

The sole responsibility for the content published on this document lies with the authors. It does not necessarily reflect the opinion of the Innovation and Networks Executive Agency (INEA) or the European Commission (EC). INEA or the EC are not responsible for any use that may be made of the information contained therein.

T4.1

Tool for Ancillary Services Procurement in Day- Ahead Operation Planning of Distribution Networks

D4.2



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 864298.

DOCUMENT CONTROL PAGE

DOCUMENT	D4.2 Specification of the tractable tool
TYPE	Deliverable
DISTRIBUTION LEVEL	Public
DUE DELIVERY DATE	31 / 08 / 2022
DATE OF DELIVERY	
DELIVERABLE RESPONSIBLE	LIST
AUTHOR(S)	Muhammad Usman, Florin Capitanescu
OFFICIAL REVIEWER(S)	

DOCUMENT HISTORY

VERSION	AUTHORS	DATE	CHANGES
1.0	Muhammad Usman, Florin Capitanescu	03/08/2022	Template and first version of D4.2 including description of T4.1
1.1	Muhammad Usman, Florin Capitanescu	26/08/2022	Minor changes on the document

Table of Contents

1. SEQUENTIAL LINEARIZATION ALGORITHM (SLA)-BASED TOOL FOR ANCILLARY SERVICES PROCUREMENT IN DAY-AHEAD OPERATIONAL PLANNING OF THE DISTRIBUTION NETWORK (TASK 4.1)	7
1.1. Functional description of SLA-based stochastic multi-period optimal power flow (S-MP-OPF) tool	7
1.2. Technical description of SLA-based S-MP-OPF tool	8
1.2.1. Stochastic Multi-Period Flexibility Procurement Framework based on novel MILP Model (M1)	8
1.2.2. Sequential Linearization Algorithm	10
1.2.2.1. Role of inner- and outer-loops	12
1.2.3. Trustable Approximation Criterion	12
1.3. Input and output requirements	13
1.3.1. Input data	13
1.3.2. Output data	13
1.4. Implementation and run of SLA-based S-MP-OPF tool in Julia (User Guide)	13
2. APPENDICES	17
2.1. Appendix A.1	17
2.2. Appendix A.2 (Installation of Julia and relevant Packages)	18
2.3. Appendix A.3 (Scenario Generation Tool)	19
2.4. Appendix A.4	20
3. BIBLIOGRAPHY	20

List of Figures

Fig. 1 High level functional description of tool 4.1	8
Fig. 2: Flow Chart of the proposed sequential Linearization Algorithm	11
Fig. 3: Tractable Tool main file showing packages, user-defined FUNCTIONS, and input files	14
Fig. 4: Available Flexible Options	15
Fig. 5: Run button of Juno IDE	16
Fig. 6: Format of Bus Data	17
Fig. 7: Format of branch data	18
Fig. 8: Format of Flexible Load Data	18
Fig. 9: Format of Electrical Energy Storage Data	18
Fig. 10: Wind power scenarios	20
Fig. 11: Solar power scenarios	20

List of Tables

Table 1: Coefficients of variables in linearized active and reactive power flows, and longitudinal branch current limit expression.....	10
---	----

Abbreviations and Acronyms

<i>ADS</i>	Active Distribution System
<i>DER</i>	Distributed Energy Resources
<i>DSO</i>	Distribution System Operator
<i>EES</i>	Electical Energy Storage
<i>FL</i>	Flexible Loads
<i>OPF</i>	Optimal Power Flow
<i>PoL</i>	Point of Linearization
<i>PV</i>	Photo-Voltaic
<i>S-MP-MILP</i>	Stochastic Multi-Period Mixed Integer Linear Programming
<i>RES</i>	Renewable Energy Sources
<i>TSO</i>	Transmission System Operator

Executive Summary

This deliverable titled “Tools for ancillary services procurement in day-ahead operation planning of distribution networks” corresponds to a sequential linearization algorithm-based tool, which is the outcome of task T4.1 of WP4 of ATTEST project, for solving stochastic multi-period optimal power flow in active distribution systems. The overarching objective of T4.1 is to develop a novel tractable tool for distribution system operators to enable them to procure ancillary services (for voltage control and congestion management) in the day-ahead operation planning of distribution networks. This deliverable aims to describe in detail the functional and technical specifications of the developed tool, as well as its implementation in Julia programming language along with a user guide, which allows the potential users to successfully run the developed tool.

1. Sequential linearization algorithm (SLA)-based stochastic multi-period optimal power flow (S-MP-OPF) tool for ancillary services procurement in day-ahead operational planning of the distribution network (Task 4.1)

The accelerated penetration of renewable energy sources (RES) units, distributed energy resources (DER) and flexibility deployments in distribution networks has put them in a transition phase from passive network operation to active grid management. Consequently, advanced tools, which must take-into-account the features of the power systems of the future, are needed to model, analyze and optimally operate such networks. The existing tools, which distribution system operators (DSOs) currently use to manage their networks, either lack advanced modelling and analyzing features or are ineffectively utilized due to large computational costs and poor observability of distribution networks. Accordingly, future power networks can no longer be analyzed and operated based on the existing tools and thus there is a strong need to develop advanced tools, which must cater the features of such energy systems, being able to perform an in-depth analysis of these networks, thus leading to a reliable and optimal day-ahead and real-time operation by DSO. Furthermore, these tools must also foster a tight coordination between transmission system operator (TSO) and DSOs at their interfaces to share the flexibility present in the distribution networks.

The state-of-the-art tools for optimal operation planning of distribution networks consider the multi-temporal aspects but largely ignore the forecast uncertainty and interaction with TSO. Furthermore, existing tools are generally computationally costly, and these computational costs are expected to increase if the tools are extended to capture flexibility, uncertainties, interactions with the TSO and other factors. As a result, the scalability of these tools remains an open issue for real-world power systems. Consequently, to carry out the optimal operation of distribution systems of the future, it becomes incumbent to develop a tractable optimal power flow (OPF) tool that on the one hand, models every aspect of these grids and on the other hand, remains computationally efficient.

To achieve this overarching objective, the task T4.1 proposes a day-ahead operational planning tool for distribution networks which employs a novel tractable sequential linearization algorithm (SLA) to procure ancillary services by taking-into-account all aspects of future distribution networks such as flexible resources, uncertainty aspects of renewable DERs and interaction with TSO.

In the following, the high level functional and technical description of the developed tool is provided along with its implementation and execution details in Julia.

1.1. Functional description of SLA-based S-MP-OPF tool

The objective of this tool is to determine the optimal flexibility scheduling of available DER to support the procurement of ancillary services (congestion management and voltage control) by the DSO on a 24-hour basis. The tool optimizes the use of flexibility by mitigating RES units' uncertainties and ensure that network capacity is never exceeded during the real-time operation stage of distribution networks.

The developed tool takes-into-account (i) the uncertainty modelling of RES units, (ii) modern flexible DER such as electrical energy storages (EES) and flexible loads (FL), (iii) aggregated flexibility of low-voltage systems at medium-voltage/low-voltage interface, and (iv) actions of network management controllers of medium-voltage grid and interaction with TSO. At its input, the tool utilizes distribution network data as well as generation scenarios representing the uncertain behavior of RES.

Furthermore, the output of the TSO/DSO coordination mechanism defines the constraints of this tool to ensure that TSO and DSO do not procure conflicting ancillary services in the market. The tool then minimizes the overall cost of network operation which consists of the expected cost associated with the DER deviation from the market schedule i.e., the cost required to re-dispatch the active and reactive power, and provides optimal set-point values of distribution control means along with the cost of procuring the ancillary services at its output.

The resulting OPF problem, involving all the above-mentioned features, becomes a stochastic multi-period mixed-integer non-linear programming (MINLP) problem in its basic formulation [1] which cannot be solved for large-size real world power systems due to the current limitations of state-of-the-art MINLP solvers. Consequently, to break-down the high computational complexity of the resulting problem, a novel tractable methodology [2], based on mixed-integer linear model (MILP) and sophisticated heuristic SLA, is developed which ensures the tractability as well as scalability of the resultant tool. The first novelty of the proposed methodology is the formulation of MILP model, which develops novel linear approximations of all nonlinear constraints i.e., active and reactive power flows and branch currents, which ensure tractability. The second novelty is the proposal of SLA which improves the accuracy of proposed MILP model in an iterative manner. A detailed description of proposed methodology is provided in the next section.

A high-level functional diagram of the developed tool is shown in Fig. 1.

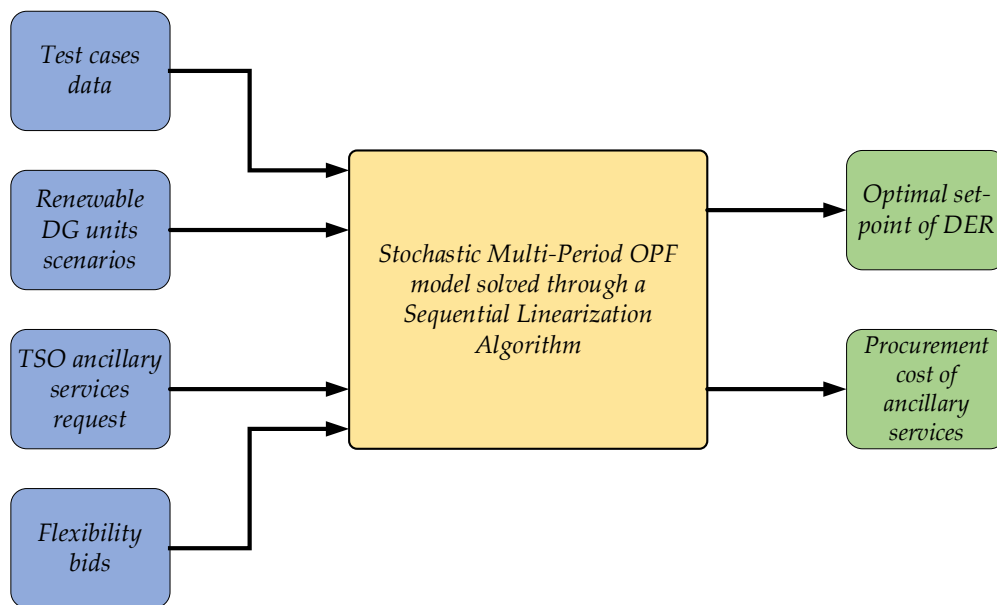


FIG. 1 HIGH LEVEL FUNCTIONAL DESCRIPTION OF TOOL 4.1

1.2. Technical description of SLA-based S-MP-OPF tool

The proposed tractable tool is based on stochastic multi-period mixed-integer linear programming (S-MP-MILP) OPF formulation and SLA which incorporates the S-MP-MILP model. A brief and high-level description of each part is given below.

1.2.1. Stochastic Multi-Period Flexibility Procurement Framework based on novel MILP Model (M1)

The proposed MILP model (M1) for flexibility procurement through S-MP-OPF framework in day-ahead operation planning of medium-voltage active distribution systems (ADSs) is presented below.

$$\min \sum_{s \in S} \sum_{t \in T} \pi_s \left\{ \left(\sum_{i \in G} c_{i,p}^{curt} P_{i,s,t}^{curt} + \sum_{i \in B} c_{i,b}^{str} (P_{i,s,t}^{dch} - P_{i,s,t}^{ch}) + c_{i,l}^{fl} \sum_{i \in F} (P_{i,s,t}^{od} + P_{i,s,t}^{ud}) \right) \Delta T \right. \\ \left. + c_{ij}^{olte} |r_{ij,s,t} - r_{ij,s,t}^0| + c_p^{viol} (P_{s,t}^{TD} - P_t^{TD})^2 + c_q^{viol} (Q_{s,t}^{TD} - Q_t^{TD})^2 \right\} \quad (1)$$

$$P_{i,t}^e + (P_{i,s,t}^{al} - P_{i,s,t}^{curt}) + P_{i,s,t}^{str} - (P_{i,s,t}^d + P_{i,s,t}^{od} - P_{i,s,t}^{ud}) = \sum_{j \in N} P_{ij,s,t} \forall i \in N, ij \in L \cup L^T \quad (2)$$

$$Q_{i,t}^e + Q_{i,s,t}^{res} - \left\{ Q_{i,s,t}^d + \tan(\varphi_{i,s,t}) (P_{i,s,t}^{od} - P_{i,s,t}^{ud}) \right\} = \sum_{j \in N} Q_{ij,s,t} \quad \forall i \in N, ij \in L \cup L^T \quad (3)$$

$$P_{ij,s,t} = \alpha_{i,s,t}^p V_{i,s,t}^2 + \alpha_{j,s,t}^p V_{j,s,t}^2 + \beta_{ij,s,t}^p \theta_{ij,s,t} + \gamma_{i,s,t}^p \quad \forall ij \in L \quad (4)$$

$$Q_{ij,s,t} = \alpha_{i,s,t}^q V_{i,s,t}^2 + \alpha_{j,s,t}^q V_{j,s,t}^2 + \beta_{ij,s,t}^q \theta_{ij,s,t} + \gamma_{i,s,t}^q \quad \forall ij \in L \quad (5)$$

$$P_{fj,s,t} = \alpha_{f,s,t}^p V_{f,s,t}^2 + \alpha_{j,s,t}^p V_{j,s,t}^2 + \beta_{fj,s,t}^p \theta_{fj,s,t} + \gamma_{f,s,t}^p \quad \forall fj \in L^T, j \in N, f \in N_f \quad (6)$$

$$Q_{fj,s,t} = \alpha_{f,s,t}^q V_{f,s,t}^2 + \alpha_{j,s,t}^q V_{j,s,t}^2 + \beta_{fj,s,t}^q \theta_{fj,s,t} + \gamma_{f,s,t}^q \quad \forall fj \in L^T, j \in N, f \in N_f \quad (7)$$

$$\underline{V}_i^2 \leq V_i^2 \leq \overline{V}_i^2 \quad \forall i \in N \quad (8)$$

$$(g_{ij}^2 + b_{ij}^2) \{ V_{i,s,t}^2 + V_{j,s,t}^2 - 2V_{i,s,t} V_{j,s,t} \cos(\theta_{ij,s,t}) \} \leq |\bar{I}_{ij}|^2 \quad \forall ij \in L \cup L^T \quad (9)$$

$$\underline{P}_i^e \leq P_{i,s,t}^e \leq \overline{P}_i^e \quad \forall i \in E \quad (10)$$

$$\underline{Q}_i^e \leq Q_{i,s,t}^e \leq \overline{Q}_i^e \quad \forall i \in E \quad (11)$$

$$0 \leq P_{g,s,t}^{curt} \leq P_{g,s,t}^{al} \quad \forall g \in G \quad (12)$$

$$-\tan(\bar{\phi}_{i,s,t}) (P_{i,s,t}^0 - P_{i,s,t}^{curt}) \leq Q_{i,s,t}^{res} \leq \tan(\bar{\phi}_{i,s,t}) (P_{i,s,t}^0 - P_{i,s,t}^{curt}) \quad \forall g \in G \quad (13)$$

$$0 \leq P_{l,s,t}^{od} \leq M \cdot P_{l,s,t}^d \cdot \chi_{l,s,t}^{od} \quad \forall l \in F \quad (14)$$

$$0 \leq P_{l,s,t}^{ud} \leq M \cdot P_{l,s,t}^d \cdot (1 - \chi_{l,s,t}^{od}) \quad \forall l \in F \quad (15)$$

$$\sum_{t \in T} P_{l,s,t}^{od} \cdot \Delta T = \sum_{t \in T} P_{l,s,t}^{ud} \cdot \Delta T \quad \forall l \in F \quad (16)$$

$$P_{b,s,t}^{str} = P_{b,s,t}^{ch} + P_{b,s,t}^{dch} \quad \forall b \in B \quad (17)$$

$$-\beta_{b,s,t}^{ch} \cdot P_b^{ch,r} \leq P_{b,s,t}^{ch} \leq 0 \quad \forall b \in B \quad (18)$$

$$0 \leq P_{b,s,t}^{dch} \leq (1 - \beta_{b,s,t}^{ch}) \cdot P_b^{dch,r} \quad \forall b \in B \quad (19)$$

$$SoC_{b,s,t} - SoC_{b,s,t-1} = \frac{\Delta T}{E_b^r} \left(-\eta_b^{ch} P_{b,s,t}^{ch} - \frac{P_{b,s,t}^{dch}}{\eta_b^{dch}} \right) \quad \forall b \in B \quad (20)$$

$$SoC_b \leq SoC_{b,s,t} \leq \overline{SoC}_b \quad \forall b \in B \quad (21)$$

$$SoC_{b,s,t_n} = SoC_{b,s,t_0} \quad \forall b \in B \quad (22)$$

$$V_{f,s,t}^2 = \sum_{k=1}^Y r_{if}^k \times \lambda_{if,s,t}^k \quad \forall if \in L^T, k \in Y \quad (23)$$

$$\underline{V}_i^2 \cdot \mu_{if,s,t}^k \leq \lambda_{if,s,t}^k \leq \overline{V}_i^2 \cdot \mu_{if,s,t}^k \quad \forall if \in L^T, k \in Y \quad (24)$$

$$\underline{V}_i^2 \cdot (1 - \mu_{if,s,t}^k) \leq V_{i,s,t}^2 - \lambda_{if,s,t}^k \leq \overline{V}_i^2 \cdot (1 - \mu_{if,s,t}^k) \quad \forall if \in L^T, k \in Y \quad (25)$$

$$\sum_{k=1}^Y \mu_{if,s,t}^k = 1 \quad \forall if \in L^T, k \in Y \quad (26)$$

$$\left| \sum_{k=1}^Y \{ r_{if}^k \times \lambda_{if,s,t}^k \} - r_{if,s,t}^0 \right| \leq \kappa_{if,s,t} \quad \forall if \in L^T, k \in Y \quad (27)$$

The objective (1) minimizes the overall cost of network operation which consists of expected cost of DER re-dispatch from the market schedule and (not shown explicitly for formulation ease) in each time-period t and scenario s , and deviation of active and reactive power flows at TSO-DSO interface.

The constraints (2) and (3) are the active and reactive power balance constraints at each node, (4)-(5) and (6)-(7) are the linear active and reactive power flows through a line and transformer, respectively, (8) is the limit on each bus voltage magnitude, (9) represents the longitudinal branch current limit, (10)-(11) limit the active and reactive power flows at main substation, (12) and (13) model the active power curtailment and reactive power output of RES units, respectively, (14) and (15) limit the over- and under-demand of active power of FLs, (16) maintains the energy balance of FLs over whole horizon, (18) and (19) limit the active power charging and discharging of EESs, (20) models the EESs energy balance, (21) constraints the SoC and (22) maintains the SoC equal on the first and last time periods.

Eqs. (23)-(26) represent the model of an OLTC transformer as an ideal transformer in series with the transformer impedance. The introduction of a fictitious node (f) leads to $V_f^2 = r_{if}^2 V_i^2$, which is handled by the standard product of binary-continuous variables linearization approach [3], thus leading to constraints (23)-(26). Lastly, (27) models the transformer operation cost term in the objective function.

The proposed MILP model, as can be noticed in (2)-(27), is based on (V^2, θ_{ij}) variable space and as such, the presented expressions are linear in terms of V^2 variables. The derivation of linear active and reactive power flows, and branch current expressions are not presented in this report. However, one can refer to [2], [4] where the derivation of these expressions is presented at length. Lastly, the coefficients (α, β, γ) in (4)-(5) and (9) are presented in Table 1.

TABLE 1: COEFFICIENTS OF VARIABLES IN LINEARIZED ACTIVE AND REACTIVE POWER FLOWS, AND LONGITUDINAL BRANCH CURRENT LIMIT EXPRESSION

Coefficients in P_{ij} (4)		Coefficients in Q_{ij} (5)		Coefficients in I_{ij} (9)	
$\alpha_{i,s,t}^P$	$p_1 \frac{(3V_{j,s,t}^{k-1} - V_{i,s,t}^{k-1})}{2(V_{i,s,t}^{k-1} + V_{j,s,t}^{k-1})}$	$\alpha_{i,s,t}^Q$	$-p_2 \frac{(3V_{j,s,t}^{k-1} - V_{i,s,t}^{k-1})}{2(V_{i,s,t}^{k-1} + V_{j,s,t}^{k-1})}$	$\alpha_{i,s,t}^I$	$\frac{c_1 V_{i,s,t}^{k-1} + c_2 V_{j,s,t}^{k-1}}{(V_{i,s,t}^{k-1} + V_{j,s,t}^{k-1})}$
$\alpha_{j,s,t}^P$	$p_1 \frac{(3V_{i,s,t}^{k-1} - V_{j,s,t}^{k-1})}{2(V_{i,s,t}^{k-1} + V_{j,s,t}^{k-1})}$	$\alpha_{j,s,t}^Q$	$-p_2 \frac{(3V_{i,s,t}^{k-1} - V_{j,s,t}^{k-1})}{2(V_{i,s,t}^{k-1} + V_{j,s,t}^{k-1})}$	$\alpha_{j,s,t}^I$	$\frac{c_2 V_{i,s,t}^{k-1} + c_1 V_{j,s,t}^{k-1}}{(V_{i,s,t}^{k-1} + V_{j,s,t}^{k-1})}$
$\beta_{ij,s,t}^P$	$p_2 V_{i,s,t}^{k-1} V_{j,s,t}^{k-1}$	$\beta_{ij,s,t}^Q$	$p_1 V_{i,s,t}^{k-1} V_{j,s,t}^{k-1}$	$\beta_{ij,s,t}^I$	$2s_{k-1} V_{i,s,t}^{k-1} V_{j,s,t}^{k-1}$
$\gamma_{i,s,t}^P$	$\frac{\{p_1 (V_{i,s,t}^{k-1} - V_{j,s,t}^{k-1})^2 / 2\} - p_2 V_{i,s,t}^{k-1} V_{j,s,t}^{k-1} \theta_{ij,s,t}^{k-1}}{p_2 V_{i,s,t}^{k-1} V_{j,s,t}^{k-1} \theta_{ij,s,t}^{k-1}}$	$\gamma_{i,s,t}^Q$	$\frac{\{-p_2 (V_{i,s,t}^{k-1} - V_{j,s,t}^{k-1})^2 / 2\} - p_1 V_{i,s,t}^{k-1} V_{j,s,t}^{k-1} \theta_{ij,s,t}^{k-1}}{p_1 V_{i,s,t}^{k-1} V_{j,s,t}^{k-1} \theta_{ij,s,t}^{k-1}}$	$\gamma_{i,s,t}^I$	$\frac{-2s_{k-1} V_{i,s,t}^{k-1} V_{j,s,t}^{k-1} \theta_{ij,s,t}^{(k-1)} - c_{k-1} (V_{i,s,t}^{k-1} - V_{j,s,t}^{k-1})^2}{c_{k-1} (V_{i,s,t}^{k-1} - V_{j,s,t}^{k-1})^2}$

$p_1 = -c_{k-1}g_{ij} - s_{k-1}b_{ij}$; $p_2 = s_{k-1}g_{ij} - c_{k-1}b_{ij}$; $c_1 = 1 + c_{k-1}$; $c_2 = 1 - 3c_{k-1}$; k is the iteration number

1.2.2. Sequential Linearization Algorithm

The proposed MILP model in the previous section is solved through the following SLA, shown in Fig. 2, which hinges on the assumption that instead of solving a S-MP-OPF framework based on MINLP model, solving a reduced number of times the S-MP-OPF framework based on the presented MILP model is much faster computationally and scalable. The steps of SLA are as follows.

1. Run AC power flow $s \times t$ times to obtain the *initial* point-of-linearization (PoL) in each scenario s / time-period t .
2. Calculate the coefficients of linearized active and reactive power injections, and branch current expressions, as reported in Table 1, using the initial PoL.
3. Build and solve the model M1 around the current PoL.
4. Evaluate the trustable approximation criterion (see section 1.2.3). If it is satisfied, go to step 7.
5. Update PoL based upon the value of trustable approximation criterion in the current inner-loop iteration i.e., either update PoL greedily by the recovered solution (voltage magnitude and angle) of model M1 or keep it the same in the next iteration if divergence in the model M1 solution, with respect to the previous inner-loop iteration, is detected.
6. Go to step 3 i.e., build and solve model M1 around the updated PoL.
7. Check the AC feasibility of the recovered approximated OPF (model M1) solution. If no constraints (voltage magnitude and branch loading limits) are violated, the algorithm terminates.
8. Update the PoL in each scenario s and time-period t by AC power flow solution (voltage magnitude and angle) determined in the outer loop and go to step 3.

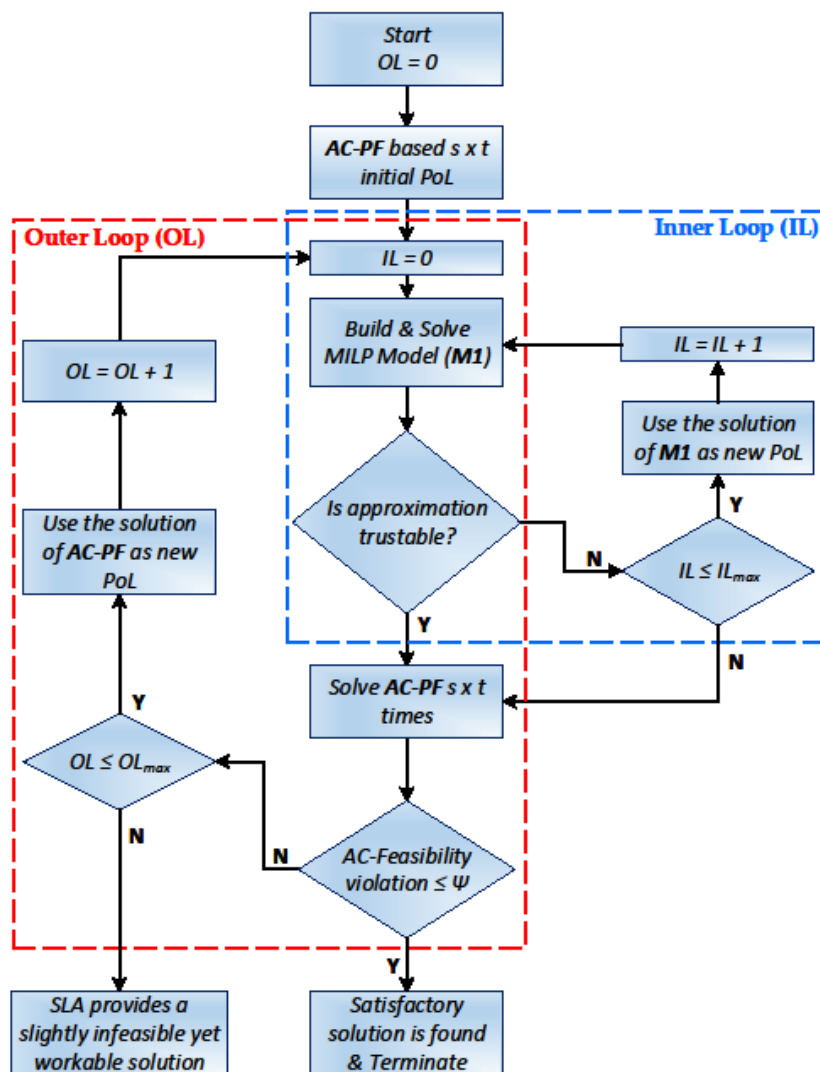


FIG. 2: FLOW CHART OF THE PROPOSED SEQUENTIAL LINEARIZATION ALGORITHM

1.2.2.1. Role of inner- and outer-loops

The objective of the inner-loop is to ensure the convergence of proposed model M1 by properly updating the PoL based on the value of trustable approximation criterion in the current inner-loop iteration.

It can be noticed that if criteria $\Delta S_i^{ll} < \Delta S_i^{ll-1}, \forall i \in N$ and $\Delta S_i^{ll} < \varepsilon^*, \forall i \in N$, where $\varepsilon^* > \varepsilon$, and ΔS_i is mentioned in (28), are satisfied, it means that the model M1 solution lies in the close vicinity of the optimal solution, and as such one can greedily update the PoL by using model M1 solution as a new PoL. Hence, the necessity of computing a new PoL through **s x t** AC power flow is avoided and the *estimated* AC power flow solution (nodes voltage magnitude and angles are recovered from model M1 and are not determined by running **s x t** AC power flow; thus, the recovered solution (termed as estimated AC power flow solution) is used as a new PoL.

On the other hand, if the above-mentioned criteria $\Delta S_i^{ll} < \Delta S_i^{ll-1}, \forall i \in N$ and $\Delta S_i^{ll} < \varepsilon^*, \forall i \in N$, are not satisfied, it means that the model M1 solution either moves towards the optimal solution but still far from it i.e., ($\Delta S_i^{ll} < \Delta S_i^{ll-1}, \forall i \in N$ but $\Delta S_i^{ll} > \varepsilon^*, \forall i \in N$) or it starts moving away from the optimal solution i.e., ($\Delta S_i^{ll} > \Delta S_i^{ll-1}, \forall i \in N$). Under both circumstances, the decision variables are restricted within a trust region, model M1 is solved again at the current PoL (or the best encountered solution in previous iterations can be used as a new PoL), which is updated, at the solution of model M1, in the subsequent inner-loop iteration.

The role of outer-loop is to determine the AC feasibility of the computed model M1 solution once trustable approximation criterion is satisfied in the inner-loop. This means that if all constraints (voltage magnitude and branch overload) are either satisfied or below the chosen feasibility threshold at the obtained model M1 solution, an AC feasible solution has been provided by the model M1 and, resultantly, SLA terminates. On the other hand, if constraints violate the feasibility threshold, the outer-loop updates the PoL at the solution of AC power flow and model M1 is solved again around the new PoL in the next inner-loop iteration.

1.2.3. Trustable Approximation Criterion

The normalized relative difference between the non-linear apparent power injection S^{nl} and its linear estimation S^{lin} at each node, calculated over the whole time-horizon and uncertainty scenarios, is chosen as the trustable approximation criterion (28). This chosen criterion evaluates the accuracy of proposed model M1 solution in the inner-loop of SLA as shown above.

$$\Delta S_i = \max \left\{ \frac{\sum_{s \in S} \sum_{t \in T} S_{s,t,i}^{nl} - \sum_{s \in S} \sum_{t \in T} S_{s,t,i}^{lin}}{S \times T \times \max(\sum_{s \in S} \sum_{t \in T} S_{s,t,i}^{nl})} \right\} \quad \forall i \in N \quad (28)$$

The satisfaction of approximation criterion ($\Delta S_i^{ll} < \varepsilon, \forall i \in N$) ensures that the probability of the obtained M1 solution to lie in the close vicinity of the benchmark M0 (S-MP-OPF MINLP model) solution is high. Resultantly, the optimality gap (difference between the optimal solutions of M0 and M1) can be zero or negligible. However, note that the satisfaction of this criterion does not guarantee that the obtained solution is AC feasible; this needs to be further verified by plugging the value of control variables obtained from M1 in the AC power flow.

A complete description of the proposed tractable tool can be found in [2].

1.3. Input and output requirements

The input and output data of developed S-MP-MILP based OPF tool is as follows.

1.3.1. Input data

The input data of this tool is

- i) Test cases which are developed in T2.3 (Test cases) of WP2 and datasets from Croatian DSO (HEP ODS) which are provided in WP7. The test cases provide the necessary network data such as the information related to network buses, lines, loads, transformers and generation units except the information about RES. Moreover, note that the available test data is enriched with the information about RES units, FLs and EES. Please refer to Appendix A.1 which depicts the network data information, as well as the additional information about the RES, FLs and EES.
- ii) Ancillary services bids offered by flexibility providers as defined in T2.6 (Market simulator).
- iii) RES units' (photo-voltaic and wind) uncertainty scenarios which are generated using Scenario Generation Tool. Please refer to appendices A.3 and A.4 for further information about the Scenario Generation Tool and uncertainty scenarios, respectively.
- iv) Specific TSO requests (from T4.4 of WP4) for ancillary services in terms of active and reactive power set-points at substations interfacing TSO and DSOs in order to ensure that TSO and DSOs do not procure conflicting ancillary services in the ancillary services market. This is shown in the objective function (1) of the proposed S-MP-MILP model (M1).

1.3.2. Output data

The output data of this tool is

- i) Expected cost of the procurement of ancillary services for congestion management and voltage control
- ii) Optimal set-points of each DER/optimal re-adjustment of flexible assets corresponding to various ancillary services during each uncertainty scenario and time-period, which is saved in an output excel file
- iii) Nodes voltage magnitude and angle, and magnitude of branch current
- iv) Total number of nodes voltage and branch current violation as well as the magnitude of maximum violation
- v) Total network losses

1.4. Implementation and run of SLA-based S-MP-OPF tool in Julia (User Guide)

The proposed S-MP-MILP OPF model is implemented and programmed in Julia language with JuMP being used as a modeling layer, and CPLEX is used to solve it. In the following, a brief description is provided about successfully running this tool in Julia.

1. The stand-alone version (i.e., without integrating the tool in ICT platform which is being developed in WP6 of ATTEST project) consists of 'main_sc_milp.jl' file which calls different packages, custom functions, being written/programmed in the course of tool development, and input files (test network data and scenario generation data), as shown in Fig. 3, to successfully run this tool.

2. The installation of Julia and packages is explained in Appendix A.2 (Installation of Julia and relevant Packages), whereas the generation of scenario data, which is carried out through **Scenario Generation Tool**, is explained in Appendix A.3 (Scenario Generation Tool). The output of this tool i.e., RES uncertainty data, is saved in 'scenario_gen.ods' file, as shown in Fig. 3, which is then called in 'main_sc_milp.jl' file.

```
main_sc_milp.jl | sql_loop.jl | interface_excel.jl | data_nw_functi... | data_reader.jl | sql_opf_model.jl
1  using JuMP, Ipopt, OdsIO, MathOptInterface, Juniper, Plots, LinearAlgebra, Dates,
2  CPLEX, Cbc, XLSX, DataFrames
3
4  #-----Accessing current folder directory-----
5  println(pwd());
6  cd(dirname(@__FILE__))
7  println(pwd());
8
9  ##### ----- Compilation of User-Defined Functions ----- #####
10 ##### -----
11 include("ac_power_flow_func_check.jl")
12 include("ac_power_flow_model_check.jl")
13 include("ac_power_flow_model_opf.jl")
14 include("ac_power_flow_opf_functions.jl")
15
16 include("data_nw_functions.jl")
17 include("power_flow_functitons_milp.jl")
18 include("loop_finding.jl")
19
20 include("opf_functions.jl")
21 include("sql_loop.jl")
22 include("sql_opf_model.jl")
23 include("opf_model_sub_functions.jl")
24 include("model_sol_recovery.jl") | > recovery_cost_flexibility_no_str
25
26 include("data_nw_functions_pf.jl")
27 include("non_linear_power_equations.jl")
28
29 include("data_types.jl")
30 include("data_reader.jl")
31 ##### -----
32 ##### ----- Select the Input file ----- #####
33 ##### -----
34 ##### ----- UK Network Files ----- #####
35 filename_mat = "input_data/uk_dx_01_2020.ods"
36 filename_addt = "input_data/uk_dx_01_2020_flex.ods"
37
38 ##### -----
39 ##### ----- Reading Input Data Files ----- #####
40 ##### -----
41 include("interface_excel.jl")
42 filename = "input_data/scenario_gen.ods"
43 include("interface_scenarios.jl")
44
```

FIG. 3: TRACTABLE TOOL MAIN FILE SHOWING PACKAGES, USER-DEFINED FUNCTIONS, AND INPUT FILES

3. The developed tool takes input data from two files for each network. The first file '**network_name_dx_01_2020.ods**' is the exact replica of the MATPOWER file that is provided in T2.3 data set, whereas the '**network_name_dx_01_2020_flex.ods**' contains all the additional data that is assumed by the tool developer, to enrich the data being provided in T2.3, to include the inter-temporal and uncertainty features in the tool T4.1. Note that '**network_name**' is replaced by the abbreviation of each country i.e., uk, pt etc.
4. After the selection of input test case file (i.e., the network which needs to be simulated), the next point would be the selection of flexibility options that need to be considered in the network. Fig. 4 shows the flexible options that can be activated (set option = 1)/deactivated (set option = 0) in a network. The selection of a particular flexible option can be done in 'constants.jl' file.

Moreover, another very important point that must be kept in mind that for OLTC transformer, FLs and EES, it is also required to activate/deactivate these flexible options in the input data file. This can be done by setting the '**status**' field (for FLs and EES) and '**status_oltc_transformer**' for OLTC transformer, equal to 1 (activate) / 0 (deactivate), in the '**Loads_Addt**', '**Storage_Addt**' and '**Lines_Addt**' sheets of '**network_dx_year_flex.ods**' file, respectively for these flexible options. For the remaining flexible options (i.e., active power curtailment and reactive power provision from RES), it is only required to set the status in the '**main_sc_milp.jl**' file.

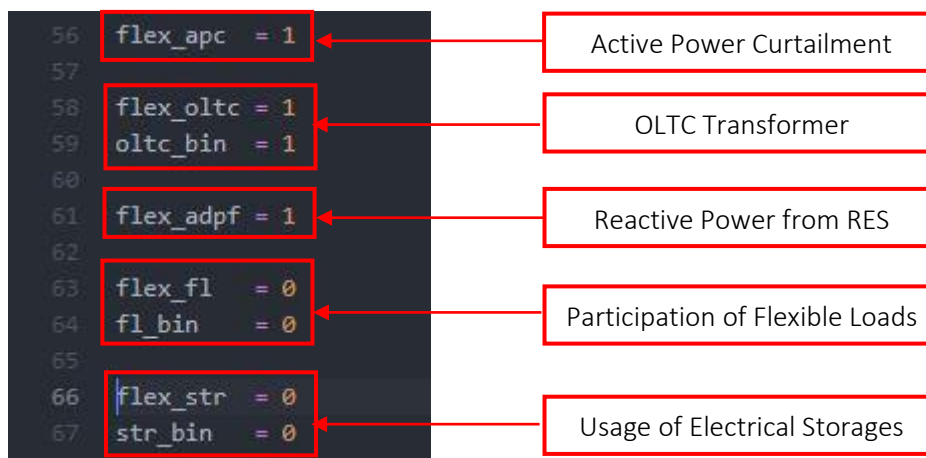


FIG. 4: AVAILABLE FLEXIBLE OPTIONS

5. Once flexible options are set, press the Run All button as shown in Fig. 5. This starts the execution of S-MP-MILP model.

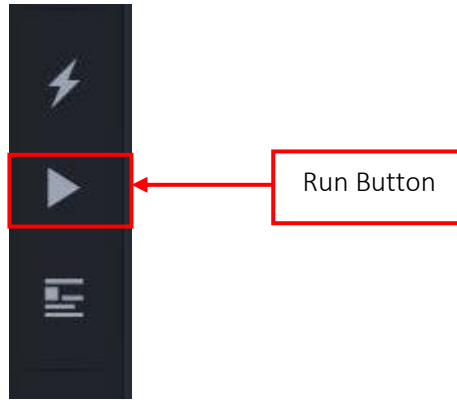


FIG. 5: RUN BUTTON OF JUNO IDE

6. During the course of execution, several user-defined functions are called which read the input test case data, simulate the AC power flow and execute the proposed tractable methodology (M1). In the following, a list of functions and/or files is shown to specify the role of each user-defined function and/or file being called in 'main_sc_milp.jl' file.

Functions/File	Brief Description
interface_excel (file)	Read the input test case data from .ods file
interface_scenarios (file)	Read the RES uncertainty data from 'scenario_gen.ods' file
Constants (file)	Contains input parameters that are to be set by the end-user of the tool
constants_pf_opf (file)	Parameters that are based on the input data as well as those defined in the 'constants' file and do not need to be updated/changed by the end-user
data_lines (function)	Takes the network lines information as an input and calculates line admittances
data_nodes (function)	Takes the nodes/buses information as an input and restructure the data into 'node_data' structure
data_load (function)	Takes the load data as an input and defines increment/decrement flexibility cost parameters along with the index information of FLs
data_gen (function)	Take the generation, both conventional and renewable, sources and define index information of non-curtable (conventional) and curtable (RES) generation sources
ac_power_flow_model_check (function)	Newton Raphson based AC power flow module
recovery_branch_current_pf (function)	Calculates the branch current once AC power flow is solved
constraint_violation_pf (function)	Calculates the nodes voltages and branch current violations after solving AC power flow

sql_loop (function)	Function which contains several sub-functions to implement the proposed tractable methodology M1
sql_opf_model (sub-function of sql_loop)	Function which implements the S-MP-MILP OPF model
opf_model_initialization (sub-function of sql_opf_model)	Function which initializes the decision and control variables of model M1
opf_model_objective (sub-function of sql_opf_model)	Function which defines the objective function of model M1
opf_model_power_balance_cons (sub-function of sql_opf_model)	Function which defines the linear active and reactive power balance, and nodes voltage constraint (2)-(8) of model M1
network_constraints_no_function_new_milp (sub-function of sql_opf_model)	Function which defines the constraints (9)-(27) of model M1
opf_model_solve (sub-function of sql_opf_model)	Function which calls the CPLEX solver and solves the model M1
opf_model_solution (sub-function of sql_opf_model)	Function which retrieves the values of decision and control variables after model M1 is solved
s_inj_power_max_power (sub-function of sql_opf_model)	Function which implements the trustable approximation loop criterion

2. Appendices

2.1. Appendix A.1

Appendix A. 1 presents an example of network data that acts as an input to the developed tool. The data format relies on the well-established MATPOWER format, which corresponds to deterministic and single period OPF problems. In the context of T4.1, the available data set is further enriched by multi-period load profiles, uncertainty scenarios of RES and time-coupled DER such as FLs and EES. As reported earlier in the point 3 of section 1.4, the basic information about the network such as number of buses, lines, single-period deterministic load and generation values is reported in the 'network_dx_01_2020.ods' file, whereas the additional information, as mentioned above, is presented in the 'network_dx_01_2020_flex.ods' file

In the following, nodes and branch data of a distribution network are shown in Figs. 6 and 7, respectively, whereas the information about FLs and EES as shown in Fig. 8 and Fig. 9, respectively.

bus_i	type	area	Vm	Va	baseKV	zone	Vmax	Vmin
8	1	1	1	0	35	1	1.1	0.9
9	3	1	1	0	110	1	1.1	0.9
10	1	1	1	0	35	1	1.1	0.9
11	1	1	1	0	35	1	1.1	0.9
12	1	1	1	0	35	1	1.1	0.9
13	1	1	1	0	35	1	1.1	0.9

FIG. 6: FORMAT OF BUS DATA (NETWORK_DX_01_2020.ODS FILE)

fbus	tbus	r	x	b	rateA	rateB	rateC	ratio	angle	status	angmin	angmax
10	15	0.021	0.029	0.0000	21	21	21	0	0	1	-360	360
10	11	0.014	0.010	0.0008	23	23	23	0	0	1	-360	360
10	23	0.007	0.009	0.0008	35	35	35	0	0	1	-360	360
11	12	0.021	0.029	0.0000	21	21	21	0	0	1	-360	360
12	13	0.014	0.010	0.0008	23	23	23	0	0	1	-360	360
13	8	0.014	0.010	0.0008	23	23	23	0	0	1	-360	360

FIG. 7: FORMAT OF BRANCH DATA (NETWORK_DX_01_2020.ODS FILE)

bus_i	Status	cost_inc	cost_dec
8	0	1	1
9	0	1	1
10	0	1	1
11	0	1	1
12	0	1	1
13	0	1	1
14	1	1	1
15	0	1	1

FIG. 8: INFORMATION ABOUT FLEXIBLE LOADS (NETWORK_DX_01_2020_FLEX.ODS)

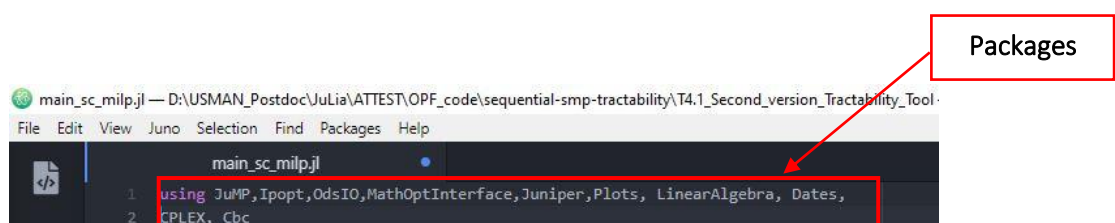
bus_j	Ps	Qs	energy	eRat	chRat	disRat	chEff	disEff	thermalRat	qmin	qmax	r	x	ploss	qloss	status	soc_0	socMin	socMax
16	0	0	1	1	0.5	0.5	0.9	0.9	10	-0.5	0.7	0	0.01	0	0	1	0.500	0	1
19	0	0	1	1	0.5	0.5	0.9	0.9	10	-0.5	0.7	0	0.01	0	0	1	0.500	0	1
29	0	0	1	1	0.5	0.5	0.9	0.9	10	-0.5	0.7	0	0.01	0	0	1	0.500	0	1

FIG. 9: FORMAT OF ELECTRICAL ENERGY STORAGE DATA (NETWORK_DX_01_2020_FLEX.ODS)

2.2. Appendix A.2 (Installation of Julia and relevant Packages)

Appendix A.2 provides the necessary information needed to install Julia and corresponding packages needed to run the proposed tractable tool.

1. Download and install **Julia** from the following link for Windows/Mac <https://julialang.org/downloads/> or download Julia for Visual Studio Code from <https://code.visualstudio.com/docs/languages/julia> which is a free editor for Julia language.
2. There are two available versions of Julia 1. “**Current stable release**” 2. “**Release with long term support**”. Download the “Current stable release” which right now is v1.7.3. The provided codes are developed using v1.4.2 but running it on the latest version is possible and user will not face any problem as the developers have claimed.
3. After the installation of Julia, the next step involves installing all the packages that are being used in the code. The packages that are being used in the code are defined at the start of ‘main_sc_milp.jl’ file (followed by the word “using”) as shown in the below figure.



4. The instructions for the installation of packages can be found at <https://docs.julialang.org/en/v1/stdlib/Pkg/> and are briefly explained below.
 - a. Let's say, one needs to install JuMP package. On Julia REPL, press] and one will enter Pkg REPL (Read Execution Print Loop).
 - b. Write "add JuMP" and press enter. This will start downloading and installing the package.
 - c. Once the package is installed (assuming no problem occurs), press *backspace* to go back to Julia REPL and executes the instructions mentioned in step 5.
5. Once the package is installed, it needs to be compiled before being used in the code. There are two ways of doing this.
 - a. One is to use the "using" command as shown in the above figure. Write "using JuMP" in Julia REPL and the package will start compiling.
 - b. The other way (very rarely, it can happen that "using" command does not compile the package successfully) is to use "Pkg.build("xxx")" command where "xxx" is the name of package.
6. Once all the packages are installed and compiled, the codes can be executed by pressing the "Run all" button on Julia command bar.

2.3. Appendix A.3 (Scenario Generation Tool)

1. The scenario generation tool generates the scenarios of PV and wind RES which become the input of SLA-based S-MP OPF tool. The main file of this tool is "**scenario_gen_tool.jl**" which calls Julia and Python packages as well as other Julia files. Note that this tool needs to be run before running the proposed tractable tool.
2. This tool calls several functions from Python using "PyCall" package which is an interface between Julia and Python. The 'PyCall' package uses *Conda.jl* Julia package and installs a minimal Python distribution via Miniconda that is private to Julia.
 - a. With this package installed, one can use "pyimport(xxx)" command to call the python function and use it in Julia. Let's say, one wants to use the **ARIMA** function which is available in the submodule ("**arima_model**") of Python package ("**statsmodels**"). One can simply call it in Julia by first importing the "arima_model"
 - i. `arima_py = Pyimport("statsmodels.tsa.arima_model")`
and subsequently, calls the ARIMA function using
 - ii. `arima_py.ARIMA()`
3. The input file of this code is "**data_scenarios.ods**" which takes the wind speed and solar power data and generates RES scenarios.
4. The output file of this code is "**scenario_gen.ods**" which contains the wind and solar RES uncertainty scenarios as well the joint probability of these scenarios. This file is then used in the '**main_sc_milp.jl**' file as shown in Fig. 3

2.4. Appendix A.4

Appendix A.4 presents the scenarios of wind and solar power which are generated using scenario generation tool. Figs. 8 and 9 show the wind and solar power data, respectively. In both figures, for the sake of simplicity, only 10 scenarios are shown and for each scenario, the values corresponding to a horizon of 10 hours are reported.

Wind Power Scenarios										
Scn/Time	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
s1	0.01	0.06	0.08	0.24	0.31	0.29	0.06	0.14	0.18	0.63
s2	0.30	0.52	0.35	0.12	0.33	0.36	0.30	0.12	0.11	0.06
s3	0.00	0.03	0.18	0.10	0.05	0.06	0.12	0.37	0.21	0.06
s4	0.17	0.18	0.11	0.09	0.19	0.25	0.12	0.03	0.04	0.17
s5	0.29	0.51	0.71	0.69	0.68	0.13	0.07	0.19	0.16	0.05
s6	0.11	0.15	0.13	0.16	0.14	0.11	0.11	0.26	0.11	0.27
s7	0.21	0.06	0.24	0.54	0.25	0.28	0.16	0.02	0.03	0.01
s8	0.06	0.12	0.25	0.08	0.07	0.11	0.29	0.21	0.18	0.20
s9	0.05	0.11	0.11	0.01	0.00	0.00	0.00	0.02	0.23	0.28
s10	0.03	0.01	0.07	0.21	0.33	0.44	0.24	0.16	0.16	0.25

FIG. 10: WIND POWER SCENARIOS

Solar Scenarios										
Scn/Time	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
s1	0.00	0.00	0.00	0.00	0.00	0.07	0.37	0.57	0.69	0.74
s2	0.00	0.00	0.00	0.00	0.00	0.09	0.40	0.52	0.71	0.75
s3	0.00	0.00	0.00	0.00	0.00	0.10	0.37	0.59	0.63	0.75
s4	0.00	0.00	0.00	0.00	0.00	0.08	0.39	0.57	0.66	0.70
s5	0.00	0.00	0.00	0.00	0.00	0.07	0.37	0.52	0.64	0.78
s6	0.00	0.00	0.00	0.00	0.00	0.10	0.39	0.52	0.65	0.76
s7	0.00	0.00	0.00	0.00	0.00	0.10	0.40	0.60	0.70	0.67
s8	0.00	0.00	0.00	0.00	0.00	0.08	0.41	0.57	0.72	0.71
s9	0.00	0.00	0.00	0.00	0.00	0.10	0.36	0.53	0.65	0.76
s10	0.00	0.00	0.00	0.00	0.00	0.06	0.38	0.55	0.68	0.76

FIG. 11: SOLAR POWER SCENARIOS

3. Bibliography

- [1] M. Usman and F. Capitanescu, "A Stochastic Multi-period AC Optimal Power Flow for Provision of Flexibility Services in Smart Grids," in *IEEE Power Tech*, Madrid, Spain, 2020.
- [2] M. Usman and F. Capitanescu, "A Novel Tractable Methodology to Stochastic Multi-Period AC OPF in Active Distribution Systems using Sequential Linearization Algorithm," *IEEE Transactions on Power Systems*, 2022.
- [3] W. Wei, "Tutorials on Advanced Optimization Methods," 2020.
- [4] M. Usman and F. Capitanescu, "Usman, M., & Capitanescu, F. (2021, September). A New Second-Order Linear Approximation to AC OPF Managing Flexibility Provision in Smart Grids," in *International Conference on Smart Energy Systems and Technologies (SEST)*, 2021.