, L., Escobar, J., Poveda, M., Coluna, N., Collin, A., Rovere, E.L., Trindade, A., Pereira, O.: Biomass residues as electricity generation source in low HD source in regions of Brazil. In: UNESP (ed.) *The XI Latin. Cong. of Elec. Gener. and Transm. CLAGTEE*. pp. 1–8 (2015)
12. Driouch, Y., Parente, M., Tronci, E.: A methodology for a complete simulation of cyber-physical energy systems. In: *IEEE Work. on Envir., Energ., and Struc. Monit. Syst. (EESMS)*. pp. 1–5 (2018)
13. Empresa de Pesquisa Energética EPE: Sistemas Isolados - Planejamento Ciclo 2018 - 2023. <http://www.epe.gov.br/sites-pt/publicacoes-dados-abertos/publicacoes> (2018), [Accessed 04 April 2019]
14. Gadelha, M., Monteiro, F., Morse, J., Cordeiro, L., Fischer, B., Nicole, D.: ESBMC 5.0: An industrial-strength C model checker. In: *33rd ACM/IEEE Int. Conf. on Aut. Soft. Engin. (ASE'18)*. pp. 888–891. ACM, New York, NY, USA (2018)
15. Gadelha, M.Y.R., Cordeiro, L.C., Nicole, D.A.: An efficient floating-point bit-blasting API for verifying C programs. *CoRR* **abs/2004.12699** (2020), <https://arxiv.org/abs/2004.12699>

16. Gow, J., Manning, C.: Development of a photovoltaic array model for use in power-electronics simulation studies. In: Proceedings of the 14th IEE Electric Power Applications Conference. vol. 146(2), pp. 193–200 (1999)
17. Hansen, A., Sørensen, P., Hansen, L., Bindner, H.: Models for a stand-alone PV system. No. 1219 in Denmark. Forskningscenter Risoe. Risoe-r, Forskningscenter Risoe (2001)
18. HOMER: The HOMER microgrid software. <http://www.homerenergy.com/software.html> (2017), [Accessed 1st June 2019]
19. Hussein, M., LEAL FILHO, W.: Analysis of energy as a precondition for improvement of living conditions and poverty reduction in sub-Saharan Africa. In: Scientific Research and Essays. vol. 7(30), pp. 2656–2666 (2012)
20. IEA: World Energy Outlook 2018. IEA, Paris (2018)
21. Karekesi, S., Lata, K., Coelho, S.: Renewable Energy - A Global Review of Technologies, Policies and Markets, chap. Traditional Biomass Energy: Improving Its Use and Moving to Modern Energy Use, pp. 231–261. London: Earthscan (2006)
22. Khatib, T., Elmenreich, W.: Optimum availability of standalone photovoltaic power systems for remote housing electrification. *Int. Journal of Photoenergy* **2014**(Article ID 475080), 5 pages (2014)
23. Kroening, D., Tautschnig, M.: CBMC C Bounded Model Checker (competition contribution). In: Tools and Alg. for the Const. and An. of Sys. (TACAS). vol. LNCS 8413, pp. 389–391 (2014)
24. Pinho, J., Galdino, M.: Manual de Engenharia para Sistemas Fotovoltaicos. CEPTEL CRESESB, Rio de Janeiro/RJ (2014)
25. Pradhan, S., Singh, S., Choudhury, M., Dwivedy, D.: Study of cost analysis and emission analysis for grid connected PV systems using RETSCREEN 4 simulation software. *Int. J. of Eng. Res. & Tech.* **4**(4), 203–207 (2015)
26. PVsyst: Logiciel Photovoltaïque. <https://www.pvsyst.com/> (2020), [Accessed 24 April 2020]
27. Sengupta, A., Mukhopadhyay, S., Sinha, A.: Automated verification of power system protection schemes Part I: Modeling and specifications. In: *IEEE Tran. on Power Del.* vol. 30(5), pp. 2077–2086 (2015)
28. Swarnkar, N., Gidwani, L., Sharma, R.: An application of HOMER Pro in optimization of hybrid energy system for electrification of technical institute. In: *Int. Conf. on Energ. Eff. Tech. for Sust. (ICEETS)*. pp. 56–61 (2016)
29. Trindade, A.: Ferramenta de análise comparativa de projetos de eletrificação rural com fontes renováveis de energia na Amazônia. In: *IX Congresso sobre Geração Distribuída e Energia no Meio Rural - AGRENER GD*. p. n.pag. (2013)
30. Trindade, A., Cordeiro, L.C.: Optimal sizing of stand-alone solar PV systems via automated formal synthesis. *CoRR* **abs/1909.13139** (2019), <http://arxiv.org/abs/1909.13139>
31. Trindade, A.B., Cordeiro, L.C.: Automated formal verification of stand-alone solar photovoltaic systems. *Solar Energy* **193**(1), 684–691 (2019)
32. Trindade, A.B., Degelo, R.D.F., Junior, E.G.D.S., Ismail, H.I., Silva, H.C.D., Cordeiro, L.C.: Multi-core model checking and maximum satisfiability applied to hardware-software partitioning. *IJES* **9**(6), 570–582 (2017)